A Model-Based Approach to the Non-Unique Oligonucleotide Probe Selection Problem

Lili Wang
School of Computer Science, University of Windsor
5115 Lambton Tower, 401 Sunset Avenue
Windsor, N9B 3P4, Ontario, Canada
01-519-253-3000
wang111v@uwindsor.ca

Alioune Ngom
School of Computer Science, University of Windsor
5115 Lambton Tower, 401 Sunset Avenue
Windsor, N9B 3P4, Ontario, Canada
01-519-253-3000
angom@cs.uwindsor.ca

ABSTRACT
In order to accurately measure the gene expression levels in oligonucleotide microarray experiments, it is crucial to design "unique", highly specific and sensitive probes for the identification of biological agents such as genes in a sample. It is difficult to design unique probes for very closely related genes, such as the known strains of HIV genes. The "non-unique" probe selection problem consists of determining a set of probes, not necessarily unique, that can uniquely identify targets while containing a minimal number of probes. In this paper, we describe a simple model-based method to obtain a near minimal non-unique probe set. Preliminary experimental results are encouraging and at least comparable, if not better, to those obtained by two recently published methods.

Keywords
Probe selection, Probabilistic model, Hybridization, Microarray.

1. INTRODUCTION
Oligonucleotide microarrays, commonly known as gene chips, are widely used techniques in molecular biology, providing a fast and cost-effective method of performing thousands of DNA hybridization experiments simultaneously [2][16]. Short (typically around 24 base pairs) strands of known DNA sequence, called probes, are affixed to specific positions on a chip's surface. A fluorescently labeled RNA sample is then washed over this surface. Some of this RNA will hybridize to complementary strands of DNA. The amount of RNA hybridized to each position on the microarray can be inferred from fluorescence measurements [14].

In order to measure the expression level of a specific gene in a sample, we must design a microarray containing DNA strands complementary to the gene. Typically, the total length of probes used to hybridize a gene is only a small fraction of the length of the gene [16]. The success of a microarray experiment depends on the quality of probe sets that are used [13]. Expression levels can only be accurately measured if a good set of probes is chosen. 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, there is no guarantee that unique probes will hybridize to their intended targets. Cross-hybridization, (hybridization to non-target sequences), self-hybridization (a probe hybridizing to itself) and non-sensitive hybridization (a probe may not hybridize to its present target) are hybridization errors that usually occur and must be taken into consideration for accurate measurement of the expression levels. Many parameters such as secondary structure, salt concentration, GC content, hybridization energy, melting temperature and so on, also affect the hybridization quality of probes [15] and their values must be carefully selected in the design of high quality probes. The design of unique probes is particularly difficult when targets to be identified are closely related and very similar to each other. Examples of difficult targets to be identified in a microarray experiment are the different strains of HIV or HPV viruses. One way around this problem is to devise a method that can make use of *non-unique* probes, which hybridize to more than one target. *Non-unique* probes are designed to hybridize to multiple targets, and this paper focuses on the non-unique probe selection problem. Applications include the detection of pathogenic bacteria in foods, laboratory diagnosis of bacteria responsible for infections including acute upper respiratory infections, detection of many human viruses, and the detection of viral RNA or DNA that is relevant to pathologies of the central nervous system [5].

Previous studies [1][3][5][9][10][11] applied various methods to *unique* and *non-unique* probe selection problems as well as other related probe selection problems, such as the String Barcoding method described by Rash *et al.* [9], the Maximum Distinguishing Probe Set (MDPS) and the Minimum Cost Probe Set (MCPS) problems [1]. Klau et al. [3] presented an Integer Linear Program (ILP) formulation for solving real and artificial instances of the non-unique probe selection problem. Meneses et al. [5] proposed a greedy non-random heuristic that produced at least comparable (and sometimes better) solutions than ILP solutions.

Bionetics'07, December 10-12, 2007, Budapest, Hungary.

2. NON-UNIQUE PROBE SELECTION PROBLEM

A probe is said to *separate* two sequences *a* and *b* if it is a substring of exactly one of *a* or *b*. For example, if *a* = AGGCAATT and *b* = CCATATTGG, the probe p_i = GCAA separates *a* and *b* since it is only a substring of *a*. While the probe p_j = ATT does not separate *a* and *b* since it is a substring of both *a* and *b* [5].

Assume we are given the target-probe incidence matrix $H = [h_{ij}]$ where h_{ij} =1 if and only if probe p_j hybridizes to target t_i and h_{ij} = 0 otherwise. Table 1 shows an example of a target-probe incidence matrix. Given a target-probe incidence matrix *H*, the goal is to select a minimal subset of probes that determines the presence or absence of specified targets. In Table 1, if only one of the targets $t_1, \ldots, t_4$ is in the sample, then the goal is to select a minimal set of probes that allow us to infer the presence of a single target. In this case, the set with probes p_1, p_2, p_3 is sufficient for detecting the presence of a single target, because for t_1, probe p_1 and p_2 hybridize while p_3 does not; for t_2, probe p_1 and p_3 hybridize while p_2 does not; for t_3, probe p_2 and p_3 hybridize while p_1 dose not; and finally for t_4, probe p_3 hybridizes but not p_1 or p_2.

The problem becomes more complicated when both targets t_1 and t_2 are in the sample. Suppose we use the set p_1, p_2 and p_3. This will cause a problem because the set hybridizes to all of the targets $t_1, \ldots, t_4$ and no distinction can be identified between the case where only t_1 and t_2 are in the sample and where t_3 is also in the sample. One possible solution is to select probes $p_1, \ldots, p_6$, but selecting all probes is not an economical solution, because the number of probes is proportional to the cost of experiment. In this case, using probes p_2, p_3 and p_5 instead of entire probe set will resolve this particular experiment. If we require that all pairs of targets be distinguishable, the entire probe set will not be sufficient and additional probes will be added.

Table 1. 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

Due to errors in microarray experiments, it is usually required that two targets be separated by more than one probe and that each target has more than one probe hybridized.

Two targets *a* and *b* are said to be *s*-separable if there exist *s* probes such that each separates *a* and *b*. A target *a* is said to be covered by a probe *p* if *p* hybridizes to *a*.

The non-unique probe selection problem consists of determining a minimal subset $P = \{p_1, p_2, \ldots, p_n\}$ of probes such that:

1. All probes exhibit high specificity and sensitivity, and satisfy the criteria of homogeneity.
2. All targets in $T = \{t_1, t_2, \ldots, t_m\}$ are covered by at least *c* probes from *P*.
3. All pairs of targets from *T* are separated by at least *s* probes from *P*.

Here *specificity* means that each probe should only hybridize to its subset of targets T_p and should not cross-hybridize to targets in $T - T_p$. Sensitivity means that each probe should hybridize to its low-abundant targets in T_p with clear signal. Homogeneity means that all probes are designed under the same experiment conditions such as hybridization temperature, melting temperature, salt concentration and so on. We assume that the probes are already designed to meet the criteria of specificity, sensitivity and homogeneity, and that the hybridization matrix is constructed. We are therefore only interested to determine an optimal subset of probes given the hybridization matrix, the initial probe set, and the target set.

Klau *et al.* [3] formulated the non-unique probe selection problem as an Integer Linear Program (ILP). Let $P = \{p_1 \ldots p_n\}$ denote the set of candidate probes, $T = \{t_1 \ldots t_m\}$ denote the set of targets, and $C = \{(i,k) : 1 \le i \le k \le m\}$ denote the set of all combinations of target indices. Let x_j be 1 if probe $p_j \in P$ is chosen and 0 otherwise. Then the non-unique probe selection is a constraint satisfaction problem [3] with objective function

$$\min \sum_{j=1}^{n} x_j$$

subject to constraints

1) $\sum_{j=1}^{n} h_{ij} x_j \ge c_{\min}$ for all $i \in \{1,\ldots,m\}$ [Coverage]

2) $\sum_{j=1}^{n} | h_{ij} - h_{kj} | \, x_j \ge h_{\min}$ for all $(i,k) \in C$ [Separation]

3) $x_j \in \{0,1\}$ j = 1, …, *n*

Where $| h_{ij} - h_{kj} |$ in the separation constraints stands for the absolute value of the difference between h_{ij} and h_{kj}. To satisfy the separation constraints, any two rows of the hybridization matrix should have a Hamming distance of at least h_{min}. Likewise, each target must be covered by at least c_{min} probes, to satisfy the coverage constraints. These two parameters, called coverage and separation parameters, are fixed constant in this paper.

Klau *et al.* [4] proved that the non-unique probe selection problem is NP-hard using a reduction from the set cover problem. Note that the coverage and separation constraints can easily be checked. If they are not satisfied then there is no feasible solution and the probe set is empty. It maybe necessary to add *unique virtual* probes into the initial probe set in order to ensure that feasible solutions will exist [3]. For example, when the coverage parameter is not satisfied for a given target then unique virtual probes, which hybridize only to that target, are added in order to meet the coverage requirements for the target.

3. MODEL-BASED APPROACH

The Bayesian optimization algorithm [6][7][8] combines the idea of using probabilistic models to guide optimization with methods for learning and sampling Bayesian networks. In Pelikan [6][7], Bayesian optimization algorithm evolves a population of candidate solutions to given optimization problem. The initial population is generated at random. The population is then updated for a number of iterations using selection and variation operators. We define probabilistic models in a way similar to [6] but without evolution or learning. The models represent the ability of the probes to satisfy the minimum coverage and separation constraints. We use the models only to guide our search for minimal non-unique probe sets. We designed a non-random greedy heuristic, guided by our models, to search for near optimal non-unique probe sets. The models are updated in an iterative manner.

3.1 Coverage Probabilistic Model

Given the target-probe incidence matrix H, the minimum coverage parameter c_{min} and a set of candidate probes P_{sol}, we can define for each target a uniform probability distribution over the set P_{sol} as

$$prob(p,t) = \begin{cases} \frac{c_{\min}}{|P_t|}, \text{ if } c_{min} \le |P_t| \\ 1 \quad , \text{ if } c_{min} > |P_t| \end{cases} \qquad p \in P_t, t \in T$$

where P_t is the set of probes that hybridize to target t, and $prob(p, t)$ is the amount of energy that p contributes in order to satisfy the coverage constraint for target t. For instance, given H in Table 1 and $P_{sol} = \{p_1,\ldots, p_6\}$, we obtain the coverage probability distribution matrix shown in Table 2.

We want to choose a minimum number of probes such that the minimum coverage constraint is satisfied for each target. The "greedy" strategy is to select probes that cover the largest number of targets. For example in Table 1, the best choice between p_2 and p_6 is p_6 because p_6 covers more targets than p_2. In Table 2, we can use the maximum probability value in column p_2 and in column p_6 as the objective value needed to make decision between p_2 and p_6. We define the coverage probability vector C over the set P_{sol} as $C(p_i) = C_i = \max\{prob(p_i, t) \mid t \in T_{p_i}, 1 \le i \le |P_{sol}|\}$, where T_{pi} is the set of targets covered by p_i. Each entry C_i in C represents the maximum amount of energy that p_i can contribute to satisfy the minimum coverage constraint, given c_{min} and all targets. Better definitions of C is possible, however this definition is simple and gives excellent results. Note that if $c_{min} > |P_t|$ for a given t, we add $(c_{min} - |P_t|)$ unique virtual probes which hybridize only to t, this ensure that our solution is feasible (or non-empty).

Table 2. Coverage Probabilistic Model

	p_1	p_2	p_3	p_4	p_5	p_6
t_1	$c_{min}/4$	$c_{min}/4$	0	$c_{min}/4$	0	$c_{min}/4$
t_2	$c_{min}/3$	0	$c_{min}/3$	0	0	$c_{min}/3$
t_3	0	$c_{min}/5$	$c_{min}/5$	$c_{min}/5$	$c_{min}/5$	$c_{min}/5$
t_4	0	0	$c_{min}/3$	$c_{min}/3$	$c_{min}/3$	0
C	$c_{min}/3$	$c_{min}/4$	$c_{min}/3$	$c_{min}/3$	$c_{min}/3$	$c_{min}/3$

3.2 Separation Probabilistic Model

We define a separation probabilistic distribution matrix and a separation probability vector S in a similar way as in previous section. In Table 3, t_{x-y} is the target pair (t_x, t_y), $1 \le x \le y \le m$, and the probes that separate pair t_{x-y} have non-zero values in that row. As for the coverage constraint, we also want to select the minimum number of probes that satisfy the minimum separation constraint. Therefore we define S_i as the maximum value in column p_i. Again, better definitions of S are possible. Also, if $h_{min} > |P_{t_{x-y}}|$, for a given pair t_{x-y}, we add $(h_{min} - |P_{t_{x-y}}|)$ unique virtual probes which separate only t_{x-y}, in order to generate the feasible solution ($P_{t_{x-y}}$ is the set of probes that separate pair t_{x-y}).

Table 3. Separation Probabilistic Model

	p_1	p_2	p_3	p_4	p_5	p_6
t_{1-2}	0	$h_{min}/3$	$h_{min}/3$	$h_{min}/3$	0	0
t_{1-3}	$h_{min}/3$	0	$h_{min}/3$	0	$h_{min}/3$	0
t_{1-4}	$h_{min}/5$	$h_{min}/5$	$h_{min}/5$	0	$h_{min}/5$	$h_{min}/5$
t_{2-3}	$h_{min}/4$	$h_{min}/4$	0	$h_{min}/4$	$h_{min}/4$	0
t_{2-4}	$h_{min}/4$	0	0	$h_{min}/4$	$h_{min}/4$	$h_{min}/4$
t_{3-4}	0	$h_{min}/2$	0	0	0	$h_{min}/2$
S	$h_{min}/3$	$h_{min}/2$	$h_{min}/3$	$h_{min}/3$	$h_{min}/3$	$h_{min}/2$

If $C_i = 1$ or $S_i = 1$ then probe p_i must be selected to satisfy the coverage or separation constraints since it contributes 100% of its energy.

4. MODEL-BASED ALGORITHM

As in Meneses *et al.*[5], our algorithm filters out probes that can't contribute to an optimal solution. The decision to exclude such probes is made *locally*. That is, it depends only on the current candidate probe set and the current models, in a given generation. The heuristic uses no past or global information to find a minimal set. Our method consists of three phases: *Initialization Phase*, *Construction Phase*, and *Reduction Phase*.

In the Initialization Phase, given the set of probes P and the incidence matrix H we build the initial model vectors C and S as described in previous section and create another vector V as $V_i = \max(C_i, S_i)$ that combines C and S. V is defined in such a way that only probes that contribute 100%, in either the coverage satisfaction or the separation satisfaction, are selected. All probes that are necessary to satisfy at least one constraint should be included in the initial solution set; such probes have values $V_i = 1$. V is then used to guide the construction and reduction phases. The reason of using V is that a probe needs to be selected to satisfy the coverage and separation constraints at the same time.

In the Construction Phase, we initially start with a candidate set P_{sol} that contains only probes with values $V(p) = 1$. We then add probes into P_{sol} from $P - P_{sol}$ to generate the feasible solution. There maybe some redundant probes in P_{sol}, but they will be deleted during the Reduction Phase to generate a near minimal final solution set.

In the Reduction Phase, we re-build the model vector V, using the current solution P_{sol} and the new incidence matrix implied by P_{sol}. We then attempt to delete probes p from P_{sol}, in increasing order of their values $V(p)$, such that $P_{sol} - \{ p \}$ remains feasible. We only try to delete those probes with value $V(p) < 1$. Next is our algorithm.

Initialization Phase:

1. Given incidence matrix H, probe set $P = \{p_1 \ldots p_n\}$, and target set $T = \{t_1 \ldots t_m\}$
    a. Compute the coverage probabilistic model vector $C = [C_1, C_2, \ldots, C_n]$;
    b. Compute the separation probabilistic model vector $S = [S_1, S_2, \ldots, S_n]$;
2. Set $V = [V_1, V_2, \ldots, V_n]$, $V_i = V(p_i) = \max(C_i, S_i)$
3. Add unique virtual probes, if necessary.
4. Generate initial solution set of probes as

$$P_{sol} = \begin{cases} \{p_i \in P \mid v(p_i) = 1, 1 \le i \le n\} \\ \phi, \quad if\ v(p_i) < 1, \forall i, p_i \in P. \end{cases}$$

Construction Phase:

1. For each target t not covered by at least c_{min} probes, add one probe p from $P - P_{sol}$ into P_{sol}, such that p hybridizes to t and has the highest possible value $v(p)$. Repeat this process one target at a time until the coverage constraint is satisfied for all such targets.
2. For each pair of targets not separated by at least h_{min} probes, add one probe p from $P - P_{sol}$ into P_{sol}, such that p distinguish this pair of targets and has the highest possible value $v(p)$. Repeat this process until the separation constraint is satisfied for each pair of targets.

Reduction Phase:

1. Update the incidence matrix H as $h_{ij} = 0$ for each $p_j \in P - P_{sol}$, $1 \le i \le m$, $1 \le j \le n$. Then re-compute new C, S and V models from H as in the Initialization Phase.
2. Set $P_{del} = \{ p \in P_{sol} \mid v(p) < 1 \}$ and sort P_{del} in increasing order.
3. Repeat: select p from P_{del}, if $P_{sol} - \{p\}$ is feasible then delete p from P_{sol}, until every probe in P_{del} has been tried.
4. Return final P_{sol} obtained in Step 3.

In the final solution, the selected probes are such that all the constraints are satisfied and P_{sol} is near minimal.

In the algorithm, there is no random selection and variation at any point. The candidate probes are sorted using $v(p)$ and attempted for deletion in that order. The final solution may not be an absolute optimal set since this algorithm uses a greedy selection strategy.

5. EXPERIMENTS

Experiments were conducted on two groups of data, which will be described in detail in the following section. Programs were implemented in C language and ran on a cluster of two SunFire V880 under Sun Solaris 10. Each machine has 8 SPARC 1.2GHz processors and 16 GB memory.

5.1 Data Description

Two groups of data have been used in the experiments. All the date sets described below were kindly provided to us by Dr. Pardalos [5]. Here we use the same data description and classification as in [5].

5.1.1 Group 1

This group is comprised of both artificial and real (Meiobenthos) data sets. Benthoses are the organisms that reside on the sea floor and at the bottoms of lakes and rivers [5].

A data set of 679 target Meiobenthos sequences was constructed by clustering 1230 28S rDNA sequences from different organisms present in the Meiobenthos and arbitrarily selected a representative from each cluster. 149 clusters contained two or more sequences each, and were representative of approximately 56% of all Meiobenthos sequences. Schliep *et al.* [12], Klau *et al.* [3] and Meneses *et al.* [5] used the same data set in their experiments.

The Random Evolutionary Forest Model (RANFOR) software was used to generate the artificial data in this group. In total, ten independent data sets were generated using two different Forest models. Sets a1-a5 with 256 targets each were constructed from one model, and sets b1-b5 with 400 targets each were constructed from the second model. The probes for each of the ten artificial data sets were generated using Promide software [5]. A detailed description of the artificial data set construction is given in Klau *et al.* [3]. Table 4 details the number of targets and probes for each data set.

Table 4. Numbers of targets and candidate probes for each data set in Group 1

Set	Targets	Probes
a1	256	2786
a2	256	2821
a3	256	2871
a4	256	2954
a5	256	2968
b1	400	6292
b2	400	6283
b3	400	6311
b4	400	6223
b5	400	6285
M	679	15139

5.1.2 Group 2

The second group of data consists of one HIV-1 data set and one HIV-2 data set. The HIV-1 and HIV-2 sequences were chosen in particular because of their biological significance and because the sequences were very closely related and similar within each set. This made them good candidates for the non-unique probe selection problem.

Two hundred sequences of each type were downloaded from NCBI (the National Center for Biotechnology Information). Candidate probes for the sequences were generated using Primer3 with default parameters, which included: length between 18 and 27 nucleotides, melting temperature between 57 and 63 ℃, and GC content between 20 and 80%. 40 probes for each sequence were generated for each data set, and duplicate probes were deleted before the target-probe incident matrix was constructed. Table 5 details the number of targets and probes for HIV-1 and HIV-2 data set used in experiments.

Table 5. Numbers of targets and candidate probes for each data set in Group 1

Set	Targets	Probes
HIV-1	200	4806
HIV-2	200	4686

5.2 Results

The input parameters for all experiments were $c_{min} = 10$ and $h_{min} = 5$. We first discuss results for Group 1 data, which were also used in [3][5]. Then we will discuss the results from the Group 2 data, which were used only in [5].

5.2.1 Results for Group 1 Data

In Table 6, we compared our results with that of [3] and [5] on the same Group 1 data sets. In the table, *Art* is the number of virtual probes added to obtain the final solution set; *ILP*, *Me* and *W* show, respectively, the size of the final solution obtained by *ILP* [3], *Me* [5], and *W* (our approach) on these data sets. The last three columns are the pairwise comparisons between the three methods, given as differences (*N* - *O*); where *N* is the newer method and *O* is the older method; the percentage of improvement, *Imp*%, of *N* over *O* is also given in parenthesis (a negative value signifies that *N* is *Imp*% better than *O* in terms of size of solution (also, the smaller is the percent improvement, the better is method *N*).

Table 6. Comparison of results for Group 1

Set	Art.	ILP	Me	W	(Me-ILP)	(W-Me)	(W-ILP)
a1	6	503	568	549	+65 (+13%)	-19 (-3%)	+46 (+9%)
a2	2	519	560	552	+41 (+7%)	-8 (-1%)	+33 (+6%)
a3	16	516	613	590	+97 (+19%)	-23 (-4%)	+74 (+14%)
a4	2	540	597	579	+57 (11%)	-18 (3%)	+39 (+7%)
a5	4	504	605	583	+101 (+20%)	-22 (-4%)	+79 (+16%)
b1	0	879	961	974	+82 (+9%)	+13 (+1%)	+95 (+11%)
b2	1	938	976	1013	+38 (+4%)	+37 (+4%)	+75 (+8%)
b3	5	891	951	953	+60 (+7%)	+2 (+0.2%)	+62 (+7%)
b4	0	915	1001	1019	+86 (+9%)	+18 (+2%)	+104 (+11%)
b5	3	946	1022	1019	+76 (+8%)	-3 (-0.3%)	+73 (+8%)
M	75	3158	2336	2084	-822 (-26%)	-252 (-11%)	-1074 (-34%)

Our method *W* performed significantly better than *ILP* and *Me* on the Meiobenthos data (which is the only real and large data set here); we performed substantially better than *Me* on the *a* data sets, but very close to *Me* on the *b* data sets. Note that *ILP* still performs better on the artificial *a*'s and *b*'s data sets than *Me* and *W*. In *ILP* [3], the absolute running times are in the range of 50-1700s, while in our method *W*, the absolute running times are reduced to 20-748s.

In *ILP* [3], the candidate probe sets were first reduced in size by a greedy heuristic algorithm and then CPLEX software was used to find the final solution. CPLEX (http://www.ilog.com/products/cplex/) is one of the leading mathematical programming software packages. Because only the probes from the reduced set were used by CPLEX, CPLEX was not aware of the additional candidate probes in the original pool. For the *M* data set, the candidate probes were reduced from 15,139 to 3851 by the greedy heuristic. CPLEX could only choose from the 3851 probes [5], whereas *Me* and *W* were both allowed to choose from all 15,139 probes.

In *Me* [5], there is no random selection at any point in the algorithms, but they only use the number of the targets to which each probe binds to sort the probes, and there is no other additional information used to direct the searching process. In the data sets, the range of the number of targets to which each probe binds is very small ([1, 40] for the *M* data set) and many probes have the same number of targets to hybridize. Thus given two candidate probes, it is not easy to identify which probe is better than another for inclusion into a candidate solution. In our method, the models store information about the current probe set in such a way that the algorithm can decide which probes to consider best for selection.

To see why we performed worse on the *b* data sets, we plotted the value $v(p)$ of probes p of each *a* data set and *b* data set. Figure 1 and Figure 2 show such plot for data set *a3* and *b2*. All the other *a* and *b* data sets behave similarly to *a3* and *b2*. Also, the *a* data sets contain more larger values $v(p)$ than the *b* data sets. Moreover, a great majority of probes in sets *b* have a value $v(p)$ below 0.3 during the Construction Phase. The *b* data sets were also created in such a way that they contain much more *unique* probes than *non-unique* probes, more than in the *a* data sets.

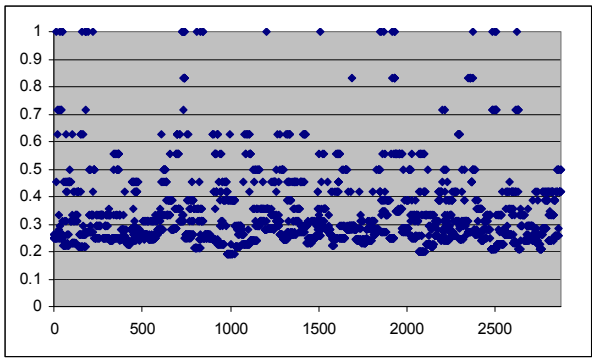

Figure 1. *v*(*p*) values in data set a3.

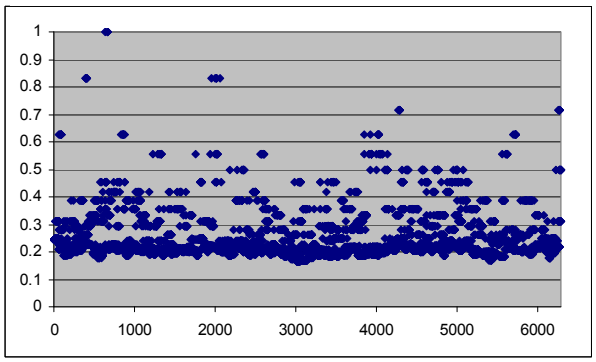

Figure 2. *v*(*p*) values in data set b2.

5.2.2 Results for Group 2 Data

We compared our results on the Group 2 data sets with those of [5]. See Table 7. We performed substantially better than *Me*. [3] on this group.

Table 7. Comparison of results for Group 2

Set	Art.	Me	W	(W-Me)
HIV-1	20	531	487	-44 (-8%)
HIV-2	35	578	506	-72 (-12%)

6. CONCLUSIONS

In this paper, we described a model-based algorithm to generate a near minimal set of non-unique oliginucleotide probes. Our approach used probabilistic models of coverage and separation constraints to guide the search for non-unique probes. Compared with the ILP method of [3] and the approach of [5], we obtained at least comparable results on most artificial data sets and better results on the real data sets.

In future work, we plan to use better models for coverage and separation constraints, and incorporate probe design factors required for high specificity and sensitivity. We are also currently developing evolutionary methods for the probe selection problems.

7. ACKNOWLEDGMENTS

We thank Dr. Pardalos for providing us with all the data sets used in this paper.

8. REFERENCES

[1] Borneman, J., Chroback, M., Vedova, G.D., Figueroa, A., and Jiang, T. 2001. Probe selection algorithms with applications in the analysis of microbial communities. Bioinformatics 17, s39-s48.

[2] Huang, Y., Chang, C., Chan, C., Yeh, T. Chang, Y., Chen, C., and Kao, C. 2005. Integrated minimum-set primers and unique probe design algorithms for differential detection on symptom-related pathogens. Bioinformatics 21, 4330-4337.

[3] Klau, G.W., Rahmann, S., Schliep, A., Vingron, M., and Reinert, K. 2004. Optimal robust non-unique probe selection using integer linear programming. Bioinformatics 20, i186-i193.

[4] Klau, G.W., Rahmann, S., Schliep, A., Vingron, M., and Reinert, K. 2007. Integer linear programming approaches for non-unique probe selection. Discrete Applied Mathematics 155, 840-856.

[5] Meneses, C.N., Pardalos, P.M., and Ragle, M.A. 2007. A new approach to the non-unique probe selection problem. Annals of Biomedical Engineering 35, 4, 651-658.

[6] Pelikan, M. 2005. Hierarchical Bayesian Optimization Algorithm. Springer, Berlin Heidelberg.

[7] Pelikan, M., Goldberg, D. E., and Cantú-Paz, E. 1999. BOA: The Bayesian optimization algorithm. In Proceeding of the Genetic and Evolutionary Computation Conference (GECCO-99), i525-i532.

[8] Pelikan, M., Ocenasek, J., Trebst, S., Troyer, M., and Alet, F. 2004. Computational complexity and simulation of rare events of Ising spin glasses. In Proceedings of the Genetic and Evolutionary Computation Conference (GECCO-2004), 2, 36-47.

[9] Rash, S., and Gusfield. 2002. String barcoding: Uncovering optimal virus signatures. In Proceedings of the Sixth Annual International Conference on Computational Biology (April 2002), 254-261.

[10] Rahmann, S. 2003. Fast large-scale oligonucleotide selection using the longest common factor approach. Journal of Bioinformatics and Computational Biology 1, 2, 343-361.

[11] Rahmann, S., Müller, T., and Vingron, M. 2004. Non-unique probe selection by matrix condition optimization. In Currents in Computational Molecular Biology 2004. San Diego Supercomputing Center, San Diego, USA.

[12] Schliep, A., Torney, D.C., and Rahmann, S. 2003. Group testing with DNA chips: generating designs and decoding experiments. In Proceedings of the 2nd IEEE Computational Systems Bioinformatics Conference (CSB'03), 84-93.

[13] Shin, S.Y., Lee, I.H., and Zhang, B.T. 2006. Microarray probe design using ε-multi-objective evolutionary algorithms with thermodynamic criteria. Lecture Notes in Computer Science, EvoBio 2006, 3907:184-195.

[14] Snustad, D.P. and Simmons, M.J. 1999. Principles of Genetics, 2nd Edition, Wiley, New York.

[15] Sung, W.K. and Lee, W.H. 2003. Fast and accurate probe selection algorithm for large genomes, In Proceedings of the 2003 IEEE Bioinformatics Conference (CSB 2003), 65-74.

[16] Tobler, J.B., Molla, M.N., Nuwaysir, E.F. Green R.D., and Shavlik, J.W. 2002. Evaluating machine learning approaches for aiding probe selection for gene-expression arrays. Bioinformatics 18, s164-s171.