Sequential Forward Selection Approach to the Non-unique Oligonucleotide Probe Selection Problem

Lili Wang[1], Alioune Ngom[1,*], and Luis Rueda[2,**]

[1] School of Computer Science, 5115 Lambton Tower
University of Windsor, 401 Sunset Avenue
Windsor, Ontario, N9B 3P4, Canada
{wang111v,angom}@uwindsor.ca
[2] Department of Computer Science,
University of Concepción. Edmundo Larenas 215,
Concepción, VIII Region, Chile
lrueda@udec.cl

Abstract. In order to accurately measure the gene expression levels in microarray experiments, it is crucial to design *unique*, highly specific and highly sensitive oligonucleotide probes for the identification of biological agents such as genes in a sample. Unique probes are difficult to obtain for closely related genes such as the known strains of HIV genes. The *non-unique* probe selection problem is to find one of the smallest probe set that is able to uniquely identify targets in a biological sample. This is an NP-hard problem. We present heuristic for finding near-minimal non-unique probe sets. Our method is a variant of the *sequential forward selection* algorithm, which used for feature subset selection in pattern recognition systems. The heuristic is guided by a probe set selection criterion which evaluates the efficiency and the effectiveness of a probe set in classifying targets genes as present or absent in a biological sample. Our methods outperformed all currently published greedy algorithms for this problem.

Keywords: Probe Selection, Gene Expression.

1 Introduction

Oligonucleotide microarrays are widely used tools, in molecular biology, providing a fast and cost-effective method for monitoring the expression of thousands of genes simultaneously [7]. In order to measure the expression level of a specific gene in a sample, one must design a microarray containing short strands of known DNA sequences of 8 to 30 bp, called *oligonucleotide probes*, which are

[*] Research partially funded by Canadian NSERC Grant #RGPIN228117-2006 and CFI grant #9263.

[**] Research partially funded by Chilean FONDECYT Grant #1060904.

M. Chetty, A. Ngom, and S. Ahmad (Eds.): PRIB 2008, LNBI 5265, pp. 262–275, 2008.

complementary to the gene's segments, called *targets*. These targets, if present in the sample, should bind to their complementary probes by means of *hybridization*. The success of a microarray experiment depends on how well each probe hybridizes to its target under specified experimental conditions such as temperature and salt concentration. However, choosing good probes is a difficult task since different sequences have different hybridization characteristics.

A probe is *unique*, if it is designed to hybridize to a single target. However, due to hybridization errors, there is no guarantee that unique probes will hybridize to their intended targets only. Many parameters such as secondary structure, salt concentration, GC content, free energy and melting temperature also affect the hybridization quality of probes [7], and their values must be carefully determined to design high quality probes. It is particularly difficult to design unique probes for closely related genes, given the probe length and melting temperature constraints. An alternative approach is to devise a method that can make use of *non-unique* probes, i.e. probes that are designed to hybridize to at least one target [7]. Also, a smaller probe set can be used with non-unique probes than can be with unique probes. Minimizing the number of probes in a microarray experiment is also a reasonable objective, since it is proportional to the cost of the experiment. The *non-unique probe selection problem* is to determine a smallest set of probes able to identify all targets present in a biological sample. This is an NP-hard problem [1], for which several approaches have been proposed recently [2][6][7][8][9].

Schliep *et al.* [7] first introduced the non-unique probe selection problem and described a simple but fast greedy heuristic, which computes an approximate solution that guarantees $s_{\min}$-separation for pairs of small target groups. Klau *et al.* [1] proposed two ILP formulations for this problem, respectively for single targets and for target groups, then solved it using the ILP solver CPLEX on pre-reduced problem instances. They also proved that the non-unique probe selection problem is NP-hard. Meneses *et al.* [2] proposed a deterministic greedy heuristic, for single targets only, which first constructs an initial feasible solution through local search, and then applies a reduction method to further reduce this solution. Ragle *et al.* [6] developed an *optimal cutting-plane* ILP heuristic, for single targets only, to find optimal solutions within practical computational limits. Wang *et al.* [8] proposed deterministic greedy heuristics that select probes based on their ability to help satisfy the constraints. Recently, Wang *et al.* [9] combined the probe selection functions with evolutionary methods and produced results that are at least comparable to those obtained by the method of [6], which is the best published approach for this problem.

2 Non-unique Probe Selection Problem

Given a target set, $T = \{t_1, \ldots, t_m\}$, and probe set, $P = \{p_1, \ldots, p_n\}$, an $m \times n$ *target-probe incidence matrix* $H = [h_{ij}]$ is such that $h_{ij} = 1$, if probe p_j hybridizes to target t_i, and $h_{ij} = 0$ otherwise. Table 1 shows an example of a matrix with $m = 4$ targets and $n = 6$ probes. A probe p_j *separates* two targets,

Table 1. A 4 × 6 target-probe incidence matrix

	p_1	p_2	p_3	p_4	p_5	p_6
t_1	1	1	0	1	0	1
t_2	1	0	1	0	0	1
t_3	0	1	1	1	1	1
t_4	0	0	1	1	1	0

t_i and t_k, if it is a substring of either t_i or t_k, that is, if $|h_{ij} - h_{kj}| = 1$. For example, if t_i = AGGCAATT and t_k = CCATATTGG, then probe p_j = GCAA separates t_i and t_k, since it is a substring of t_i only, whereas probe p_l = ATT does not separate t_i and t_k, since it is a substring of both targets [2]. Two targets, t_i and t_k, are *s-separated*, $s \geq 1$, if there exist at least s probes such that each separates t_i and t_k; in other words, the Hamming distance between rows i and k in H is at least s. For example, in Table 1 targets t_2 and t_4 are 4-separated. A target t is *c-covered*, $c \geq 1$, if there exist at least c probes such that each hybridizes to t. In Table 1, target t_2 is 3-covered. Due to hybridization errors in microarray experiments, it is required that any two targets be $s_{\min}$-separated and any target be $c_{\min}$-covered; usually, we have $s_{\min} \geq 2$ and $c_{\min} \geq 2$. These two requirements are called *separation constraints* and *coverage constraints*.

Given a matrix H, the aim of the non-unique probe selection problem is to find a minimal probe set that determines the presence or absence of specified targets, and such that all constraints are satisfied. In Table 1, if $s_{\min} = c_{\min} = 1$ and assuming that exactly one of $t_1, \ldots, t_4$ is in the sample, then the goal is to select a minimal set of probes that allows us to infer the presence or absence of a single target. In this case, a minimal solution is $\{p_1, p_2, p_3\}$ since for target t_1, probes p_1 and p_2 hybridize while p_3 does not; for target t_2, probes p_1 and p_3 hybridize while p_2 does not; for target t_3, probes p_2 and p_3 hybridize while p_1 does not; and finally for target t_4, only probe p_3 hybridize. Thus, each single target will be identified by the set $\{p_1, p_2, p_3\}$, if it is the only target present in the sample; moreover, all constraints are satisfied. For $s_{\min} = c_{\min} = 2$, a minimal solution that satisfies all constraints is $\{p_2, p_3, p_5, p_6\}$. Of course, $\{p_1, \ldots, p_6\}$ is a solution but it is not minimal, and hence is not cost-effective.

Stated formally, given an $m \times n$ matrix H with a target set $T = \{t_1, \ldots, t_m\}$ and a probe set $P = \{p_1, \ldots, p_n\}$, and a minimum coverage parameter $c_{\min}$, a minimum separation parameter $s_{\min}$ and a parameter $d_{\max} \geq 1$, the aim of the non-unique probe selection problem is to determine a subset $P_{\min} = \{q_1, q_2, \cdots, q_s\} \subseteq P$ such that:

1. $s = |P_{\min}| \leq n$ is minimal.
2. Each target $t_i \in T$ is $c_{\min}$-covered by some probes in $P_{\min}$.
3. Each target-pair $(t_i, t_k) \in T \times T$ is $s_{\min}$-separated by some probes in $P_{\min}$.
4. Each pair of small groups of targets is $s_{\min}$-separated by some probes in $P_{\min}$.

This problem was proved to be NP-hard, in [1], by performing a reduction from the *set covering* problem. It is NP-hard even for $c_{\min} = 1$ or $s_{\min} = 1$. The work of [1] formulated the non-unique probe selection problem as an *integer linear programming* (ILP) problem. Let $C = \{(i,k) \mid 1 \leq i < k \leq m\}$ be the set of all combinations of target indices. Assign $x_j = 1$ if probe p_j is chosen and 0 otherwise. We have:

$$\text{Minimize: } \sum_{j=1}^{n} x_j \ . \tag{1}$$

Subject to:

$$x_j \in \{0,1\} \qquad 1 \leq j \leq n \ , \tag{2}$$

$$\sum_{j=1}^{n} h_{ij} x_j \geq c_{\min} \qquad 1 \leq i \leq m \ , \tag{3}$$

$$\sum_{j=1}^{n} |h_{ij} - h_{kj}| \, x_j \geq s_{\min} \qquad 1 \leq i < k \leq m \ . \tag{4}$$

Function (1) minimizes the number of probes. The probe selection variables are binary-valued in Restriction (2). Constraints (3) and (4) are the coverage and separation constraints, respectively. Note that Constraints (4) are for single targets only. As opposed to this, in [1], another ILP formulation was proposed, which includes the separation constraints for small groups of targets. In this paper, we solve the ILP formulation, above, using a deterministic greedy heuristic based on a feature subset selection method used in pattern recognition. Note that one can easily check if the probes in the original set of candidate satisfy all the constraints. If not, then there are no feasible solutions. In this case, we can insert *unique virtual probes* in the original probe set only for those targets or target-pairs that are not $c_{\min}$-covered or $s_{\min}$-separated. This will ensure the existence of feasible solutions.

3 Probe Selection Functions

We want to select a minimum number of probes such that each target is $c_{\min}$-covered and each target-pair is $s_{\min}$-separated. Consider a target-probe incidence matrix, H, the parameters $c_{\min}$ and $s_{\min}$, the initial feasible candidate set of probes, $P = \{p_1, \ldots, p_n\}$, and the set of targets $T = \{t_1, \ldots, t_m\}$. Let P_{t_i} be the set of probes hybridizing to target t_i, and $P_{t_{ik}}$ be the set of probes separating the target-pair t_{ik}. A probe $p \in P_{t_i}$ is an *essential covering probe* if and only if $|P_{t_i}| = c_{\min}$. In Table 1, for instance, the probes in $P_{t_2} = \{p_1, p_3, p_6\}$ are essential covering probes if $c_{\min} = 3$. *Essential separating probes* are defined similarly. Essential probes must be contained in any minimal solution; that is, removing any such probe will make the solution unfeasible. A *redundant probe* is the one for which a feasible solution remains feasible when the probe is removed. Note that a probe may be redundant for some candidate solutions but non-redundant for others. There is a degree of redundancy between probes such that highly redundant probes are in very few or no minimal solutions. Our approach associates with each probe and each probe set a *degree of contribution* to minimal solutions (or, *degree of non-redundancy*)[8]. This degree corresponds to the ability of a probe, or a probe set, to help satisfy *all* the constraint.

3.1 Coverage Function

We want to choose the minimum number of probes such that each target is $c_{\min}$-covered. Given H, the parameter $c_{\min}$, the probe set $P = \{p_1, \ldots, p_n\}$ and the target set $T = \{t_1, \ldots, t_m\}$, we defined the function $\mathrm{cov}_{\mathrm{drc}} : P \times T \mapsto [0, 1]$ in [8] as follows:

$$\mathrm{cov}_{\mathrm{drc}}(p_j, t_i) = h_{ij} \times \frac{c_{\min}}{|P_{t_i}|}, \quad p_j \in P_{t_i}, \quad t_i \in T \ , \tag{5}$$

where, P_{t_i} is the set of probes hybridizing to target t_i; $\mathrm{cov}_{\mathrm{drc}}(p_j, t_i)$ is the amount that p_j contributes to satisfy the coverage constraint for target t_i. For target t_i, p_j is likely to be redundant for a larger value of $|P_{t_i}|$ and likely to be non-redundant for a smaller value of $|P_{t_i}|$. We defined the *coverage function* $C_{\mathrm{drc}} : P \mapsto [0, 1]$ in [8] as follows:

$$C_{\mathrm{drc}}(p_j) = \max_{t_i \in T_{p_j}} \{\mathrm{cov}_{\mathrm{drc}}(p_j, t_i) \quad | \quad 1 \leq j \leq n\} \ , \tag{6}$$

where T_{p_j} is the set of targets covered by p_j. $C_{\mathrm{drc}}(p_j)$ is the maximum amount that p_j can contribute to satisfy the minimum coverage constraints. Table 2 shows the coverage function table produced from Table 1.

Function C_{drc} favors the selection of probes that $c_{\min}$-cover targets t_i that have the smallest subsets P_{t_i}; these are the essential or near-essential covering probes. In Table 2, for example, target t_2 has the minimal value $|P_{t_2}| = 3$, and hence any probe that covers it can be selected first. In particular, function C_{drc} guarantees the selection of near-essential covering probes that $c_{\min}$-cover *dominated targets*; t_i *dominates* t_k if $P_{t_k} \subset P_{t_i}$. In Table 2, for example, t_3 dominates t_4 since $P_{t_4} = \{p_3, p_4, p_5\} \subset \{p_2, p_3, p_4, p_5, p_6\} = P_{t_3}$. Any $c_{\min}$-cover of the dominated target t_k will also $c_{\min}$-cover all its dominant targets, and

Table 2. Coverage function table obtained from Table 1

	p_1	p_2	p_3	p_4	p_5	p_6
t_1	$\frac{c_{\min}}{4}$	$\frac{c_{\min}}{4}$	0	$\frac{c_{\min}}{4}$	0	$\frac{c_{\min}}{4}$
t_2	$\frac{c_{\min}}{3}$	0	$\frac{c_{\min}}{3}$	0	0	$\frac{c_{\min}}{3}$
t_3	0	$\frac{c_{\min}}{5}$	$\frac{c_{\min}}{5}$	$\frac{c_{\min}}{5}$	$\frac{c_{\min}}{5}$	$\frac{c_{\min}}{5}$
t_4	0	0	$\frac{c_{\min}}{3}$	$\frac{c_{\min}}{3}$	$\frac{c_{\min}}{3}$	0
C_{drc}	$\frac{c_{\min}}{3}$	$\frac{c_{\min}}{4}$	$\frac{c_{\min}}{3}$	$\frac{c_{\min}}{3}$	$\frac{c_{\min}}{3}$	$\frac{c_{\min}}{3}$
C_{dps}	$\frac{c_{\min}}{6}$	$\frac{c_{\min}}{8}$	$\frac{c_{\min}}{4}$	$\frac{c_{\min}}{4}$	$\frac{c_{\min}}{6}$	$\frac{c_{\min}}{4}$

therefore, more targets are $c_{\min}$-covered. Probes covering the dominated target t_k have larger $\mathrm{cov}_{\mathrm{drc}}$ values than probes covering its dominant targets t_i, since $|P_{t_k}| < |P_{t_i}|$, and hence they will be selected first.

We would also like to favor the selection of *dominant probes*; p_j *dominates* p_l if $T_{p_l} \subset T_{p_j}$. In Table 2, for instance, p_6 dominates p_1 since $T_{p_1} = \{t_1, t_2\} \subset \{t_1, t_2, t_3\} = T_{p_6}$. Selecting dominant probes instead of dominated probes covers more targets. In the example, however, we have $C_{\mathrm{drc}}(p_1) = C_{\mathrm{drc}}(p_6)$, and hence p_1 could be selected for target coverage rather than p_6, depending on a particular order of the probes. On the other hand, p_6 dominates p_2 and $C_{\mathrm{drc}}(p_6) > C_{\mathrm{drc}}(p_2)$, and hence p_6 will be selected first. To favor the selection of a dominant probe among dominated probes equal in value C_{drc}, we penalize each probe p by an amount proportional to $|T_p|$, as follows:

$$C_{\mathrm{dps}}(p_j) = C_{\mathrm{drc}}(p_j) \times \frac{1}{m - |T_{p_j}| + 1} \ , \tag{7}$$

and probes that cover fewer targets are penalized more than probes that cover more targets. Table 2 shows the values of C_{dps} for each probe.

3.2 Separation Function

We want to choose the minimum number of probes such that each target-pair is $s_{\min}$-separated. We defined the function $\mathrm{sep}_{\mathrm{drc}} : P \times T^2 \mapsto [0, 1]$ in [8] as follows:

$$\mathrm{sep}_{\mathrm{drc}}(p_j, t_{ik}) = |h_{ij} - h_{kj}| \times \frac{s_{\min}}{|P_{t_{ik}}|}, \quad p_j \in P_{t_{ik}}, \quad t_{ik} \in T^2 \ , \tag{8}$$

where, $P_{t_{ik}}$ is the set of probes separating target-pair t_{ik}; $\mathrm{sep}_{\mathrm{drc}}(p_j, t_{ik})$ is what p_j can contribute to satisfy the separation constraint for target-pair t_{ik}. We defined the *separation function* $S_{\mathrm{drc}} : P \mapsto [0, 1]$ in [8] as follows:

$$S_{\mathrm{drc}}(p_j) = \max_{t_{ik} \in T^2_{p_j}} \{\mathrm{sep}_{\mathrm{drc}}(p_j, t_{ik}) \quad | \quad 1 \le j \le n\} \ , \tag{9}$$

where $T^2_{p_j}$ is the set of target-pairs separated by p_j. $S_{\mathrm{drc}}(p_j)$ is the maximum amount that p_j can contribute to satisfy the minimum separation constraints. The full separation function table can be found in [9].

Function S_{drc} also favors the selection of probes that $s_{\min}$-separate target-pairs t_{ik} which have the smallest subsets $P_{t_{ik}}$ and further favors the selection of near-essential separating probes that $s_{\min}$-separate *dominated target pairs*. To favor the selection of a dominant probe that has the same value, S_{drc}, as some of its dominated probes, we penalize each probe p by an amount proportional to $|T^2_p|$, as follows:

$$S_{\mathrm{dps}}(p_j) = S_{\mathrm{drc}}(p_j) \times \frac{1}{\frac{m(m-1)}{2} - |T^2_{p_j}| + 1} \ , \tag{10}$$

and probes that separate fewer target-pairs are penalized more than probes that separate more target-pairs.

3.3 Selection Function

We want to select the minimum number of probes such that all coverage and separation constraints are satisfied; that is, we must select a probe according to its ability to help satisfy both coverage *and* separation constraints. In [8], we combined functions C_{drc} and S_{drc} into a single probe selection function, $D_{\mathrm{drc}} : P \mapsto [0, 1]$ as follows:

$$D_{\mathrm{drc}}(p_j) = \max\{(C_{\mathrm{drc}}(p_j), S_{\mathrm{drc}}(p_j)) \mid 1 \leq j \leq n\} \ . \tag{11}$$

$D_{\mathrm{drc}}(p_j)$ is the degree of contribution of p_j, that is, the maximum amount required for p_j to satisfy all constraints. D_{drc} ensures that all essential probes p_j will be selected for inclusion in the subsequent candidate solution, since $C_{\mathrm{drc}}(p_j) = 1$ or $S_{\mathrm{drc}}(p_j) = 1$. With our definition of D_{drc}, probes p that cover dominated targets or separate dominated target-pairs have the highest $D_{\mathrm{drc}}(p)$ values. By selecting a probe p to cover a dominated target t_i or to separate a dominated target-pair t_{ik}, we are also selecting p to cover as many targets as possible (all targets that dominate t_i) or to separate as many target-pairs as possible (all target-pairs that dominate t_{ik}). This is the main greedy probe selection strategy in our heuristics in Section 5. In this paper, we use the following probe selection function, $D_{\mathrm{dps}} : P \mapsto [0, 1]$:

$$D_{\mathrm{dps}}(p_j) = \max\{(C_{\mathrm{dps}}(p_j), S_{\mathrm{dps}}(p_j)) \mid 1 \leq j \leq n\} \ , \tag{12}$$

to favor the dominant probes among all probes that have equal values in D_{drc}; this is the secondary greedy selection principle. These two greedy principles together allow larger coverage and separation when using D_{dps} than D_{drc} in a greedy search method.

4 Subset Selection Criteria

Given the initial probe set, $P = \{p_1, \ldots, p_n\}$, the sequential search algorithm, discussed in Section 5, greedily selects the best subset of probes among a collection, $\mathcal{P} \subseteq 2^P$, of subsets; 2^P is the power set of P. In this section, we define the criteria required to decide which is the best subset to select. Let $P^{1\ldots u} = \{q_1, \ldots, q_u\} \subseteq P$ be a probe set to be evaluated, where $q_j \in P, 1 \leq j \leq u$ and $1 \leq u \leq n$, and $P^{1\ldots 0} = \emptyset$. $P^{1\ldots u}$ $c_{\min}$-covers a target t_i if at least $c_{\min}$ probes in $P^{1\ldots u}$ cover t_i. $P^{1\ldots u}$ $s_{\min}$-separates a target-pair t_{ik} if at least $s_{\min}$ probes in $P^{1\ldots u}$ separate t_{ik}. Our aim is to select the subset $P^{1\ldots u}$ which $c_{\min}$-covers as many target as possible and $s_{\min}$-separates as many target-pairs as possible, or, which satisfies all the constraints with the least cardinality u.

4.1 Coverage Criterion

Given a collection $\mathcal{P} \subseteq 2^P$, we want to choose the subset $P^{1\ldots u} \subseteq P$ such that each target is $c_{\min}$-covered by $P^{1\ldots u}$. Given the matrix H, the parameter $c_{\min}$,

Table 3. Example of subset coverage obtained from Table 1

	$\{p_3\}$	$\cup$	$\{p_1\}$	$=$	P_{31}	P_{32}	P_{34}	P_{35}	P_{36}
t_1	0	$+$	$\frac{c_{\min}}{4}\frac{2}{4}$	$=$	$\frac{c_{\min}}{8}$	$\frac{c_{\min}}{8}$	$\frac{3c_{\min}}{16}$	0	$\frac{3c_{\min}}{16}$
t_2	$\frac{c_{\min}}{3}\frac{3}{4}$	$+$	$\frac{c_{\min}}{3}\frac{2}{4}$	$=$	$\frac{5c_{\min}}{12}$	$\frac{c_{\min}}{4}$	$\frac{c_{\min}}{4}$	$\frac{c_{\min}}{4}$	$\frac{c_{\min}}{2}$
t_3	$\frac{c_{\min}}{5}\frac{3}{4}$	$+$	0	$=$	$\frac{3c_{\min}}{20}$	$\frac{c_{\min}}{10}$	$\frac{3c_{\min}}{10}$	$\frac{c_{\min}}{10}$	$\frac{3c_{\min}}{10}$
t_4	$\frac{c_{\min}}{3}\frac{3}{4}$	$+$	0	$=$	$\frac{3c_{\min}}{20}$	$\frac{c_{\min}}{4}$	$\frac{c_{\min}}{2}$	$\frac{5c_{\min}}{12}$	$\frac{c_{\min}}{4}$
C_{dps}	$\frac{c_{\min}}{3}\frac{3}{4}$	$+$	$\frac{c_{\min}}{3}\frac{2}{4}$	$=$	$\frac{5c_{\min}}{12}$	$\frac{c_{\min}}{4}$	$\frac{c_{\min}}{2}$	$\frac{5c_{\min}}{12}$	$\frac{c_{\min}}{2}$

the candidate probe set $P = \{p_1, \ldots, p_n\}$ and the target set $T = \{t_1, \ldots, t_m\}$; to evaluate the ability of subset $P^{1\ldots u}$ to $c_{\min}$-cover T, we generalize the coverage function as follows:

$$C_{\text{dps}}(P^{1\ldots u}) = \max_{t_i \in T_{P^{1\ldots u}}} \left\{ \sum_{j=1}^{j=u} \text{cov}_{\text{drc}}(q_j, t_i) \times \frac{1}{m - \left|T_{q_j}\right| + 1} \quad \middle| \quad q_j \in P^{1\ldots u} \right\} \;, \quad (13)$$

where $T_{P^{1\ldots u}} = T_{q_1} \cup \ldots \cup T_{q_u}$ is the set of targets covered by $P^{1\ldots u}$. $C_{\text{dps}}(P^{1\ldots u}) : 2^P \mapsto \Re^+$ is the maximum amount that $P^{1\ldots u}$ can contribute to satisfy the minimum coverage constraints. Table 3 shows an example of a subset coverage table obtained from Table 1, given five subsets. In the example, P_{ab} means the subset $\{q_a, q_b\}$. We also show, for P_{31}, the computation of Equation (13).

Clearly, $C_{\text{dps}}(P^{1\ldots u})$ is maximal if $C_{\text{dps}}(q_j)$ is maximal for each $q_j \in P^{1\ldots u}$. Thus, for subsets of probes, function C_{dps} favors the selection of those subsets that contain probes having the highest coverage values. For example in Table 2, probes p_3, p_4 and p_6 have the highest coverage values, and hence, subsets such as P_{34} and P_{36} have the best values. C_{dps} indicates only how much a subset contributes in satisfying the coverage constraints, not how well the subset satisfies the coverage constraints. For instance, in the table, subsets P_{31} and P_{35} produce a tie, but P_{31} should be preferred since it covers more targets. Also, between the two subsets, which attain the same value of C_{dps}, the one that satisfies all coverage constraints (or, closer to satisfying all coverage constraints) should be preferred. We define the *coverage criterion*, $F_{C_{\text{dps}}} : 2^P \mapsto \Re^+$, as follows:

$$F_{C_{\text{dps}}}(P^{1\ldots u}) = C_{\text{dps}}(P^{1\ldots u}) \times \frac{|T_{P^{1\ldots u}}| - |U_{P^{1\ldots u}}|}{m - |U_{P^{1\ldots u}}|} \times \frac{\sum_{t_i \in T \setminus U_{P^{1\ldots u}}} \text{fea}\left(P_{t_i}^{1\ldots u}\right)}{(m - |U_{P^{1\ldots u}}|) \cdot c_{\min}} \;, \quad (14)$$

where, $U_{P^{1\ldots u}}$ is the set of targets already $c_{\min}$-covered by $P^{1\ldots u}$ (probes need not be selected to cover such targets); $P_{t_i}^{1\ldots u}$ is the set of probes in $P^{1\ldots u}$ that cover t_i, and fea : $2^P \mapsto \Re^+$ defined as

$$\text{fea}\left(P_{t_i}^{1\ldots u}\right) = \begin{cases} \left|P_{t_i}^{1\ldots u}\right| \text{, if } \left|P_{t_i}^{1\ldots u}\right| < c_{\min} \\ c_{\min} \text{ , otherwise} \end{cases} \;, \quad (15)$$

specifies how much the coverage constraint is satisfied on t_i; the sum equals $(m - |U_{P^{1\ldots u}}|)\, c_{\min}$ when all coverage constraints are satisfied. Hence, the second term penalizes subsets that cover fewer targets and the third term penalizes subsets that satisfy fewer coverage constraints. $F_{C_{\text{dps}}}$ is maximal when all three terms are maximal.

4.2 Separation Criterion

The derivation of the *separation criterion* is similar to that of coverage, except that we use terms and variables related to separation; such as, target-pair, $s_{\min}$, and so on, in the equations below. Given a collection $\mathcal{P} \subseteq 2^P$, we want to choose the subset $P^{1\ldots u} \subseteq P$ such that each target-pair is $s_{\min}$-separated by $P^{1\ldots u}$. Consider the matrix H, the parameter $s_{\min}$, the candidate probe set $P = \{p_1, \ldots, p_n\}$ and the target set $T = \{t_1, \ldots, t_m\}$. Following the same reasoning as in Section 4.1, we obtain the following equations for separation:

$$S_{\text{dps}}(P^{1\ldots u}) = \max_{t_{ik} \in T^2_{P^{1\ldots u}}} \left\{ \sum_{j=1}^{j=u} \text{sep}_{\text{drc}}(q_j, t_{ik}) \times \frac{1}{\frac{m(m-1)}{2} - \left|T^2_{q_j}\right| + 1} \quad | \ q_j \in P^{1\ldots u} \right\} , \tag{16}$$

where $T^2_{P^{1\ldots u}} = T^2_{q_1} \cup \ldots \cup T^2_{q_u}$ is the set of target-pairs separated by $P^{1\ldots u}$. $S_{\text{dps}}(P^{1\ldots u}) : 2^P \mapsto \Re^+$ is the maximum amount that $P^{1\ldots u}$ can contribute to satisfy the minimum separation constraints. The *separation criterion* is given by:

$$F_{S_{\text{dps}}}(P^{1\ldots u}) = S_{\text{dps}}(P^{1\ldots u}) \times \frac{\left|T^2_{P^{1\ldots u}}\right| - \left|U^2_{P^{1\ldots u}}\right|}{\frac{m(m-1)}{2} - \left|U^2_{P^{1\ldots u}}\right|} \times \frac{\sum_{t_{ik} \in T^2 \setminus U^2_{P^{1\ldots u}}} \text{fea}\left(P^{1\ldots u}_{t_{ik}}\right)}{\left(\frac{m(m-1)}{2} - \left|U^2_{P^{1\ldots u}}\right|\right) \cdot s_{\min}} , \tag{17}$$

where, $U^2_{P^{1\ldots u}}$ is the set of target-pairs already $s_{\min}$-separated by $P^{1\ldots u}$ (probes need not be selected to separate such target-pairs); $P^{1\ldots u}_{t_{ik}}$ is the set of probes in $P^{1\ldots u}$ that separate t_{ik}, and fea : $2^P \mapsto \Re^+$ defined as

$$\text{fea}\left(P^{1\ldots u}_{t_{ik}}\right) = \begin{cases} \left|P^{1\ldots u}_{t_{ik}}\right| , \text{ if } \left|P^{1\ldots u}_{t_{ik}}\right| < s_{\min} \\ s_{\min} , \text{ otherwise} \end{cases} , \tag{18}$$

specifies how much the separation constraint is satisfied on t_{ik}; the sum equals $\left(\frac{m(m-1)}{2} - \left|U^2_{P^{1\ldots u}}\right|\right) s_{\min}$ when all separation constraints are satisfied. Thus, the second term penalizes subsets that separate fewer target-pairs and the third term penalizes subsets that satisfy fewer separation constraints. $F_{S_{\text{dps}}}$ is maximal when all three terms are maximal.

4.3 Selection Criterion

As in the selection function of Section 3.3, we combine both the coverage criterion and the separation criterion into a single subset *selection criterion*

$$F_{D_{\text{dps}}}(P^{1\ldots u}) = \max\left\{ \ F_{C_{\text{dps}}}(P^{1\ldots u}) \ , \ F_{S_{\text{dps}}}(P^{1\ldots u}) \ \right\} , \tag{19}$$

which specifies the degree to which a subset of probes satisfies *all* constraints.

5 Sequential Forward Probe Selection Algorithm

In this section, a sub-optimal technique from pattern recognition is applied for the first time, to the best of our knowledge, to the non-unique probe selection problem. In particular, the well-known *sequential forward selection* (SFS) algorithm [5], for feature subset selection, is adapted to find near-minimal feasible probe sets. Feature selection (FS) constitutes one of the two principal phases of pattern recognition system design, the other being the design of pattern classification stage which employs the selected features. The main goal of FS is to select a subset of d features from the given set of D measurements, $d < D$, without significantly degrading or with possibly improving the performance of the recognition system. Given a suitable criterion function for assessing the *effectiveness* of feature subsets to classify data, FS is reduced to a combinatorial search problem that finds an optimal subset based on the selected measure. The SFS is among the methods[3][4][5] proposed by researchers to avoid searching the feature space exhaustively.

A microarray design experiment is a pattern recognition system where the measurements are provided by a biological sample and a target set (augmented with the set of all target-pairs, if non-unique probes are used), and where the classifier system is a probe set that classifies each target, or target-pair, as present or absent in the sample. However, with microarrays, the problem is to reduce the complexity of the classifier system (i.e., the size of the probe set) while still able to correctly classify each target and target-pair as present or absent in the biological sample. Here, the feature space representing the sample, which includes the targets and the target-pairs, is not subject to optimization.

We adapt the SFS to find a near minimal probe set as follows: the best probe set is constructed by adding, to the current non-feasible probe set, one probe at a time until we obtain a feasible probe set with the hope it has the least cardinality u. More specifically, to form the best feasible subset of probes, the starting point of the search is the empty set, $P^{1\ldots0}$, which is then successively built up. This is known as the bottom up approach. This method is generally sub-optimal since the best probe is always added to a working subset of probes, $P^{1\ldots u}$.

The *sequential forward probe selection* (SFPS) method (Algorithm 1) is based on the SFS algorithm. SFPS uses the $F_{D_{\mathrm{dps}}}$ function as the criterion for selecting the best subset among a collection of probe sets. The best probe, q^+, to insert in a working subset, $P^{1\ldots u}$, is the one that maximizes the criterion, $F_{D_{\mathrm{dps}}}$, when it is included. SFPS terminates when $P^{1\ldots u}$ is feasible; which is then reduced to a near-minimal solution, $P_{\min}$, in Algorithm 2, by removing the redundant probes.

SFPS locally searches the power set, 2^P, of the probe set P. That is, at each subset selection step, the neighborhood of the working subset $P^{1\ldots u} \in 2^P$ is the collection $\mathcal{P}^{1\ldots(u+1)} = \big\{P^{1\ldots u} \cup \{q_1\}, P^{1\ldots u} \cup \{q_2\}, \ldots, P^{1\ldots u} \cup \{q_{n-u}\}\big\} \subset 2^P$, $q_j \in P \setminus P^{1\ldots u}$ for $1 \leq j \leq n-u$. The subset to select is the one in $\mathcal{P}^{1\ldots(u+1)}$ that maximizes the criterion $F_{D_{\mathrm{dps}}}$.

Algorithm 1. Sequential Forward Probe Selection

```
Input: $T = \{t_1, \ldots, t_m\}$, $P = \{p_1, \ldots, p_n\}$, and $H = [h_{ij}]$
Output: Near-minimal solution $P_{\min}$
  Compute $D_{\mathrm{dps}}(p)$ for all $p \in P$;
  $u \leftarrow$ number of essential probes;
  $P^{1 \ldots u} \leftarrow$ set of essential probes;
  repeat
    $q^+ \leftarrow \arg\max_{q \in P \setminus P^{1 \ldots u}} F_{D_{\mathrm{dps}}}\left(P^{1 \ldots u} \cup \{q\}\right)$;
    $P^{1 \ldots (u+1)} \leftarrow P^{1 \ldots u} \cup \{q^+\}$;
    $u \leftarrow u + 1$;
  until $P^{1 \ldots u}$ is feasible;
  Return $P_{\min} \leftarrow \mathrm{Reduction}(P^{1 \ldots u}, P, T, H)$.
```

Algorithm 2. Reduction

```
Input: $P^{1 \ldots u}$, $P$, $T$, $H$
Output: Reduced solution $P_{\mathrm{red}}$
  $P_{\mathrm{red}} \leftarrow P^{1 \ldots u}$;
  $H \leftarrow H|_{P_{\mathrm{red}}}$, /* restrict to $P_{\mathrm{red}}$ */;
  Compute $D_{\mathrm{dps}}(q)$ for all $q \in P_{\mathrm{red}}$;
  Sort $P_{\mathrm{del}} \leftarrow \{q \in P_{\mathrm{red}} \mid D_{\mathrm{dps}}(q) < 1\}$ in increasing $D_{\mathrm{dps}}(q)$;
  if $P_{\mathrm{red}} \setminus \{p\}$ is feasible for each $q \in P_{\mathrm{del}}$ then
    $P_{\mathrm{red}} \leftarrow P_{\mathrm{red}} \setminus \{q\}$;
  end if
  Return $P_{\mathrm{red}}$.
```

6 Computational Experiments

We performed experiments to show the minimization ability of SFPS and that it outperform all the greedy methods currently published in literature for the non-unique probe selection problem. The programs were written in C and all tests ran on two Intel Xeon$^{\mathrm{TM}}$ CPUs 3.60GHz with 3GB of RAM under Ubuntu 6.06 i386.

We conducted experiments on ten artificial data sets and three real data sets, that were kindly provided by Dr. Ragle and Dr. Pardalos [6]. These data sets were used in all previous studies mentioned in Section 1, except for HIV-1 and HIV-2 sets which were used only in [2][6][8][9]. Table 4 shows, in the second and third columns, the dimension $|T| \times |P|$ (number of targets × number of probes) of the incidence matrix for each set (M for Meiobenthos is the largest set). Column A is the number of required virtual probes inserted into P to maintain the feasibility of the initial probe sets P. Due to space constraints, we refer the readers to [1][2][7] for the full details on the construction of these data sets. All experiments were performed with parameters $c_{\min} = 10$ and $s_{\min} = 5$, as in all previous studies.

Table 4 shows, for all data sets, the minimum sizes $|P_{\min}|$ attained by the greedy methods, GrdS of [7], GrdM of [2], DRC and DPS of [8], our SFPS

Table 4. Size of $P_{\min}$ for each heuristic

Set	$\lvert T\rvert$	$\lvert P\rvert$	A	GrdS	GrdM	DRC	DPS	SFPS	ILP
a1	256	2786	6	1163	568	549	547	530	503
a2	256	2821	2	1137	560	552	537	516	519
a3	256	2871	16	1175	613	590	577	557	516
a4	256	2954	2	1169	597	579	578	557	540
a5	256	2968	4	1175	605	583	571	558	504
b1	400	6292	0	1908	961	974	921	883	879
b2	400	6283	1	1885	976	1013	942	890	938
b3	400	6311	5	1895	951	953	915	896	891
b4	400	6223	0	1888	1001	1019	956	920	915
b5	400	6285	3	1876	1022	1019	969	933	946
M	679	15139	75	3851	2336	2084	2068	2036	3158
HIV-1	200	4806	20	-	531	487	472	468	-
HIV-2	200	4686	35	-	578	506	501	492	-

method, and the integer linear programming technique, ILP of [1]. In the table, the final $P_{\min}$'s include the virtual probes inserted into P.

Table 5 reports the improvements, Imp, of SFPS over GrdS, GrdM, DRC, DPS and ILP, computed as in Equation 20 below.

$$\mathrm{Imp} = \frac{P_{\min}^{\mathrm{SFPS}} - P_{\min}^{\mathrm{Heu}}}{P_{\min}^{\mathrm{Heu}}} \times 100 \ , \tag{20}$$

where Heu is either GrdS, GrdM, DRC, DPS or ILP. A negative (positive) value of Imp means that a SFPS result is Imp% better (worse) than Heu result. Consequently, Imp is negative when SFPS returns a probe set smaller than $P_{\min}^{\mathrm{Heu}}$. Therefore, the smaller the value of Imp, the better is SFPS.

SFPS substantially outperformed all the other greedy methods in all instances. GrdS and GrdM use different local search methods to find probes that satisfy the constraint on each target and target-pair. They use no probe selection function and thus, they do not *know* which probes are good or bad to select. DRC and DPS use a local search method similar to that in GrdM, but are guided by probe selection functions to decide which probes are best to select or not. SFPS uses the probe selection function, D_{dps}, of DPS but only to evaluate the effectiveness of each individual probe in a probe set. SFPF does not *select* the best probes, as in DPS, to construct a near minimal probe set; it uses the criterion $F_{D_{\mathrm{dps}}}$ to select the best subset from a collection of probe sets, as explained in Section 5. DPS locally searches the probe set P, where the neighborhood of a probe $q \in P$ is the set of probes that cover the same targets and separate the same target-pairs as q. SFPS locally searches the power set 2^P; which is *more global* than DPS search strategy. Therefore, as expected, SFPS performs better than

Table 5. Improvements of SFPS over GrdS, GrdM, DRC, DPS and ILP

Set	GrdS	GrdM	DRC	DPS	ILP
a1	−54.43	−6.69	−3.46	−3.11	+5.37
a2	−54.62	−7.86	−6.52	−3.91	−0.58
a3	−52.60	−9.14	−5.59	−3.47	+7.95
a4	−52.35	−6.70	−3.80	−3.63	+3.15
a5	−52.51	−7.77	−4.29	−2.28	+10.71
b1	−53.72	−8.12	−9.34	−4.13	+0.46
b2	−52.79	−8.81	−12.14	−5.52	−5.12
b3	−52.72	−5.78	−5.98	−2.08	+0.56
b4	−51.27	−8.09	−9.72	−3.77	+0.55
b5	−50.27	−8.71	−8.44	−3.72	−1.37
M	−47.13	−12.84	−2.30	−1.55	−35.53
HIV-1	-	−11.86	−3.90	−0.85	-
HIV-2	-	−14.88	−2.77	−1.80	-

DPS, DRC, GrdM and GrdS, due to its ability to assess the effectiveness of a probe set and its ability to search 2^P.

Also, SFPS achieved a greater reduction on the M set than all the others methods, including ILP. The authors of [1], first applied GrdS to reduce the initial probe sets (and to reduce the ILP running time), and then further optimized the *reduced probe sets* with ILP solver CPLEX (CPLEX is one of the leading mathematical programming software packages available and few heuristics, if any, are able to compete with its results). CPLEX was *restricted* to search only a small portion of the solution space, hence ILP was not aware of the full initial probe sets. SFPS had no such restriction. The improvements of SFPS over ILP are still quite small, but it implies that one could obtain better results than ILP, with better functions, than our D_{dps} or $F_{D_{\mathrm{dps}}}$, or with a better search method, than our SFPS method.

7 Conclusions and Future Research

In this paper, the sequential forward search algorithm is applied for the first time to solve the non-unique probe selection problem. SFPS outperformed all the currently published greedy algorithms for non-unique probes and gave results close to the optimal search method of ILP. SFPS also suffers from the *nesting effect* of SFS; that is, a probe that was selected cannot be discarded later to correct a wrong decision, and hence, the solution tends to be sub-optimal. The main cause of the nesting effect is the use of a non-monotonic criterion such as our $F_{D_{\mathrm{dps}}}$ criterion. We are investigating sequential methods, such as the *floating search* methods of [5], which reduces the nesting effect and cope with non-monotonic criterion functions.

References

1. Klau, G.W., Rahmann, S., Schliep, A., Vingron, M., Reinert, K.: Integer Linear Programming Approaches for Non-unique Probe Selection. Discrete Applied Mathematics 155, 840–856 (2007)
2. Meneses, C.N., Pardalos, P.M., Ragle, M.A.: A New Approach to the Non-Unique Probe Selection Problem. Annals of Biomedical Engineering 35(4), 651–658 (2007)
3. Moret, B.M.E., Shapiro, H.D.: On Minimizing a Set of Tests. SIAM Journal on Scientific and Statistical Computing 6(4), 983–1003 (1985)
4. Payne, R.W., Preece, D.A.: Identification Keys and Diagnostic Tables: a Review. Journal of the Royal Statistical Society, Series A 143(3), 253–292 (1980)
5. Pudil, P., Ferri, F.J., Novovičová, J., Kittler, J.: Floating Search Methods for Feature Selection with Nonmonotonic Criterion Functions. In: IAPR 12th International Conference on Pattern Recognition, Jerusalem, Israel, vol. 2, pp. 279–283 (1994)
6. Ragle, M.A., Smith, J.C., Pardalos, P.M.: An Optimal Cutting-Plane Algorithm for Solving the Non-Unique Probe Selection Problem. Annals of Biomedical Engineering 35(11), 2023–2030 (2007)
7. Schliep, A., Torney, D.C., Rahmann, S.: Group Testing with DNA Chips: Generating Designs and Decoding Experiments. In: IEEE Computer Society Bioinformatics Conference (CSB 2003), pp. 84–91. IEEE Press, Stanford (2003)
8. Wang, L., Ngom, A., Rueda, L., Gras, R.: Selection Based Heuristics for the Non-Unique Oligonucleotide Probe Selection Problem in Microarray Design. In: IEEE/ACM Transactions on Computational Biology and Bioinformatics (under review) (2008)
9. Wang, L., Ngom, A., Gras, R.: Non-Unique Oligonucleotide Microarray Probe Selection Method Based on Genetic Algorithms. In: 2008 IEEE Congress on Evolutionary Computation, Hong Kong, China, pp. 1004–1011. IEEE Press, Los Alamitos (2008)