Selecting oligonucleotide probes for whole-genome tiling arrays with a cross-hybridization potential.

Christoph Hafemeister, Roland Krause, and Alexander Schliep

Abstract—For designing oligonucleotide tiling arrays popular, current methods still rely on simple criteria like Hamming distance or longest common factors, neglecting base stacking effects which strongly contribute to binding energies. Consequently, probes are often prone to cross-hybridization which reduces the signal-to-noise ratio and complicates downstream analysis.

We propose the first computationally efficient method using hybridization-energy to identify specific oligonucleotide probes. Our Cross Hybridization Potential (CHP) is computed with a Nearest Neighbor Alignment, which efficiently estimates a lower bound for the Gibbs free energy of the duplex formed by two DNA sequences of bounded length. It is derived from our simplified reformulation of t-gap insertion-deletion-like metrics. The computations are accelerated by a filter using weighted ungapped q-grams to arrive at seeds.

The computation of the CHP is implemented in our software OSProbes, available under the GPL, which computes sets of viable probe candidates. The user can choose a tradeoff between running time and quality of probes selected.

We obtain very favorable results in comparison with prior approaches with respect to specificity and sensitivity for cross-hybridization and genome coverage with high-specificity probes. The combination of OSProbes and our Tileomatic method, which computes optimal tiling paths from candidate sets, yields globally optimal tiling arrays, balancing probe distance, hybridization conditions and uniqueness of hybridization.

1 Introduction

DNA microarrays have received a recent surge of interest due to their ability to investigate complete genomes with tiling arrays, which do not target specific transcripts of genes but rather cover the complete genome with oligonucleotide probes. Complete bacterial genomes can be represented densely on single microarray chips, expanding the use of tiling arrays beyond the most popular model organisms [1]. For the human genome, similar set-ups spanning several chips have been constructed. As several vendors make such low-volume custom microarrays available at highly competitive prices, targeted or whole genome studies elucidating gene expression, protein-DNA binding or chromosomal aberrations have become routine for many laboratories. Next generation sequencing technologies still signify a major investment and are not obtainable for all labs. Moreover, they are only competitive if their maximal throughput of DNA reads generated is actually used in experiments. In many settings, for example for microbial genomes, tiling arrays provide a good cost efficiency. Tiling arrays rather complement next generation sequencing for large eukaryotic genomes as they can be used to select specific genomic regions for sequencing [2], [3].

The computational challenge in designing tiling arrays is to find an optimal set of oligonucleotide probes, called a tiling path, which balances inter-probe distances, hybridization conditions and, most importantly, the potential of probes to cross-hybridize, that is to bind outside their intended target position on the genome. Cross-hybridization decreases the signal-to-noise ratio and greatly complicates downstream analysis. Most methods for normalization take all probes into account, see e.g. [4], and thus a sizable amount of cross-hybridizing oligonucleotide probes will in fact also change the normalized hybridization intensities of others. To address this problems, several approaches have been published [5], [6] which try to estimate the potential of cross-hybridization and correct for this effect on the probe level. Note that any thorough approach for estimating the potential of cross-hybridization would have to solve the very problem we are addressing, on a slighter smaller scale, namely for an individual chip. We argue that the computational effort is better spent on the design side, as is by no means guaranteed that one can remove the increases in variances and errors for *bad* probes post-facto, even if the potential of cross-hybridization can be assessed.

The problem of tiling array design can be broken down in two sub-problems: first, one needs to identify all probe candidates for a genome which have a low potential for cross-hybridization, recording relevant probe properties such as the melting temperature T_m. Second, one needs to compute a tiling path. The last problem we solved recently by proposing a linear-time algorithm

- *Christoph Hafemeister and Roland Krause are with the Department of Computational Molecular Biology, Max Planck Institute for Molecular Genetics, Ihnestr. 73, 14195 Berlin, Germany and the Free University Berlin, Germany. E-mail: {christoph.hafemeister, roland.krause}@molgen.mpg.de*
- *Alexander Schliep is with the Department of Computer Science and the BioMaPS Institute for Quantitative Biology, Rutgers The State University of New Jersey, 110 Frelinghuysen Rd, Piscataway, NJ 08854-8019 E-mail: schliep@cs.rutgers.edu*

which computes globally optimal tiling paths [7]. Some of the desired parameters for probe selection are trivial to compute, such as melting temperature, others, in particular the tendency to cross-hybridize are very complex and typically assessed by a simple heuristic.

Here, we propose a method for the generation and selection of oligonucleotide probe candidates for whole-genome tiling arrays based on their tendency to cross-hybridize. The *C*ross Hybridization Potential (CHP) is a novel measure of oligonucleotide probe binding specificity and based on thermo-dynamic calculations. Our *N*earest Neighbor Alignment (NNA) algorithm efficiently estimates a lower bound for the Gibbs free energy of the duplex formation of two DNA sequences of bounded length. It is derived from a simplified reformulation of t-gap insertion-deletion-like metrics [8]. To reduce the computational effort the Nearest Neighbor Alignment is only computed for cases which cannot safely be decided by faster hamming-distance based heuristics, for which we use gapped q-grams. Moreover, seeds for the alignment are computed with *weighted* ungapped q-grams which extend the q-gram formalism to include energy contributions from the thermodynamic model. Our method, including some routinely used filters, is implemented in our software OSProbes, which computes sets of viable probe candidates and which is available under the GPL. The computational costs of computing such candidate sets can be amortized over many tiling path computations. The combination of OSProbes with Tileomatic yields globally optimal tiling arrays, balancing probe distance, hybridization conditions and uniqueness of hybridization.

We compare our method to prior approaches with favorable results on a range of genomes measured by specificity and sensitivity for cross-hybridization and genome coverage with high-specificity probes.

Prior work: Assessing an oligonucleotide probe's potential to hybridize to an unintended position in the genome, thus giving spurious positive signals, has been studied intensively. Nevertheless the heuristic employed in many labs relies on a simple two-pass filter relying on thresholds known as Kane's first and second criteria [9]. Probes with an identity $>$ 75–80% to a non-target sequence, or contiguous perfect matches $>$ 25% of probe length are discarded. These filters are employed in many tools in use for microarray probe selection such as ROSO [10], GoArrays [11], OligoPicker [12], OligoWiz [13] and Oliz [14].

Incorporation of more accurate thermodynamic calculations were described as early as 2001 [1]. Probesel [15], Promide [16], OligoArray [17], [18] and ThermonucleotideBLAST [19] estimate probe specificity based on more accurate DNA thermodynamics. Some tiling-specific approaches focus on selecting probes in fixed windows [20], [21] but most practically relevant approaches rely on ad-hoc-methods and do not explicitly state more than one quality criterion for filtering the possible probes [22].

In the following we introduce the Cross Hybridization Potential, the Nearest Neighbor Alignment and the weighted q-gram filters. We describe additional filters, including a gapped q-gram filter for filtering based on hamming distance, and the implementation of our software. Computer experiments demonstrate the effectiveness of the Cross Hybridization Potential and its advantage over Kane's criteria. In the evaluation of the filter performance, our method compares very favorably to state-of-the-art methods for candidate generation and subsequent tiling path computation. We have calculated example candidate sets for typical small genomes and the complete human chromosome 1, demonstrating the feasibility to compute our measures for all possible probes of the human genome.

2 METHODS

2.1 Cross Hybridization Potential

We propose the Cross Hybridization Potential (CHP) as a measure for how likely it is that a probe will bind to a non-target sequence during a microarray experiment. The CHP is based on the scores of Nearest Neighbor Alignments (NNAs) at genome positions where cross-hybridization could possibly occur.

2.1.1 Nearest Neighbor Alignment

The Nearest Neighbor Alignment is an alignment which uses a scoring function that takes energy contributions from base stacking effects into account. It can be used to compute a lower bound of the free energy of duplex formation of two DNA sequences.

D'Yachkov et al. introduced t-gap block isomorphic subsequences and used them to describe abstract string metrics similar to the Levenshtein insertion-deletion metric. A particular variant of the T-gap insertion-deletion-like metrics captures key aspects of nearest neighbor thermodynamic modeling and defines a thermodynamic distance function for hybridized DNA duplexes [8].

In this section we present a slightly modified version of their algorithm and show how it is used to obtain a lower bound for the hybridization energy of two oligonucleotides. The resulting algorithm computes the score for the lowest scoring NNA, that can be interpreted as a lower bound for the Gibbs energy $\Delta_r G^\circ$ of DNA duplex formation.

We are interested in an alignment of two DNA oligonucleotides that we can interpret as a virtual secondary structure that the two molecules could form. We call this structure virtual, because the two DNA molecules are not expected to form this structure in solution. Instead, the score of the alignment obtained by the NNA algorithm will be used as a lower bound for the free energy associated with the DNA duplex that will form in vitro. This is possible, because the algorithm uses a simplified model of the hybridization energy and

TABLE 1
Thermodynamic weights of stacked pairs, in $\frac{kcal}{mol}$. For example, $F(\mathtt{G},\mathtt{A}) = -1.3$ denotes the free energy associated with the stacked pair $\begin{smallmatrix}5'\mathtt{GA}3'\\3'\mathtt{CT}5'\end{smallmatrix}$.

F	A	C	G	T
A	-1.00	-1.44	-1.28	-0.88
C	-1.45	-1.84	-2.17	-1.28
G	-1.30	-2.24	-1.84	-1.44
T	-0.58	-1.30	-1.45	-1.00

```
AAGA-TGTC---CCCGAAAGGTCAGTATAC
||||  |||      |||  ||||||||||||
AAGAG-GTCTAT--CGA-AGGTCAGTATAC
```

Fig. 1. An example of a Nearest Neighbor Alignment of two sequences of length 25. With a score of -22.3 it is the lowest-scoring alignment of the two sequences, i.e. the most stable virtual secondary structure.

takes only energetically favorable terms into account disregarding destabilizing structural elements.

First, we define the nearest neighbor score S_{nn}, which is a score for the thermodynamic stability of a sequence when being aligned to its reverse complement.

S_{nn} of a sequence s with length l is the sum of the thermodynamic weights of all adjacent base pairs, assuming a perfect matching Watson-Crick duplex

$$S_{nn}(s) = \sum_{i=1}^{l-1} F(s_i, s_{i+1}) \ . \tag{1}$$

Table 1 shows the free energy parameters used [23].

Scoring Scheme: The score of the alignment is the sum of nearest neighbor scores of all matched stretches with a minimum length of two. Mismatches and indels do not contribute to the score; they would only lead to destabilizing structures and can be omitted for the computation of a lower bound for the Gibbs energy. An example of a NNA and its score is given in Figure 1.

The use of this scoring scheme is motivated by the following observations:

- Indels or mismatches cannot increase the stability of the duplex.
- Hamming distance does not take sequence composition into account.
- Position dependence of mismatches [24], [25], is implicitly taken into account. Mismatches at the beginning or end of the sequences will disrupt only one stacked pair, whereas mismatches in the middle disrupt two stacked pairs.
- Many non-contiguous mismatches between two sequences lead to a high number of destabilizing structures [26].

Given the scoring scheme, we compute the lowest-scoring alignment of two sequences based on dynamic programming. Similar to other dynamic programming alignment algorithms, the NNA algorithm builds up an optimal alignment using previous solutions of optimal alignments of smaller subsequences.

Given two sequences $x = (x_1, x_2, \ldots, x_m)$ and $y = (y_1, y_2, \ldots, y_n)$, we compute a matrix $M : \{1, 2, \ldots, m\} \times \{1, 2, \ldots, n\} \rightarrow \mathbb{R}$, in which $M(i, j)$ equals the best score of the alignment of the two prefixes $(x_1, x_2, \ldots, x_i)$ and $(y_1, y_2, \ldots, y_j)$. Because a sequence of length one cannot be part of an alignment with stacked pairs, M is initialized with zeros in the first row and column, $M(i, 1) = 0$ for $i = 1, \ldots, m$ and $M(1, j) = 0$ for $j = 1, \ldots, n$. The remaining matrix fields are filled based on existing values and the nearest neighbor score of common suffices. $M(i, j)$ is computed recursively from the values of $M(i-1, j)$, $M(i, j-1)$, or the values on the upper left diagonal from $M(i, j)$, depending on the length of the longest common suffix of $(x_1, \ldots, x_i)$ and $(y_1, \ldots, y_j)$.

This gives rise to the following main recursion for filling the Matrix M,

$$M(i,j) = \min \begin{cases} M(i-1,j) \\ M(i,j-1) \\ D(i,j) \end{cases} , \tag{2}$$

where

$$D(i,j) = \min_{2 \le r \le \mathrm{lcs}} \left(S_{nn}(x_{[i-r+1,i]}) + M(i-r, j-r) \right) \tag{3}$$

and lcs is the length of the longest common suffix of $x_1, \ldots, x_i$ and $y_1, \ldots, y_j$. The notations $x_{[i-r+1,i]}$ and $x_{i-r+1}, \ldots, x_i$ are equivalent and used interchangeably. In the case that $\mathrm{lcs} < 2$, $D(i, j)$ defaults to 0. The NNA score of x and y, denoted as $\mathrm{nna}(x, y)$, is then given as the value of the matrix at $M(m, n)$. From the main recursion if follows that $\mathrm{nna}(x, x) = S_{nn}(x)$.

An example: Let $x = \mathtt{GAAAGG}$ and $y = \mathtt{CGAAGG}$ be two sequences with length $m = n = 6$. The score matrix after running the NNA algorithm is shown in Table 2. The gray fields are those of matched bases contributing to the final score. The score of the best NNA of x and y is -4.42, the value in $M(6, 6)$. For descriptive purposes we also show the traceback matrix TB for this example (Table 3). Here the arrows indicate which value of the three choices in Equation 2 was the minimum. In the case of $D(i, j)$, the number behind the arrow indicates which value of r minimized Equation 3. The resulting alignment is $\begin{smallmatrix}\mathtt{-GAAAGG}\\\mathtt{CGA-AGG}\end{smallmatrix}$ in single-base form, and, to better see the matched stacked pairs, in dinucleotide form $\begin{smallmatrix}\mathtt{--\ GA\ AA\ AA\ AG\ GG}\\\mathtt{CG\ GA\ A-\ -A\ AG\ GG}\end{smallmatrix}$. This example also shows why the loop over $2 \le r \le \mathrm{lcs}$ in Equation 3 is needed. This loop makes sure that potential gaps are placed at optimal positions maximizing the sum of the aligned base pairs. A greedy algorithm with $r = 2$ would lead to $\begin{smallmatrix}\mathtt{-GAAAGG}\\\mathtt{CGAA-GG}\end{smallmatrix}$ (score -4.14) and $r = \mathrm{lcs}$ would lead to $\begin{smallmatrix}\mathtt{--GAAAGG}\\\mathtt{CG--AAGG}\end{smallmatrix}$ (score -4.12), both not yielding the optimal score.

In our further application of the NNA x will be a probe candidate and y a genome region of the same length. This length limitation allows the NNA to produce a sharper bound on the free energy but gives rise

TABLE 2
M matrix after running the NNA algorithm with $x = \texttt{GAAAGG}$ and $y = \texttt{CGAAGG}$. Gray fields show matched bases.

M	C	G	A	A	G	G
G	0.0	0.0	0.0	0.0	0.0	0.0
A	0.0	0.0	-1.3	-1.3	-1.3	-1.3
A	0.0	0.0	-1.3	-2.3	-2.3	-2.3
A	0.0	0.0	-1.3	-2.3	-2.3	-2.3
G	0.0	0.0	-1.3	-2.3	-2.58	-2.58
G	0.0	0.0	-1.3	-2.3	-2.58	-4.42

TABLE 3
Traceback matrix after running the NNA algorithm with $x = \texttt{GAAAGG}$ and $y = \texttt{CGAAGG}$. Arrows indicate where the value in the corresponding M field derived from. The number behind the arrow indicates wich value of r yielded the minimum.

TB	C	G	A	A	G	G
G		←	←	←	←	←
A	↑	↑	↖ 2	←	←	←
A	↑	↑	↑	↖ 3	←	←
A	↑	↑	↑	↑	↑	↑
G	↑	↑	↑	↑	↖ 2	←
G	↑	↑	↑	↑	↑	↖ 3

to the chance of missing cross-hybridizing targets that span more than the probe candidate length. However, we have experimentally shown that this problem does not occur frequently (see supplemental material) and chances of making this kind of error are negligible.

Running time: For two sequences with the same length of k, the matrix M has k^2 fields. In the worst case, the alphabet size of the concatenation of both sequences is one, i.e. both sequences are stretches of the same single nucleotide. In this case, the lcs in Equation 3 will always be $\min(i, j)$ at every $D(i, j)$. The worst-case running time is thus $O(k^3)$. However, for our application to DNA sequences this is an unlikely scenario. If we assume two sequences independently generated from the i.i.d. model, the probability of having a lcs of length 0 is $P(\mathrm{lcs} = 0) = \frac{3}{4}$, $P(\mathrm{lcs} = 1) = \frac{1}{4}$, $P(\mathrm{lcs} = 2) = \frac{1}{16}$, $P(\mathrm{lcs} = n) = \frac{1}{4^n}$. The expected length of a lcs at field $M(i, j)$ is thus

$$E(\mathrm{lcs}) = 1\frac{1}{4} + 2\frac{1}{16} + 3\frac{1}{64} + \ldots + n\frac{1}{4^n} = \sum_{i=1}^{n} \frac{n}{4^n} \qquad (4)$$

where n is the minimum of i and j. As this sum quickly converges to 0.44, it is sufficient to assume this value as $E(\mathrm{lcs})$ in every field of the matrix. As a result, for two independent i.i.d. model sequences the expected run time is thus $< O(1.44 \cdot k^2)$. Genomic sequences are not i.i.d. and real world lcs values are on average smaller than 0.44. For the 100 million randomly selected NNA computations of the experiment described in Section 3.3 the lcs has a mean of 0.36 with a variance of 0.003.

2.1.2 From Nearest Neighbor Alignment Scores to Cross Hybridization Potential

In the previous section we introduced the Nearest Neighbor Alignment and its score, a lower bound for the free energy of oligonucleotide hybridization. We now motivate the Cross Hybridization Potential, which we interpret as a specificity measure, and use it to rank the given probes by their quality.

During a microarray experiment, there should be a high concentration of targets equally distributed over all probes. The intended target and unintended targets compete for the probe on the chip, with the duplex of probe and intended target having the greatest stability. We assume that hybridization can occur when a probe-target pair has a NNA score greater than a certain threshold. This threshold depends on the nearest neighbor score S_{nn} of the probe and a free energy value ΔE. We define ΔE as the minimum difference of NNA scores between probe vs. intended-target and probe vs. unintended-target that eliminates the chance of cross-hybridization. Thus, we define the cross-hybridization NNA score threshold as $S_{nn}(probe) + \Delta E$. For example, if a probe and its intended target have a NNA score of -65.30, and ΔE is set to 30.0, then all NNA scores of probes and unintended targets smaller than -35.30 are considered to lead to cross-hybridization.

Given a probe p and its NNA scores $T = T_1, \ldots, T_n$ with unintended targets, and a cross-hybridization threshold $\mathrm{cht} = S_{nn}(p) + \Delta E$, the CHP chp of p is defined as

$$\mathrm{chp}(p) = \frac{\sum_{i=1}^{n} \begin{cases} \mathrm{cht} - T_i & \text{if } T_i < \mathrm{cht} \\ 0 & \text{else} \end{cases}}{\Delta E}. \qquad (5)$$

Of all NNA scores that indicate cross-hybridization, the amount they are below the threshold cht is summed up. By dividing the sum by ΔE, we can interpret the chp as the number of positions where cross-hybridization will occur under the following assumptions.

Sequences with a NNA score $\leq \mathrm{cht}$ are not considered to hybridize and there is a correlation of $\mathrm{cht} - T_i$ to the affinity of probe and sequence. All sequences present on the microarray appear in high copy number and are distributed evenly over the surface of the chip. We assume the hybridization process to be stochastic, thus probability of hybridization increases linearly in the amount by which the NNA score surpasses the cht and the number of sequences present.

2.1.3 Filtering Using Weighted Seeds

To compute the CHP of a probe, we need its NNA scores with unintended targets. The simplest but computationally most expensive way to obtain these scores, would be to align the probe to all non-target positions of the genome. For large genomes and high numbers

of probes, this becomes time consuming and one can observe that a large portion of the NNA scores is above the cross-hybridization threshold not contributing to the CHP. We introduce a filter for the number of scores to be computed a priori by only considering positions in the genome where the probability of obtaining a low NNA score is high.

Our filtering method is based on the observation that low-scoring alignments have thermodynamic stable contiguous matches, which fall below a certain score threshold. Therefore, we look for stable seeds between query and database and apply the NNA algorithm to only those positions. Other large scale database search and filtering algorithms like BLAST [27], [28], FASTA [29] and QUASAR [30], which rely on the q-gram Lemma [31], [32], search for short common factors between database and query. These methods identify exact matches with a minimum number of base pairs and extend the search from there. Similarly, we also employ a *seed and extent* approach, but use the free energy contribution of common factors to define seeds.

The filter exploits the correlation between the NNA score $\mathrm{nna}(p,t)$ of a probe p and a target t and the weight of the *heaviest common factor* of p and t.

Notation: We write $s \lhd t$ if s is a factor of t; the cases that s is empty or that $s = t$ are allowed.

A *common factor* of two strings p and t is a string s with both $s \lhd p$ and $s \lhd t$. A common factor is a *heaviest common factor* if no energetically more stable factor exists. We write

$$\mathrm{hcf}(p,t) := \min\{S_{nn}(s) : s \lhd p \text{ and } s \lhd t\} \qquad (6)$$

for the weight of the heaviest common factor.

Note the minimum in the definition; the weight is the sum of free energy contributions from stacked pairs (which are all negative), and the factor which can contribute most to the overall energy associated with the duplex formation of p and t is called the *heaviest common factor*. The heavier a common factor, the lower its score.

Using the *heaviest common factor* as an indicator for cross-hybridization is motivated by the following observations:

- duplex formation needs a sufficiently stable core to initiate binding [33],
- low scoring Nearest Neighbor Alignments usually have relatively heavy common factors, and
- depending on sequence composition, the heaviest common factor need not be the longest.

The filter is controlled by a seed threshold weight w. This w determines the minimum weight a seed must have and is defined as a fraction of the nearest neighbor score of a given probe. For example, if a probe has a nearest neighbor score of -59.20, then the maximum score for a seed will be expressed as $w \cdot (-59.2)$, in which w can be in the range of $0 \leq w \leq 1$. In general, small w will result in a greater number of seeds that will be considered, which in turn leads to more positions in the database that are verified. If the score of the *heaviest common factor* of a probe and a subsequence in the database is greater than the seed threshold, the NNA score for this probe and the database subsequence is computed.

To quickly find all occurrences of a given seed in a database, a q-gram index over the database is build. For a given probe, the filter then iterates over all minimal length factors which meet the seed threshold criterion. At all occurrences of every such seed in the database, NNA score of probe and the database subsequence at this position is computed.

2.2 Additional Probe Properties

The CHP as introduced in the previous section is our main measure for probe specificity. Several other probe properties are evaluated during candidate generation by OSProbes for comparison and to speed up the calculation of the candidate set by removing probes with flaws easily recognized such as runs of particular length [34].

2.2.1 Hamming Distance

The Hamming distance filter during probe candidate generation can be used to filter candidates with a Hamming distance to a non-target position in the database below a given threshold. To speed up the computation of this approximate string matching problem, we apply gapped q-gram filtering [35] or ungapped q-gram filtering using the traditional q-gram lemma [31]. The shape of the q-gram, as well as the q-gram threshold which specifies how many matching q-grams must exist to trigger verification, are part of the user input. This allows for a flexible filter where the user can choose between smaller, lossless shapes, and larger shapes which might have a sensitivity below one but can increase the speed dramatically. For example, consider the problem of finding a sequence of length 51 with at most 10 differences (a similarity of 80% or more). The shape 11011000101 (0s are irrelevant positions) used with a threshold of one will be lossless but has only a filtration ratio of about 0.012. On the other hand, the shape 11101110100101 1 with the same threshold has a filtration ratio of 0.00004 at the cost of missing some of the less similar hits; its sensitivity is 0.976.

When the Hamming distance filter is initialized, an index using the provided shape is built over the entire database. This lookup structure is then used to quickly find all exact q-gram matches for a given probe. A counter at every database position is used to check if the q-gram threshold is met and the Hamming distance needs to be computed.

2.2.2 Uniqueness and Longest Common Factor

The uniqueness filter can be used to limit the number of perfect or near-perfect matches a probe is allowed to have. A near-perfect match is defined as a match with a very low Hamming distance, i.e. we limit the

number of mismatching base pairs so that in the worst case there is still a common subsequence of length ≥ 25. As a result, probes with a length l of $50 \leq l < 75$ are allowed one and probes with $75 \leq l < 100$ up to two mismatching base pairs for a near-perfect match. When no unique probes can be found, non-unique probes can be valuable for identifying sequences with group testing [36], [37], or they can be used when designing a minimal size tiling array spanning different strains of a given genome [38]. In addition, we compute the length of the longest common factor of probe and non-target sequences.

Both of these filters make use of a contiguous q-gram index of the database and are thus very efficient.

2.2.3 Intrinsic Properties

We compute a number of properties based on probe sequence composition and allow filtering based on quantitative criteria [39]. These properties are the maximum content of any single base, the maximum length of any single base stretch and the GC content.

In addition, we compute palindromes, which are scores as the the maximum number of contiguous complementary base pairs formed by the probe folding back to itself. Low-sensitivity probes that fold back on themselves cannot bind their target and are therefore undesirable candidates.

The melting temperature T_m of the probe and its perfect Watson-Crick complementary target is computed using the nearest neighbor method [23].

2.3 Implementation

The NNA algorithm and all filters and indices used by OSProbes were implemented in ANSI C and compiled to Python modules using SWIG. Computing all of the probe properties is a time consuming exercise even for small genomes and it is therefore useful to order filters on the complexity of property computation. Local filters, which operate only on the probe sequence, are applied before the time consuming global filters, which assess probe sequence with respect to the genome. Whenever the properties of a probe do not lie within the user defined boundaries, it is discarded and the remaining properties do not have to be evaluated. For a complete order of the filters see Table 4.

3 Experiments

We designed a series of experiments to assess specificity and sensitivity of the Cross Hybridization Potential, the performance of the weighted seed filter, and how an OSProbes candidate set compares to sets generated by other software. Finally, we designed a tiling path using OSProbes and Tileomatic and compared it to other state of the art methods. For comparability, we performed the following experiments on the genome sequence of the fungus *Trichoderma reesei*, which was used by Lemoine et al. for a comparative study of custom microarray designs [40]. We compared our results for tiling array design to those obtained with OligoTiler [20] and ArrayDesign [21]. Both outperformed other methods and were analyzed extensively by Lemoine et al. whose procedure we follow.

We used the unmasked FASTA file of the *T. reesei* genome v.2.0 [41] from the U.S. Department of Energy website[1] with 33,454,791 base pairs in 87 sequences.

3.1 NNA Scores Versus Kane's Criteria

To evaluate the performance of the *Nearest Neighbor Alignment* (NNA) score and compare it to Kane's criteria, we generated oligonucleotide pairs with a low Hamming distance and used the program *hybridize*, part of the widely used *DINAMelt* package [42], [43], to obtain free energy values for possible hybridizations. DINAMelt simulates the melting of one or two single-stranded nucleic acids in solution. The entire equilibrium denaturation profiles as a function of temperature are calculated to obtain the melting temperature T_m for a given pair of strands. Stacked pairs, interior loops, bulges and dangling bases at the ends are taken into account for the computation of free energy of a duplex, and all possible conformational states are recursively tested. All free energy values obtained during our experiments were those of the duplex at a temperature of 37°C, Na^+ concentration of 1 M and Mg^{++} concentration of 0 M. We generated one million oligonucleotide pairs s, t using the following method: Pick a random 50mer of the genome sequence as $s = t$ and a random Hamming distance h between 10 and 15. Then change h bases in t either (1) at the 5′ end, (2) in the middle, (3) at the 3′ end, (4) maximally distributed, or (5) randomly distributed. For varying free energy thresholds for hybridization $H_{true}(s,t) = f \cdot S_{nn}(s)$ with $f \in [0,1]$, we then counted misclassified oligonucleotide pairs for each method. Is the free energy $E_{true}(s,t)$ which is computed by the *hybridize* program smaller than $H_{true}(s,t)$, s and t are assumed to hybridize. Depending on the result of the two predictors *Kane's criteria* and *NNA score* the oligonucleotide pair will then be counted as false/true positive/negative. Figure 2 plots the false positive rate versus true positive rate for all H_{true}.

The NNA score showed a larger true positive rate and a smaller false positive rate than Kane's criteria. Especially when used with $\Delta E = 25$, it was a much better predictor for cross-hybridization.

The running time for 1 million NNA score computations was 31.13 seconds. The optimized version of *hybridize* running with the parameter `--energy-only` and a maximum loop length of 30 bases needed more than 353 minutes to complete the task on the same 2.33 GHz CPU.

1. http://genome.jgi-psf.org/Trire2/Trire2.home.html

TABLE 4
Order of filters and their experimental filtration ratio and running time (for experiment details see Section 3.5).

Position	Type	Filter criterion	Experimental filtration ratio and running time	
1	local	sequence composition (GC content, single base run, single base contribution)	59.24%	3:02
2	local	palindromes	99.97%	5:42
3	local	T_m	-	-
4	global	number of perfect or near-perfect matches	96.27%	1:49:22
5	global	longest common factor	86.31%	1:47:08
6	global	Hamming distance	99.51%	1463:25:09
7	global	Cross Hybridization Potential	-	2808:27:53

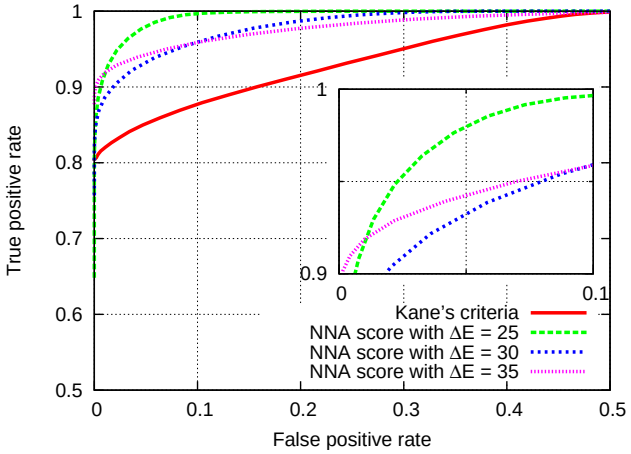

Fig. 2. Receiver operating characteristic (ROC) curves for cross-hybridization prediction by Kane's criteria and NNA score. Note the ranges of the axes, this is the upper left corner of the ROC plot.

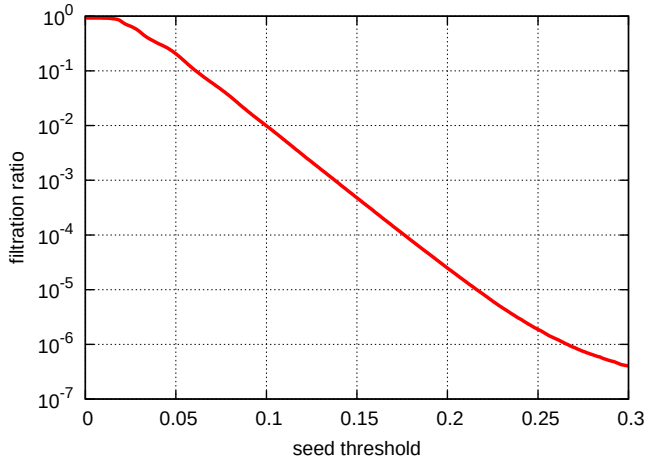

Fig. 3. Filtration ratio of the weighted seed filter for varying threshold weights w.

3.2 Weighted Seed Filter

The weighted seed filter (Section 2.1.3), which reduces the amount of NNA scores that have to be computed is controlled by the seed threshold weight w. With the following experiments we evaluated the filtration ratio as well as the sensitivity and specificity of the filter for varying w.

The filtration ratio of a filter is defined as the fraction of the number of downstream particles to the number of all upstream particles. In case of the weighted seed filter, we treat every position in the genome as an upstream particle, and every position where genome and a given oligonucleotide share a common factor of a weight greater than a certain threshold as a downstream particle. We randomly selected one billion 50mer pairs and computed the weight of the heaviest matching factor hmf, which differs from the heaviest common factor (Equation 6) that the start position in both oligonucleotides is equal. For all seed thresholds smaller than hmf, we added a downstream particle.

The filtration ratio experienced an exponential drop with increasing w (Fig. 3). At $w = 0.1$ one percent, and at $w \approx 0.14$ only 0.1% of the oligonucleotide pairs passed the filter.

Similarly, we measured sensitivity and specificity of the filter by randomly picking oligonucleotide pairs and computing their NNA score, as well as the weight of their heaviest common factor. We could then evaluate the ratio of oligonucleotide pairs which pass the filter and have a NNA score that indicates cross-hybridization, and all oligonucleotide pairs which pass the filter (specificity). In addition, we could determine the ratio of oligonucleotide pairs which cross-hybridize, and those which cross-hybridize and pass the filter (sensitivity). Another measure of sensitivity was given by the fraction of detected Cross Hybridization Potential (CHP) contribution.

The results for 240 million 50mer pairs of *T. reesei* with $\Delta E = 25$ show that the amount of detected CHP starts to fall for $w > 0.1$, where specificity is 0.65, and reaches 50% at $w = 0.15$ with a high specificity of 0.98 (Fig. 4). Seed thresholds of $w \approx 0.12$ result in a good trade-off by detecting almost 90% of the total CHP and filtering by a factor of 400, based on the previous experiment.

3.3 Candidate set for comparison

We generated probe candidates with length 60 bp. To receive a large maximal set with high coverage, we did

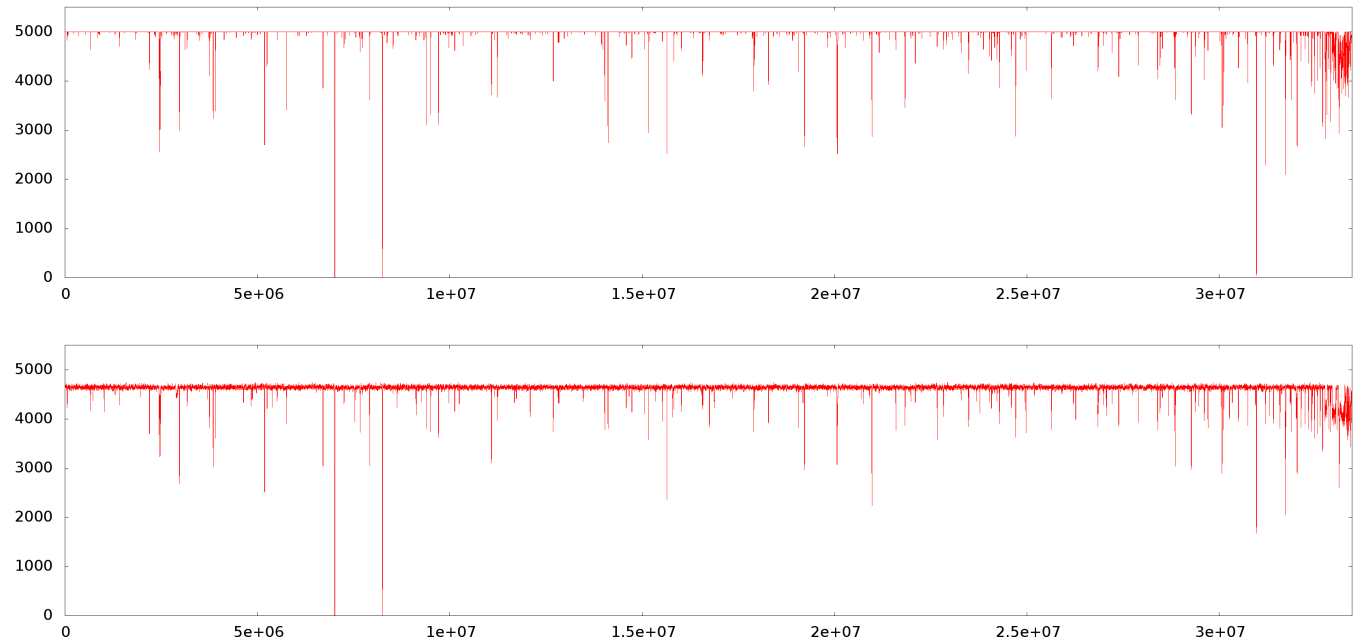

Fig. 5. Candidate probe density for the sets generated by OSProbes (top) and OligoTiler (bottom) for *T. reesei*. The graph shows the number of probes for all overlapping 5k windows. OligoTiler rejects more oligonucleotides than OSProbes but these oligonucleotides are distributed evenly across the genome. The number of large gaps ($>$ 500 bp) is comparable in both sets (OligoTiler 31, Tileomatic 34).

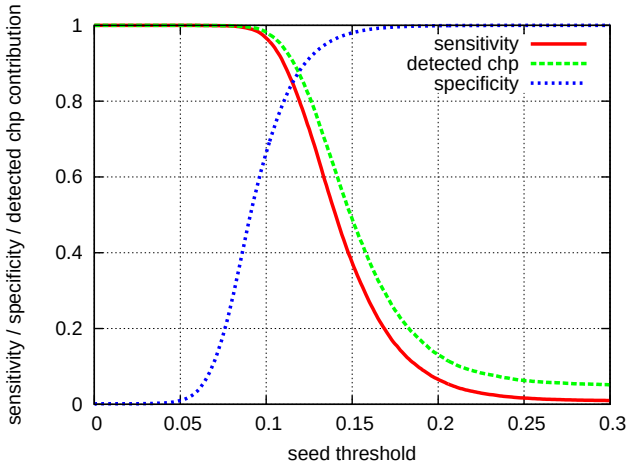

Fig. 4. Sensitivity and specificity of the weighted seed filter. When using $w = 0.13$, about 80% of the total CHP is detected, while only 10 % of the NNA scores do not contribute.

not make use of most filters. Our only requirement was a maximal uniqueness score of 1, i.e. every probe was allowed a maximum of one perfect or near-perfect match in the genome. As all oligonucleotide properties are part of the output of OSProbes, filtering can easily be done at any later time.

The gapped q-gram for the Hamming distance filter was 1110101100011011 with a threshold of 1. The CHP was computed with $\Delta E = 25$ for the NNA score thresholds, and $w = 0.13$ for determining the minimum weights of seeds.

In the absence of similar tools that can generate probe candidates, we employed OligoTiler to generate a 60mer tiling with oligonucleotide distance 1 and compared the two sets. OligoTiler was run via its web interface[2], leaving the advanced parameters at their default values (IR region = 5, IR require = 3 and repeat region overlap = 4).

Total running time of OSProbes was 9.5 CPU hours. The OligoTiler website returned a results after ca. 45 minutes. Of the 33,449,658 $60mers$ in *T. reesei* 0.1% included an unknown base and were not considered by both programs. From the total oligonucleotide set, 0.5% were filtered out by OSProbes because they were non-unique. OligoTiler rejected 7.8% of the oligonucleotides but created less gaps with a length greater than 500 bp (31 vs. 34), and the OSProbes set covered slightly less bases with at least one oligonucleotide than OligoTiler (99.8% vs. 99.9%). Thus, the oligonucleotides rejected by OligoTiler were distributed evenly across the genome (Fig. 5). In the OligoTiler set 0.4% of the candidates are non-unique and appear at multiple locations in the genome. Of the OSProbes set 7.5% of the candidates are not in the OligoTiler set; of those oligonucleotides 87% have a CHP of 0.0 indicating a strong specificity.

The OSProbes candidate set showed the same large gaps as OligoTiler, which were the result of repeats in the genome. But the overall larger candidate set could prove to be valuable during subsequent tiling path com-

2. http://tiling.gersteinlab.org/OligoTiler/oligotiler.cgi

putations, as we explored with the following experiment.

3.4 Tiling path

The candidate set generated by OSProbes holds all information about a number of probe properties. It is this information that can be used to guide the computation of an optimal tiling path with desired characteristics. Tileomatic [7] can handle this multi-criterion optimization problem gracefully by casting it into a shortest path problem. This allowed us to not only minimize the CHP of the tiling path probes, but also the variability in T_m and probe distances.

We computed tile paths for *T. reesei* using Tileomatic with a candidate set generated by OSProbes, OligoTiler, and ArrayDesign and compared the resulting oligonucleotide sets. OSProbes/Tileomatic and ArrayDesign but not OligoTiler support varying probe lengths. We used 60 bp for ease of comparison.

Tileomatic was used with the OSProbes candidate set described in the previous section and ran with the following target parameters: probe distance = 90 bp, T_m = 75°C, CHP = 0.0. The weights penalizing deviation from the target parameters were chosen as follows: 1, 1 and 10 for distance, T_m and CHP respectively.

OligoTiler was used with the same advanced parameters as in the previous section and the inter-oligonucleotide distance was set to 90.

ArrayDesign sources were obtained from the author's website[3]. As the software does not support the design of tiling arrays natively, we followed the steps described in [40] and created sequence windows of 150 bp at every 90 bp. For suffix array creation, the MAX_PREFIX_LENGTH variable was set to 15. ArrayDesign defines a uniqueness score u as a specificity measure for oligonucleotide probes. We generated two probe sets, one with the default value of $u = 0$ and one with $u = 15$ for high-specificity probes. In order to not discard any probes based on their melting temperature we set the temperature range to 20°–200°C. All remaining parameters were left at their default values.

The OligoTiler webservice returned the result ca. 6 minutes after uploading of the sequence file finished. Tileomatic took 21 minutes, and ArrayDesign needed 5 hours (u = 0) and 4.5 hours (u = 15).

OligoTiler picked the largest set (371k) with the lowest deviation in probe distance, but also showed the largest amount of probes with CHP $>$ 0.1 (35k, 9.4%). The Tileomatic set was the second largest (359k) and showed the lowest CHP; 99.99% had a value below or equal to 0.1. The two ArrayDesign sets showed a higher variance in probe distance and contained less probes than the other two sets (341k and 280k). For these sets, the number of probes with CHP $>$ 0.1 lay in between the other two methods; 19k for $u = 0$ and 11k for $u = 15$.

Probe count and CHP distribution for the different probe sets is summarized in Figure 7. Distribution of probe distances, and T_m and GC content are shown in Figure 6 and 8. Probes selected by Tileomatic show a significantly smaller variance in T_m; Bartlett's test of homogeneity of variances indicates the same variance in the other three sets (p-value 0.9638) but does not support this when comparing the Tileomatic set to the other sets (all p-values below 0.0005). Sensitivity measured as palindrome score is virtually the same for all four tiling paths and does not show any probes prone to self folding (data not shown).

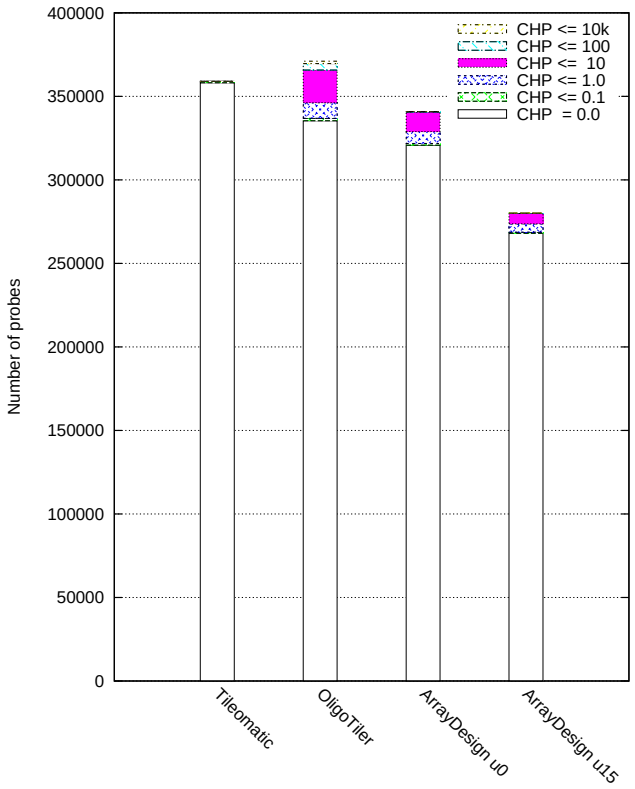

Fig. 7. Number of probes selected by each method and CHP distribution. OligoTiler picked the largest set, but Tileomatic the most probes with CHP 0.0.

Overall, the combination of OSProbes and Tileomatic resulted in the largest tiling path set with high-specificity probes. In addition, more than 90% of inter-probe distances were between 85–95 bp and close to 100% were between 70–110 bp. Furthermore, this probe set exhibits the smallest variance in T_m and GC content.

3.5 Candidate set for the human genome

We establish the feasibility to compute the CHP for suitable probes of the human genome. For in-depth studies, smaller region of interest such as the ENCODE region are selected for the design [44]. They are of the same size as the genome of *T.* reseii and can therefore be computed on modern desktop computers.

Nevertheless, we provide a set of probes for custom arrays for the complete human genome. We used the

3. http://www.ebi.ac.uk/~graef/arraydesign/

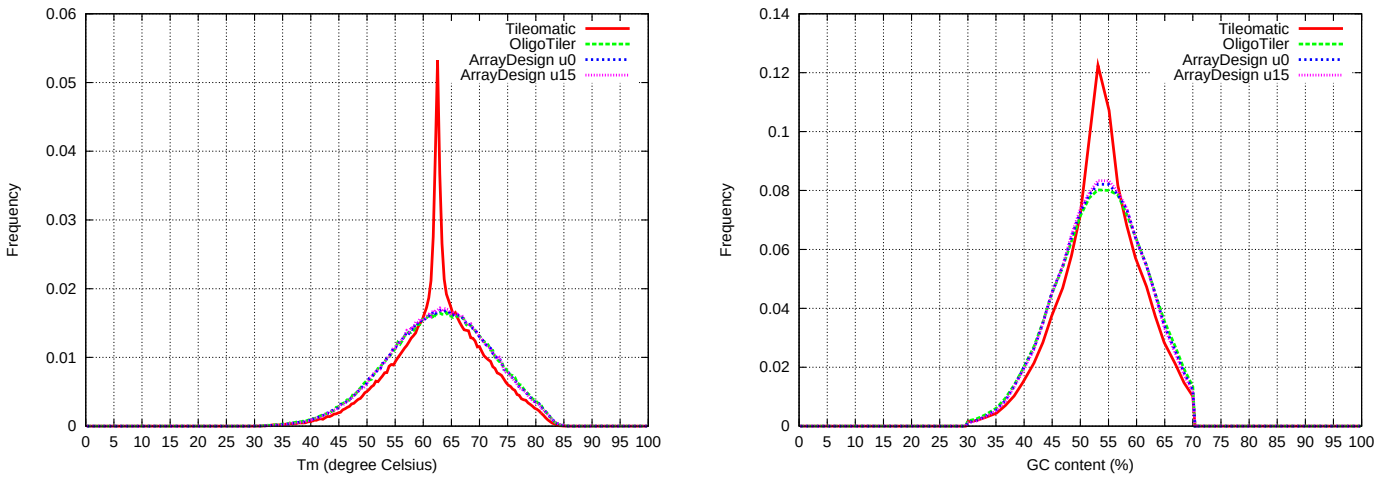

Fig. 6. Distribution of T_m and GC content of tiling paths. Tileomatic successfully minimized the variance in T_m by deviation from the probe distance target parameter.

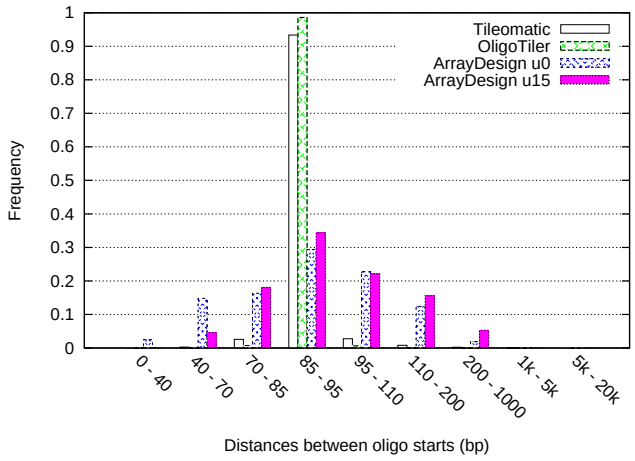

Fig. 8. Distribution of inter-probe distances. The OligoTiler and Tileomatic sets showed the smallest variation in distances between oligo starts.

repeat-masked version of Build GRCh37, downloaded from Ensembl [45]. For the computation of the CHP for probes of the human chromosome 1, the largest, we used a machine with 8 AMD Opteron 8439 SE and 256GB of memory. The total number of 60mers was 91,314,793. After use of the standard filters, 44,567,348 remained for computation of the CHP, which took 7 days utilizing about 157GB of the shared memory. Filter parameters are given in the supplemental material and filtration ratios and running times in Table 4. Candidate set computation took 23,597,628 CPU seconds of which ca. 66% is attributed to CHP computation, on average 0.23 CPU seconds per probe. This computational effort can be amortized over several design runs, as it is not necessary to rerun such analyses for a given build of the genome. The remaining chromosomes are underway. We are also investigating further algorithmic and implementational improvements to reduce running times.

4 Conclusion

While the design of oligonucleotide tiling arrays has received a lot of attention over the last years, methods still rely on simple criteria like Hamming distance to determine the susceptibility of probes to cross-hybridize. Cross-hybridization undermines the effectiveness of experiments as binding to unintended targets is detrimental to the signal-to-noise ratio and complicates downstream analysis.

We address the problem of computing sets of candidate oligonucleotide probes using thermo-dynamic considerations to predict cross-hybridization. The novel Cross Hybridization Potential (CHP) we define directly measures the quality of a given probe and has the advantage of combining local and global similarity of potential matches to the genome. Its computation is based on a Nearest Neighbor Alignment (NNA) which we derive from a simplified reformulation of t-gap insertion-deletion-like metrics. The NNA efficiently estimates a lower bound for the Gibbs free energy of the duplex formation of two DNA sequences. A novel filter using *weighted* ungapped q-grams to rapidly identify high-energy binding sites reduces the number of NNA scores that have to be computed by a factor of 400 while maintaining a sensitivity of 0.9. To reduce the computational effort the Nearest Neighbor Alignment is only computed for cases which cannot safely be decided by faster hamming-distance based heuristics, for which we use gapped q-grams. An additional suite of filters which are routinely used in the design of DNA microarrays is implement along the CHP in our software OSProbes, which computes sets of viable probe candidates.

Our experiments show that the CHP is a better predictor for cross-hybridization than Kane's criteria. Consequently, the number of oligonucleotide probes likely to cross-hybridize used in tiling arrays designed by other methods is 4.0 – 9.4%. This can be reduced to less than 0.01% by the use of OSProbes. The combi-

nation of OSProbes with our recently proposed linear-time algorithm which computes globally optimal tiling paths [7], implemented in the software Tileomatic, yields tiling arrays which are highly specific and balance inter-probe distances, striving for equal-distance probes, the probe quality with respect to cross-hybridization and the hybridization conditions, to assure that probes are for example as equi-thermal as possible. The computational costs of OSProbes can be amortized over multiple tiling arrays due to our two-step procedure. Both Tileomatic and OSProbes are available under the GPL and as a webservice at http://tileomatic.org.

Note that we found negligible differences between prior methods used for designing tiling arrays. In comparison to those prior methods we find that not only our probe qualities are significantly higher, but also that our arrays show a statistically significant lower variance ($p < 0.0005$) in melting temperatures, GC-content and other important design features, thus achieving improved signal-to-noise ratios and improved interpretability. The findings of several biological experiments will be reported elsewhere.

While the CHP was designed and evaluated in the context of tiling array design, it would be useful for other methods requiring large scale selection of probes, e.g. primer design and smaller scale oligonucleotide arrays such as those used for gene expression analysis.

ACKNOWLEDGMENT

The authors would like to thank Jörg Schreiber and Tino Polen for helpful discussions and experiments with tiling arrays designed with Tileomatic and OSProbes.

REFERENCES

[1] F. Li and G. D. Stormo, "Selection of optimal DNA oligos for gene expression arrays." *Bioinformatics*, vol. 17, no. 11, pp. 1067–1076, Nov 2001.

[2] T. J. Albert, M. N. Molla, D. M. Muzny, L. Nazareth, D. Wheeler, X. Song, T. A. Richmond, C. M. Middle, M. J. Rodesch, C. J. Packard, G. M. Weinstock, and R. A. Gibbs, "Direct selection of human genomic loci by microarray hybridization." *Nature methods*, vol. 4, pp. 903–5, 2007.

[3] R. Sasidharan, A. Agarwal, J. Rozowsky, and M. Gerstein, "An approach to compare genome tiling microarray and MPSS sequencing data for transcript mapping." *BMC Res Notes*, vol. 2, no. 1, p. 150, Jul 2009.

[4] W. Huber, A. von Heydebreck, H. Sültmann, A. Poustka, and M. Vingron, "Variance stabilization applied to microarray data calibration and to the quantification of differential expression," *Bioinformatics*, vol. 18 Suppl 1, pp. S96–104, Dec 2002.

[5] T. E. Royce, J. S. Rozowsky, and M. B. Gerstein, "Assessing the need for sequence-based normalization in tiling microarray experiments," *Bioinformatics*, vol. 23, no. 8, pp. 988–97, Apr 2007.

[6] H.-R. Chung, D. Kostka, and M. Vingron, "A physical model for tiling array analysis," *Bioinformatics*, vol. 23, no. 13, pp. i80–6, Jun 2007.

[7] A. Schliep and R. Krause, "Efficient algorithms for the computational design of optimal tiling arrays," *IEEE/ACM Trans Comput Biol Bioinform*, vol. 5, no. 4, pp. 557–567, Oct-Dec 2008.

[8] A. G. D'yachkov, A. J. Macula, W. K. Pogozelski, T. E. Renz, V. V. Rykov, and D. C. Torney, "New t-gap insertion-deletion-like metrics for DNA hybridization thermodynamic modeling." *J Comput Biol*, vol. 13, no. 4, pp. 866–881, May 2006.

[9] M. D. Kane, T. A. Jatkoe, C. R. Stumpf, J. Lu, J. D. Thomas, and S. J. Madore, "Assessment of the sensitivity and specificity of oligonucleotide (50mer) microarrays," *Nucl. Acids Res.*, vol. 28, no. 22, pp. 4552–4557, 2000.

[10] N. Reymond, H. Charles, L. Duret, F. Calevro, G. Beslon, and J.-M. Fayard, "ROSO: Optimizing oligonucleotide probes for microarrays." *Bioinformatics*, vol. 20, no. 2, pp. 271–273, Jan 2004.

[11] S. Rimour, D. Hill, C. Militon, and P. Peyret, "GoArrays: Highly dynamic and efficient microarray probe design." *Bioinformatics*, vol. 21, no. 7, pp. 1094–1103, Apr 2005.

[12] X. Wang and B. Seed, "Selection of oligonucleotide probes for protein coding sequences." *Bioinformatics*, vol. 19, no. 7, pp. 796–802, May 2003.

[13] R. Wernersson and H. B. Nielsen, "OligoWiz 2.0–integrating sequence feature annotation into the design of microarray probes." *Nucleic Acids Res*, vol. 33, no. Web Server issue, pp. W611–W615, Jul 2005.

[14] H. Chen and B. M. Sharp, "Oliz, a suite of Perl scripts that assist in the design of microarrays using 50mer oligonucleotides from the 3' untranslated region." *BMC Bioinformatics*, vol. 3, p. 27, Oct 2002.

[15] L. Kaderali and A. Schliep, "Selecting signature oligonucleotides to identify organisms using DNA arrays." *Bioinformatics*, vol. 18, no. 10, pp. 1340–1349, Oct 2002.

[16] S. Rahmann, "Fast large scale oligonucleotide selection using the longest common factor approach," *Journal of Bioinformatics and Computational Biology*, vol. 1, no. 2, pp. 343–361, July 2003.

[17] J.-M. Rouillard, C. J. Herbert, and M. Zuker, "OligoArray: Genome-scale oligonucleotide design for microarrays." *Bioinformatics*, vol. 18, no. 3, pp. 486–487, Mar 2002.

[18] J.-M. Rouillard, M. Zuker, and E. Gulari, "OligoArray 2.0: Design of oligonucleotide probes for DNA microarrays using a thermodynamic approach." *Nucleic Acids Res*, vol. 31, no. 12, pp. 3057–3062, Jun 2003.

[19] J. D. Gans and M. Wolinsky, "Improved assay-dependent searching of nucleic acid sequence databases." *Nucleic acids research*, vol. 36, no. 12, Jul 2008.

[20] P. Bertone, V. Trifonov, J. S. Rozowsky, F. Schubert, O. Emanuelsson, J. Karro, M. Y. Kao, M. Snyder, and M. Gerstein, "Design optimization methods for genomic DNA tiling arrays," *Genome Res*, vol. 16, no. 2, pp. 271–281, Feb 2006.

[21] S. Gräf, F. G. G. Nielsen, S. Kurtz, M. A. Huynen, E. Birney, H. Stunnenberg, and P. Flicek, "Optimized design and assessment of whole genome tiling arrays." *Bioinformatics*, vol. 23, no. 13, pp. i195–i204, Jul 2007.

[22] G. O. S. Thomassen, A. D. Rowe, K. Lagesen, J. M. Lindvall, and T. Rognes, "Custom design and analysis of high-density oligonucleotide bacterial tiling microarrays." *PLoS One*, vol. 4, no. 6, p. e5943, 2009.

[23] J. SantaLucia, "A unified view of polymer, dumbbell, and oligonucleotide DNA nearest-neighbor thermodynamics." *Proc Natl Acad Sci U S A*, vol. 95, no. 4, pp. 1460–1465, Feb 1998.

[24] A. E. Pozhitkov and D. Tautz, "An algorithm and program for finding sequence specific oligonucleotide probes for species identification." *BMC Bioinformatics*, vol. 3, p. 9, 2002.

[25] L. Zhang, C. Wu, R. Carta, and H. Zhao, "Free energy of DNA duplex formation on short oligonucleotide microarrays." *Nucleic Acids Res*, vol. 35, no. 3, p. e18, 2007.

[26] M. Seringhaus, J. Rozowsky, T. Royce, U. Nagalakshmi, J. Jee, M. Snyder, and M. Gerstein, "Mismatch oligonucleotides in human and yeast: Guidelines for probe design on tiling microarrays." *BMC Genomics*, vol. 9, p. 635, 2008.

[27] S. F. Altschul, W. Gish, W. Miller, E. W. Myers, and D. J. Lipman, "Basic local alignment search tool." *J Mol Biol*, vol. 215, no. 3, pp. 403–410, Oct 1990.

[28] S. F. Altschul, T. L. Madden, A. A. Schäffer, J. Zhang, Z. Zhang, W. Miller, and D. J. Lipman, "Gapped BLAST and PSI-BLAST: A new generation of protein database search programs." *Nucleic Acids Res*, vol. 25, no. 17, pp. 3389–3402, Sep 1997.

[29] W. R. Pearson, "Rapid and sensitive sequence comparison with FASTP and FASTA." *Methods Enzymol*, vol. 183, pp. 63–98, 1990.

[30] S. Burkhardt, A. Crauser, P. Ferragina, H.-P. Lenhof, E. Rivals, and M. Vingron, "Q-gram based database searching using a suffix array (QUASAR)," in *RECOMB*, 1999, pp. 77–83.

[31] P. Jokinen and E. Ukkonen, "Two algorithms for approximate string matching in static texts." *Proc. of the 16th Symposium on*

Mathematical Foundations of Computer Science, vol. 520, pp. 240–248, 1991.
[32] E. Ukkonen, "Approximate string-matching with q-grams and maximal matches," *Theor. Comput. Sci.*, vol. 92, no. 1, pp. 191–211, 1992.
[33] E.Southern, K. Mir, and M. Shchepinov, "Molecular interactions on microarrays," *Nature Genetics*, vol. 21, pp. 5–9, 1999.
[34] W. B. Langdon, G. J. Upton, and A. P. Harrison, "Probes containing runs of guanines provide insights into the biophysics and bioinformatics of Affymetrix GeneChips," *Brief Bioinform*, vol. 10, no. 3, pp. 259–277, May 2009.
[35] S. Burkhardt and J. Kärkkäinen, "Better filtering with gapped q-grams," *Fundam. Inf.*, vol. 56, no. 1,2, pp. 51–70, 2002.
[36] A. Schliep, D. C. Torney, and S. Rahmann, "Group testing with DNA chips: Generating designs and decoding experiments," in *Proceedings of the 2nd IEEE Computer Society Bioinformatics Conference (CSB 2003)*. IEEE, 2003, pp. 84–93.
[37] G. W. Klau, S. Rahmann, A. Schliep, M. Vingron, and K. Reinert, "Optimal robust non-unique probe selection using integer linear programming." *Bioinformatics*, vol. 20 Suppl 1, pp. i186–i193, Aug 2004.
[38] A. Phillippy, X. Deng, W. Zhang, and S. Salzberg, "Efficient oligonucleotide probe selection for pan-genomic tiling arrays."
[39] D. J. Lockhart, H. Dong, M. C. Byrne, M. T. Follettie, M. V. Gallo, M. S. Chee, M. Mittmann, C. Wang, M. Kobayashi, H. Horton, and E. L. Brown, "Expression monitoring by hybridization to high-density oligonucleotide arrays." *Nat Biotechnol*, vol. 14, no. 13, pp. 1675–1680, Dec 1996.
[40] S. Lemoine, F. Combes, and S. L. Crom, "An evaluation of custom microarray applications: The oligonucleotide design challenge." *Nucleic Acids Res*, vol. 37, no. 6, pp. 1726–1739, Apr 2009.
[41] D. Martinez, R. M. Berka, B. Henrissat, M. Saloheimo, M. Arvas, S. E. Baker, J. Chapman, O. Chertkov, P. M. Coutinho, D. Cullen, E. G. J. Danchin, I. V. Grigoriev, P. Harris, M. Jackson, C. P. Kubicek, C. S. Han, I. Ho, L. F. Larrondo, A. L. de Leon, J. K. Magnuson, S. Merino, M. Misra, B. Nelson, N. Putnam, B. Robbertse, A. A. Salamov, M. Schmoll, A. Terry, N. Thayer, A. Westerholm-Parvinen, C. L. Schoch, J. Yao, R. Barabote, R. Barbote, M. A. Nelson, C. Detter, D. Bruce, C. R. Kuske, G. Xie, P. Richardson, D. S. Rokhsar, S. M. Lucas, E. M. Rubin, N. Dunn-Coleman, M. Ward, and T. S. Brettin, "Genome sequencing and analysis of the biomass-degrading fungus Trichoderma reesei (syn. Hypocrea jecorina)." *Nat Biotechnol*, vol. 26, no. 5, pp. 553–560, May 2008.
[42] N. R. Markham and M. Zuker, "DINAMelt web server for nucleic acid melting prediction." *Nucleic Acids Res*, vol. 33, no. Web Server issue, pp. W577–W581, Jul 2005.
[43] R. A. Dimitrov and M. Zuker, "Prediction of hybridization and melting for double-stranded nucleic acids." *Biophys J*, vol. 87, no. 1, pp. 215–226, Jul 2004.
[44] E. Birney, J. Stamatoyannopoulos, A. Dutta, R. Guigó, T. Gingeras, E. Margulies, Z. Weng, M. Snyder, E. Dermitzakis, R. Thurman *et al.*, "Identification and analysis of functional elements in 1% of the human genome by the ENCODE pilot project," *Nature*, vol. 447, no. 7146, pp. 799–816, 2007.
[45] T. J. P. Hubbard, B. L. Aken, S. Ayling, B. Ballester, K. Beal, E. Bragin, S. Brent, Y. Chen, P. Clapham, L. Clarke, G. Coates, S. Fairley, S. Fitzgerald, J. Fernandez-Banet, L. Gordon, S. Graf, S. Haider, M. Hammond, R. Holland, K. Howe, A. Jenkinson, N. Johnson, A. Kahari, D. Keefe, S. Keenan, R. Kinsella, F. Kokocinski, E. Kulesha, D. Lawson, I. Longden, K. Megy, P. Meidl, B. Overduin, A. Parker, B. Pritchard, D. Rios, M. Schuster, G. Slater, D. Smedley, W. Spooner, G. Spudich, S. Trevanion, A. Vilella, J. Vogel, S. White, S. Wilder, A. Zadissa, E. Birney, F. Cunningham, V. Curwen, R. Durbin, X. M. Fernandez-Suarez, J. Herrero, A. Kasprzyk, G. Proctor, J. Smith, S. Searle, and P. Flicek, "Ensembl 2009," *Nucleic Acids Research*, vol. 37, no. suppl 1, pp. D690–D697, 2009.

Christoph Hafemeister received his M.Sc. in bioinformatics from Freie Universität Berlin in 2008. From 2005 to 2009, he was a research assistant at the algorithms group at the Max Planck Institute for Molecular Genetics, Department for Computational Molecular Biology in Berlin. In 2010 he joined the New York University Doctoral Program in Computational Biology. His research interests include sequence analysis and machine learning methods such as clustering using mixture models and semi-supervised learning, and the application of these models to elucidate gene regulation and developmental processes.

Roland Krause is a scientist and lecturer at the Free University of Berlin and the Max Planck Institute for Molecular Genetics, Berlin. His research focus is on the evolution of the gene regulation machinery. His undergraduate degree in biotechnology was awarded from the Mannheim University for Applied Sciences in 1999 and his doctorate in biochemistry from the University of Heidelberg in 2004.

Alexander Schliep received his Ph.D. in computer science from the Center for Applied Computer Science (ZAIK) at the University of Cologne, Germany, in 2001, working in collaboration with the Theoretical Biology and Biophysics group (T-10) at Los Alamos National Laboratory, NM. From 2002 to 2009 he held a group leader position in the Department for Computational Molecular Biology at the Max Planck Institute for Molecular Genetics, Berlin. In summer 2009 he joined Rutgers University, as an associate professor in computer science and the BioMaPS institute for quantitative biology. The research interests pursued in his group include data mining, statistical models and algorithms for analyzing complex, heterogeneous data from molecular biology.