Construction of Physical Maps From Oligonucleotide Fingerprints Data*

Guy Mayraz † Ron Shamir ‡

July 1998, revised March 1999.

Abstract

A new algorithm for the construction of physical maps from hybridization fingerprints of short oligonucleotide probes has been developed. Extensive simulations in high-noise scenarios show that the algorithm produces an essentially completely correct map in over 95% of trials. Tests for the influence of specific experimental parameters demonstrate that the algorithm is robust to both false positive and false negative experimental errors. The algorithm was also tested in simulations using real DNA sequences of *E. coli, B. subtilis, M. tuberculosis, S. cerevisiae, C. elegans, and H. sapiens.* To overcome the non-randomness of probe frequencies in these sequences, probes were preselected based on sequence statistics and a screening process of the hybridization data was developed. With these modifications, the algorithm produced very encouraging results.

*A preliminary version of the paper is to appear in Proc. RECOMB 99.

†Department of Computer Science, Sackler Faculty of Exact Sciences, Tel-Aviv University, Tel-Aviv 69978 ISRAEL. Current address: Gatsby Computational Neuroscience Unit, University College London, 17 Queen Square, London WC1N 3AR, U.K. gmz@gatsby.ucl.ac.uk URL: http://www.gatsby.ucl.ac.uk/~gmz.

‡To whom correspondence should be addressed. Department of Computer Science, Sackler Faculty of Exact Sciences, Tel-Aviv University, Tel-Aviv 69978 ISRAEL. shamir@math.tau.ac.il URL: http://www.math.tau.ac.il/~shamir. Parts of this work were performed while this author was on sabbatical at the Department of Computer Science and Engineering, University of Washington, Seattle.

1 Introduction

Physical mapping is the process of determining the relative position of landmarks along a genome section. The resulting maps are used as a basis for DNA sequencing, and for the isolation and characterization of individual genes or other DNA regions of interest (e.g., transcripts and regulatory sequences)[12]. The construction of a high-resolution sequence-ready physical maps for human and other organisms is still one of the top priorities of the Human Genome Project. The mapping process involves the production of *clones*, i.e., cloned pieces of DNA, each representing a chromosomal segment. The mapping itself is the reconstruction of the order of and physical distance between the clones. Refer to Figure 1 for an illustration of the physical mapping problem, and for definitions of some basic terminology.

Physical mapping using hybridization fingerprints of short oligonucleotides was first suggested by Poustka et al. [14] in 1986. In this technique short labeled DNA sequences, or *probes*, attach, or *hybridize*, to positions along the target DNA matching their own DNA sequence. The probes are non-unique, i.e., they occur at many points along the genome, and typically hybridize with 10%-50% of the clones. Overlapping clones can be identified by their similar fingerprints. Refer to Figure 2 for an illustration of this hybridization scenario. Poustka et al. suggested this method in order to eliminate the need to process individual clones in the restriction digestion technique. They reported on preliminary computer simulations demonstrating feasibility, and suggested the use of Bayesian statistics in data analysis. More detailed strategies were offered by Michiels et al. [11]. A likelihood ratio based on a detailed statistical model was used to make overlap decisions, and a discussion of experimental errors was also included. Craig et al. [3] used short oligonucleotides in the ordering of cosmid clones covering the Herpes Simplex Virus (HSV-I) genome. The clones were ordered manually. As each probe occurred only once or twice along the short (140K base pairs) genome, this experiment is not representative of the general problem. Lehrach et al. [9] described an integrated approach to physical mapping, suggesting the advantages of using non-unique probes. Fu et al. [5] provided an analysis of the number of clones required to avoid gaps and the number of probes needed with a simple construction method. Hoheisel et al. [7] used short probes as a check for the order of unique probes in a map of the fission yeast *S. pombe*. Cuticchia et al. [4] provided an ordering program, using the Hamming distance between consecutive clones as an objective function for simulated annealing. Its performance was demonstrated on the data of the HSV-I virus[3] (see note above). The problem received a detailed formal

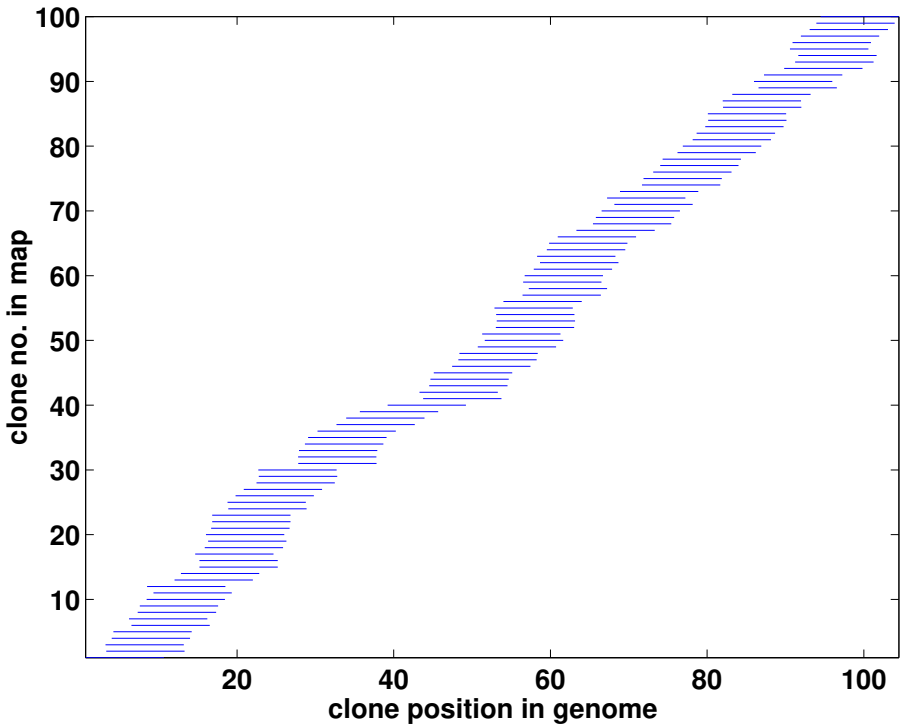

Figure 1: An example of a small physical map. The short horizontal lines represent clones. The x-axis represents position along the target genome, and the y-axis represents the clone order in the constructed map. The *clone coverage* is the total length of the clones divided by the length of the genome. The position of the clones along the target genome is lost during the cloning process, and experimental data (such as hybridization fingerprints) is later used to reconstruct the *physical map*, i.e., the position of the clones on the target genome. A *contig* is a list of clones covering a continuous section of the genome together with their physical distances. It is not always possible to produce a one contig physical map, because of unclonable regions (known as *oceans*), or low clone coverage. In a plot such as the one reproduced here, an essentially correct solution corresponds to a monotone relationship between clone order and position on the target genome. Small errors can be seen as small deviations from this monotonicity, whereas a solution containing large errors would translate into several randomly placed broken contigs with an approximately correct intra-contig order (compare Figure 3(d)). Some errors are inevitable when there are gaps (oceans), which are not covered by any of the clones in the library. Our algorithm constructs a proposed map and also pinpoints suspect links in the suggested map, indicating such unclonable regions, or specific locations which could benefit from further experiments. In Section 5.2 we define a quantitative map quality characterization.

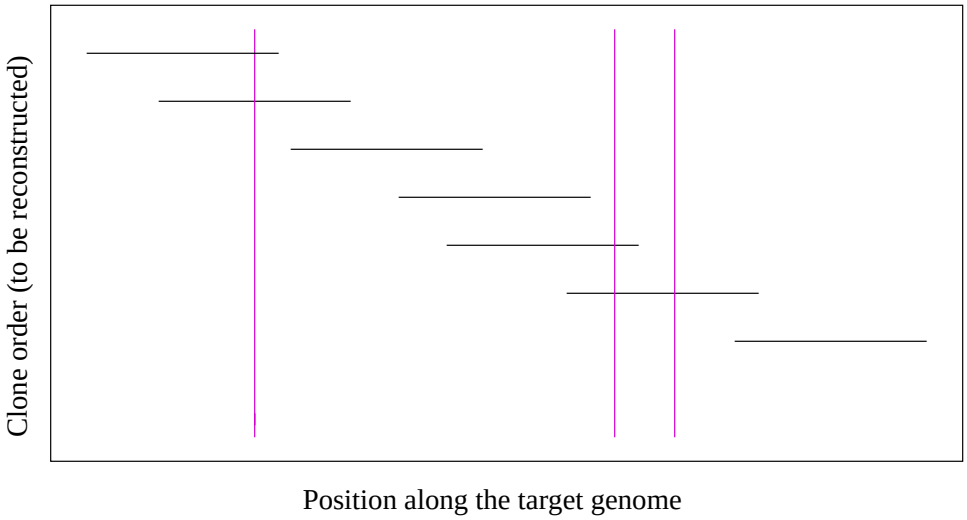

Figure 2: Clones and non-unique probes. The clones are the horizontal lines. The random occurrences of a single non-unique probe are marked by the vertical lines. We denote by A the clone-probe *occurrence matrix*: $A_{i,j} = k$ if probe j occurs k times in clone i. The corresponding column for the probe in this example would be $(1, 1, 0, 0, 1, 2, 0)$. We denote by B the clone-probe *hybridization matrix*: $B_{i,j} = 1$ or $B_{i,j} = 0$ depending on whether probe j hybridized with clone i or not. In case no noise is present hybridization occurs if and only if there is at least one occurrence of the probe. In this case the appropriate column of B would be $(1, 1, 0, 0, 1, 1, 0)$. The vector $\vec{B_i}$ of the hybridizations of clone i with all the probes is also called its *hybridization fingerprint*. The hybridization fingerprints of intersecting clones are correlated, and this fact is used in order to estimate the clone pairs overlap. Experimental noise can result in both false positive hybridizations ($B_{i,j} = 1$ when $A_{i,j} = 0$), and false negative hybridizations ($B_{i,j} = 0$ when $A_{i,j} > 0$). Nevertheless, Bayesian statistics can still be used to identify overlapping clones, provided a sufficient number of probes is used. In some experimental procedures quantitative hybridization signals are available, and these provide more information on probe occurrences than binary signals do. In this work we only rely only on binary hybridization signals.

analysis by Alizadeh et al. [1]. Local improvement methods were used to solve the problem in a noise free scenario.

In comparison with the restriction digestion method, both unique and non-unique probes hybridization techniques have much simpler handling requirements, and many clones can be analyzed in parallel. In contrast with unique probes hybridization, using non-unique oligonucleotide probes has the following advantages: (1) Probe generation is straightforward and much cheaper, (2) The number of required probes is independent of the genome size[7], (3) It is less sensitive to repeat sequences, and (4) The information theoretic content of each experiment is much higher: The overwhelming majority of unique probe hybridizations to clones in a map are negative. (RH mapping [2] is an alternative recent approach that set out to overcome the latter drawback.) This makes possible a very natural and efficient probabilistic interpretation of the hybridization data, resulting in strong robustness to experimental errors.

A disadvantage of the non-unique probes hybridization technique is that the computational analysis of the hybridization data is relatively complicated. The aim of this paper is to suggest an efficient practical algorithm. In Section 2 we suggest a statistical model for the problem. Perhaps the single most important idea in the paper is the Bayesian analysis of the overlap length between contigs, which is based on the analysis of the overlap length between two clones. Both are presented in Section 3. These overlap scores enable us to make much more efficient use of the hybridization data than was previously possible with a binary yes/no overlap decision [14, 1, 11]. In Section 4 we present the actual construction algorithm. Section 5 includes the results of simulations in a variety of experimental scenarios, including results on real DNA sequences. To the best of our knowledge, this study gives the first quantitative performance analysis in noisy experimental scenarios. We were therefore unable to include direct comparison with other papers. In the discussion section we take a close look at the inevitable simplifications made for the purpose of the general simulations, and discuss what remains to be done for a successful performance on real experimental data. An appendix gives the mathematical details of the overlap score.

Some additional information on this project is available in a web site at http://www.math.tau.ac.il/~gmz/physical_mapping. Related software is available through the authors.

2 The statistical model

We first describe the statistical model that underlies our algorithm and our simulations. In the basic simulations we make the following assumptions: (1) Clones are uniformly and independently distributed along the target genome, (2) Clones are of equal length, (3) Probe occurrences along the genome are modeled by a Poisson process with the same rate for all probes. The rate λ is related to the length of the probes *plen* by the formula $\lambda = 1/4^{plen}$. Probe occurrences on clones are then determined by the clone position. These assumptions originate in [11], where they are also discussed in depth. The same assumptions were later adopted by other researchers [1, 13]. Assumptions (1) and (2) were also used in [8] in a general analysis of mapping scenarios, and were shown to produce clone distributions that mimic quite accurately real genomic clones.

Noise is introduced into the model via false positive and false negative errors. False positives are modeled by an independent Poisson process with the same rate on each clone. Each probe occurrence results in a hybridization with probability $1-\alpha$ independently of other occurrences of the same probe. The parameter α is a measure of false negatives errors.

We believe the first assumption and the noise modeling are in good agreement with real experiments. The second assumption is a reasonable approximation for cosmid clones, which are the most likely clones to be used in physical mapping with oligonucleotides. There are nevertheless other interesting scenarios where larger clone length variability occurs. In Section 5.5 we examine the effect of variable clone lengths. The third assumption is the most problematic. In Section 6 we focus on mapping simulated data based on real genomic sequences of several organisms. We show that while the Poisson approximation appears to be valid, the assumption that all probes appear with the same Poisson rate does not hold for randomly chosen probes. We present a solution to this problem based on judicious preselection of probes. With this improvement, our mapping algorithm obtains very good results on simulations with real sequence data.

Like any probabilistic model, ours is only a model, and as such does not fully reflect reality. In spite of the above caveats these assumptions have the advantage of simplicity allowing both analytical and simulation aided investigation. The insights gained from such an investigation shed invaluable light on the more complicated experimental conditions of a real laboratory.

3 The Bayesian overlap score

Contigs are ordered, contiguous clone sets. The aim of physical mapping is to create one correct contig out of all the input clones[1]. We begin our analysis with an outline of a Bayesian score for the overlap length between clone pairs (Section 3.1), and then describe how we use it to estimate the overlap between two contigs (Section 3.2). The construction algorithm which uses these overlap scores to build the map is described in Section 4. A Bayesian overlap score for clones was suggested in several previous papers [1, 11, 14]. Such a score lends itself to local improvement methods based on the clones permutation. In these studies the score was only used to determine if two clones do or do not overlap. In contrast, we compute a more detailed overlap length estimate and contigs score, which make a better use of the fingerprint information. An alternative approach suggested in [1] starts with a formal discussion of a global Bayesian score for complete maps, and proceeds with various ways of approximating it.

3.1 Clone pairs overlap score

The next two subsections give the outline and key ideas of the overlap scores. A full description is given in Appendix A. Consider a pair of clones with an overlap section of length l_γ. With a small l_γ it is quite likely for a probe to hybridize with one clone, but not with the other. With a larger l_γ the probe is more likely to hybridize with both clones, or with neither. This observation can be made precise, giving for every possible value of l_γ its likelihood given the data. Combined with a prior on the overlap probabilities, Bayes Theorem makes it possible to calculate the probability that two clones overlap by a certain length given their fingerprint data.

Because of efficiency considerations, we make the simplifying assumption that l_γ can only take few quantized values, in steps of $1/d$ of a clone's length. With the aid of simple tables the complexity of calculating the overlap probabilities vector for a pair of clones is $O(n+d)$, where n is the number of probes. Typically, $d \ll n$ ($d = 10$ was used in our simulations), and so the total complexity is linear in the number of probes.

We finally note that in the calculation of the overlap length likelihood we assume that only a yes/no hybridization signal is available to the construction algorithm. Finer hybridization intensity distinction can be used

[1] Because of unclonable gaps the best solution sometimes includes more than one contig. We addressed this problem by an algorithm, which finds suspected links in the final solution (Section 4.2.3).

to provide improved overlap probabilities estimates.

3.2 Contig pairs overlap score

Extending the exact overlap probability calculation to more than a few clones is impractical due to the exponential growth in complexity as a function of the number of clones. We seek instead to use the information from clone pairs to find the relative placement between two contigs. This is done by taking all clone pairs from the two contigs (one clone from each contig) as independent sources of information on the contigs' relative placement. This approach runs the risk of counting the same evidence multiple times resulting in over sharp estimates. This problem is not severe, since the algorithm depends on a correct identification of the maximum likelihood overlap, rather than on the precise shape of the distribution of likely overlap lengths. In practice, the approximation was found to produce good results.

4 The construction algorithm

Section 4.1 describes the basic algorithm. Several changes described in Section 4.2 were found to improve performance.

4.1 The basic algorithm

1. Initialize a contig set, so that for each clone there is a corresponding contig consisting of that clone.

2. Calculate for all the initial contig pairs and for the two possible relative orientations their relative placement probabilities vector. The results are stored in a table.

3. Find for all contig pairs and for their two possible relative orientations their best relative placement, and its probability

4. While more than one contig remains:

   (a) Find two contigs which have relative orientation and placement that attain the highest probability over all contig pairs.

   (b) Merge the two contigs.

   (c) Update steps 2 and 3 above to reflect the last merger. If contig b was merged into contig a, then the entries of contig b are erased,

and those of all pairs (a, x) are updated. The required change is a simple combination of the previous entries for (a, x) and (b, x).

Assume that there are n probes and m clones of normalized length d, and the genome has length Ld. The bottleneck steps in the computation are steps 2 that takes $O(m^2(n + Ld))$ time and Steps 4(c) and 4(d) that take $O(mLd)$ time. Step 4 is repeated $m-1$ times, giving a total time complexity of $O(m^2(m + Ld + n))$.

4.2 Improvements to the basic algorithm

We briefly describe below several ideas that were found to improve the performance of the basic algorithm. These ideas were incorporated into the final version of the algorithm used in the simulations.

4.2.1 Initial partition into groups

The clone pairs score is used to divide the initial clone set into groups of clones, such that in each group there is a clone that overlaps all the other clones in its group with a near certain probability. Construction is done at the group level before the ordered groups are combined. This process makes for a significant improvement in the average running time. More importantly, the weighted information of many clone pairs is available when the ordered groups are combined, thus helping avoid wrong mergers of distant clones with similar fingerprints.

4.2.2 Choosing the best overlapping relative placement

In every step of the basic algorithm the contig pair to be merged is the one with the highest posterior relative placement probability, thus minimizing the probability of making an error. In practice, however, errors are unavoidable, and some are far more damaging than others. We therefore seek to avoid wrong mergers of non-overlapping contigs and other particularly large errors. One idea was to first identify the contig pairs that have the lowest probability of no overlap, and among those to pick the pair that has the highest relative overlap probability. Another idea was to apply a Gaussian filter on the probabilities vector, with the result that unimodal distributions are preferred over multi-peaked distributions, thereby decreasing the probability of large errors. Both processes decrease the probability of large errors at the cost of increasing that of smaller ones.

4.2.3 Weak points detection

When construction is completed only one contig remains. In reality, however, there are often gaps (oceans) in the stretch of the genome covered by the clones, and the sub-contigs between oceans should not be linked together. Moreover, errors in the construction process can also result in incorrectly linked sub-contigs. It is therefore highly desirable to be able in retrospect to pinpoint suspected points of incorrect merger. One promising idea is that the overlap score between pairs of clones on opposite sides of a suspected point is highly indicative of its reliability.

For any clone pair (C_i, C_j) with an overlap $o_{i,j}$ and hybridization fingerprints $(\vec{B_i}, \vec{B_j})$ define its clone pair "energy" as $E_{i,j} = -\log Pr(\vec{B_i}, \vec{B_j}|o_{i,j})$. The contig energy E is defined as $E = \sum_{i<j} E_{i,j}$. In analogy to the breaking-up of a chemical molecule, the separation of a contig into two non-overlapping parts should increase the energy substantially. However, if the two parts do not overlap in the real map, the separation energy should be quite small or even negative. We define *weak points* as points along the contig having separation energy below a threshold (determined experimentally).

When the number of errors is fairly small, a good choice of threshold should make it possible to mark all errors and oceans as weak points with only a small number of false positive detections. The information on weak points can also be used by the laboratory to pinpoint areas where additional hybridization experiments are likely to be helpful. We also make use of the weak points in our algorithm to break-up a contig and to reassemble it again. In case an error was made at an early stage, this process enables the algorithm to correct its previous error with the benefit of the additional information from other clones added at a later stage. A similar process was reported in [13] to be slightly favorable, when only the clone order was sought.

4.2.4 Simulated annealing

Simulated annealing on the clones order was used before [4, 13] where its performance was validated on small data sets. The basic problem with this approach is that the natural neighborhood relation must allow a move and reversal of a contiguous group of clones; in realistic scenarios the number of neighbors to consider becomes prohibitively large. An additional disadvantage is that it wrongly assumes clones overlap only with their immediate neighbors. The end result is a very poor performance, in both the quality

of solutions and the running time.

In order to overcome these problems, we tried an initial ordering of subgroups of tightly overlapping clones followed by simulated annealing on the order of the resulting *contigs*. The initial ordering algorithm was the same as that of Section 4.2.1, and it produced orders that are almost certainly correct. After this stage there are far less contigs than there were clones to begin with, so the number of neighbors is radically reduced. Moreover, the assumption of no overlap between non-neighboring contigs is usually accurate, unlike the similar assumption for clones. We implemented this idea, and it worked dramatically better than the original clone-level simulated annealing. However, performance was only about as good as the bottom-up algorithm of Section 4.1, and running time was much slower.

5 Simulation results

This section describes the simulations we have performed with the algorithm when clone positions, fingerprint data and errors were generated according to the statistical model described in Section 2.

5.1 Simulation setup

m clones of length l_c were positioned randomly with a uniform distribution along a genome of length L. The Poisson rate for each of the n probes was calculated from the probe length *plen* using the formula $\lambda = 1/4^{plen}$. Probe occurrences determined the probe-clone occurrence vector $\vec{A_i}$ for each clone C_i ($A_{i,j} = k$ iff probe j occurs k times on C_i). The probe hybridization probability was defined by $p(hyb_{i,j}) = p(\textit{false positive}) + (1 - p(\textit{false positive}))(1 - \alpha^{A_{i,j}})$, giving the expression $p(hyb_{i,j}) = 1 - e^{-\beta l_c}\alpha^{A_{i,j}}$. A binary hybridization pattern $\vec{B_i}$ was therefore chosen randomly according to the hybridization probabilities $p(hyb_{i,j})$.

The simulation program receives the following as instance generation parameters: L, l_c, n, α, *plen*, *f.p.prob* and the clone coverage. The number of clones is $m = \textit{coverage} \cdot L/l_c$. The false positives probability parameter *f.p.prob* is the probability of a false positive in a clone, and is related to β by the formula $\textit{f.p.prob} = 1 - e^{-\beta l_c}$.

The following combination of parameters was used as a *base scenario*: clone length $l_c = 40960$ bp (base pairs), genome length $L = 25$ clone lengths (approximately 1M bp), coverage of 10, false negative probability $\alpha = 0.2$, false positive probability $\textit{f.p.prob} = 0.05$, probe length $\textit{plen} = 8$, number

of probes $n = 500$. These parameters correspond to an experiment with cosmid clones, many probes and a realistic level of noise. Other scenarios were constructed by changing one of the parameters from its value in the base scenario.

5.2 Map accuracy measures

We developed several measures for quantifying the accuracy of maps produced by the algorithm. We report here on one measure that we have found most informative. It detects significant errors between the constructed and the correct map and ignores small differences that can be corrected by small shifts or local reordering. Sub-contigs of the constructed map between these significant errors are essentially correct, but the relative order and orientation of these sub-contigs are incorrectly determined. We call such errors *big errors*. For example, the map in Figure 3(c) contains one big error. A description of the calculation of big errors is included in Appendix B. When no big errors are found the map is essentially correct. A variable *any big* is defined to be 1 or 0 depending on whether there are any big errors. Its average is used to estimate the probability of having at least one big error.

5.3 Test examples

Refer to Figure 3 for a graphical illustration of the algorithm's results. In typical runs in the standard scenario there are only only insignificant ordering errors between nearby clones. Increasing the genome length does not have a significant effect on performance (See also Section 5.5.) The last two subfigures illustrate what happens under very adverse conditions. The weak points detected by the algorithm (Section 4.2.3) enable the experimenter to pinpoint possible errors. This information can be used for a judicious choice of additional hybridization experiments, minimizing cost and human effort.

5.4 Performance in the base scenario

1000 separate experiments were performed in the base scenario (Section 5.1) in order to obtain accurate expectation and standard deviation values. In 0.961 ± 0.006 of the cases the algorithm made no big errors. Recall (Section 5.2) that the absence of big errors is a guarantee that the map is essentially correct, as every two clones that are consecutive in reality are placed by the algorithm within a clone length from each other.

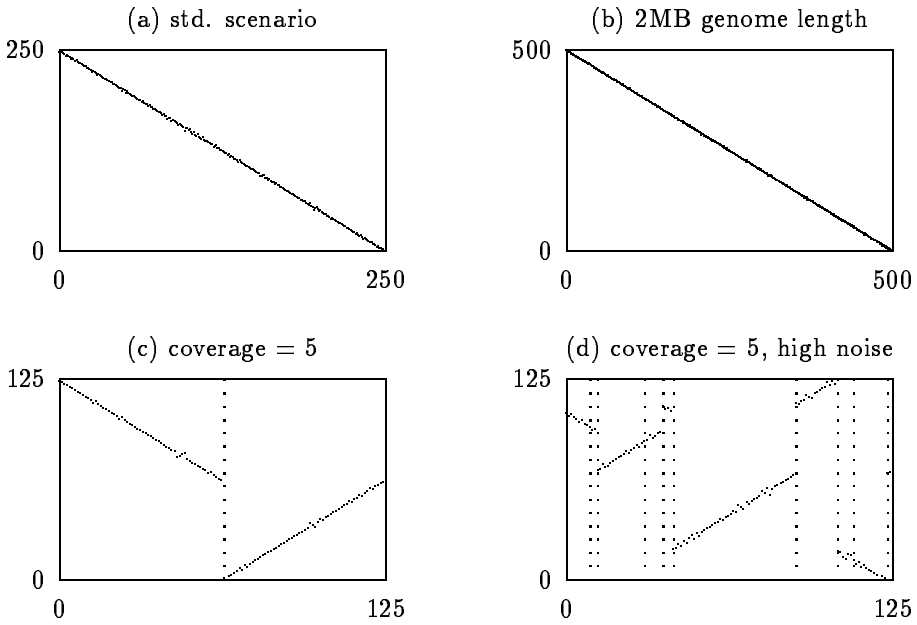

Figure 3: True clone order (y axis) vs. constructed order (x axis) in four scenarios. When applicable, weak points are shown as vertical dotted lines. The results are taken from simulations with the following scenarios: (a) base scenario, (b) a longer 2M bp genome, (c) very low coverage (5), (d) very low coverage (5) and very high noise ($\alpha = 0.3$ and $f.p.prob = 0.25$). Note that all big errors were detected as weak points, though some weak points are incorrectly predicted additional big errors.

5.5 The effect of separate parameters

The effect of separate parameters was tested by altering one parameter, while keeping the other parameters at their value in the base scenario. 1000 separate experiments were performed for each value, and the probability of having big errors was estimated by the average of *any big*.

The tested parameters (Section 5.1) were: $5 \leq L \leq 50$ (in clone lengths) in steps of $\Delta L = 5$, $7 \leq plen \leq 10$ in steps of $\Delta plen = 1$, $50 \leq n \leq 1000$ in steps of $\Delta n = 50$ up to $n = 200$ and $\Delta n = 100$ between 200 and 1000, $0 \leq \alpha \leq 0.8$ in steps of $\Delta\alpha = 0.1$, $0 \leq f.p.prob \leq 0.5$ in steps of $\Delta f.p.prob = 0.05$.

The results in Figure 4 demonstrate robustness to experimental noise. Even with $\alpha = 0.6$ the algorithm makes no big errors in the majority of the runs. The algorithm also copes well with relatively low coverage, but

performance deteriorates rapidly below coverage 7. The algorithm is most sensitive to the frequency of probe occurrences, as it is mirrored in the probes length. Sensitivity to the number of probes becomes pronounced with less than 200 probes. The results for large map sizes indicate that the algorithm is still robust in a genome of about 50 clone lengths (2M bp). We expect acceptable performance even with larger map sizes. In the discussion section we comment on the implication of these results on the design of hybridization experiments.

We also tested the influence of variable clone lengths. Clones lengths were chosen from a uniform distribution, and the construction program estimated the lengths by the maximum likelihood length as a function of the total number of hybridizations with the clone. The results in Figure 5 show a robustness to about 37% variability. Better performance can be expected with a normal distribution of clone lengths, which is probably more realistic.

5.6 Physical resources

A PC with a 100Mhz Pentium Microprocessor and 32MB of memory (256K cache) was used. The implementation was written using GNU C++, and run under the LINUX Unix type operating system. An average run of the construction algorithm (excluding instance generation) took 10 seconds in the base scenario and required 3MB of memory (memory requirements vary somewhat between runs, as the clones are divided into groups differently).

6 Results on real DNA sequences

Our next step was to test the algorithm in simulations involving real DNA sequences. These sequences were taken from a variety of representative organisms: The bacteria *Escherichia coli* [2], *Bacillus subtilis* [3], and *Mycobacterium tuberculosis* [4], the nematode *C. elegans* [5], the yeast *Saccharomyces*

[2]The E. Coli Genome Project, University of Wisconsin Madison. ftp://ftp.genetics.wisc.edu/pub/sequence/ecolim52.fas.gz

[3]Pöle Bio-Informatique Lyonnais, http://pbil.univ-lyon1.fr/nrsub/release.html

[4]The Sanger Centre. ftp://ftp.sanger.ac.uk/pub/tb/sequences

[5]The Sanger Centre. ftp://ftp.sanger.ac.uk/pub/databases/C.elegans_sequences/FINISHED_SEQUENCES/large_C._elegans_contig.

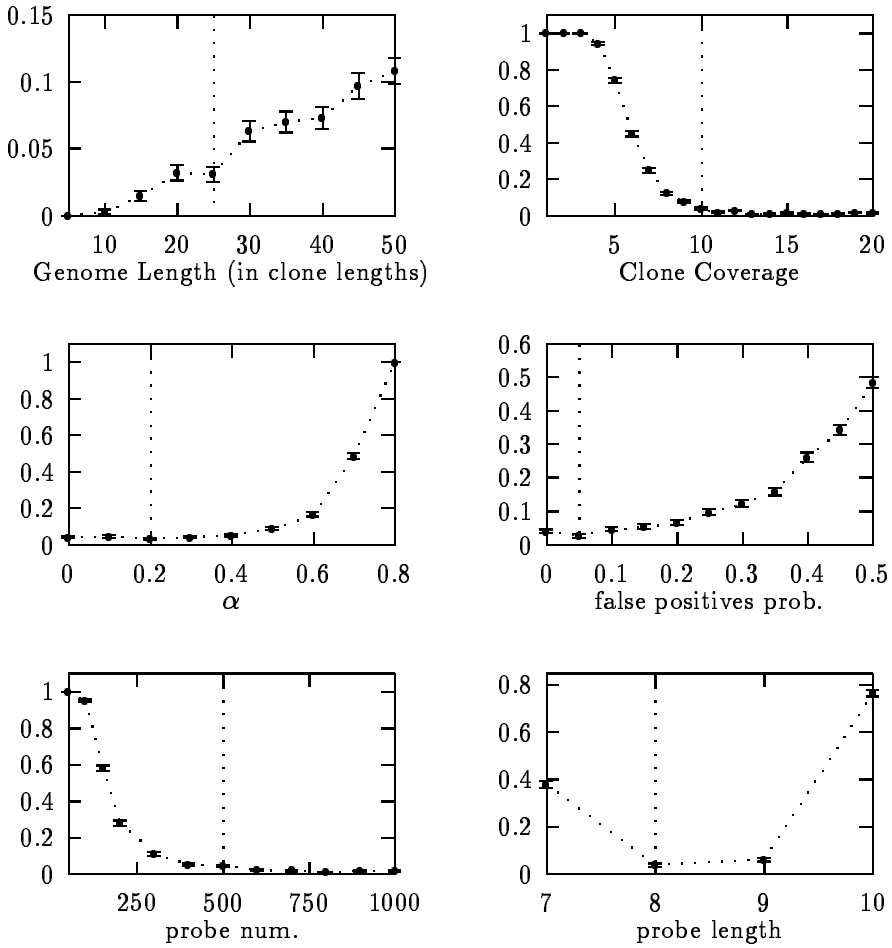

Figure 4: The influence of the various simulation parameters on the probability of having big errors (as estimated from the average of *any big*). Error bars denote one standard deviation from the mean. The vertical dotted line indicates the value of the parameter in the base scenario. Note that the effect of a decrease in the number of probes is very similar to that of an increase in the experimental noise. This is because noise decreases the informational content of each probe, an effect that can be countered by an increase in the number of probes. It is also notable that the probe size has a very significant effect, resulting from its direct influence on the frequency of probe occurrences, and therefore on the informational content of the experiment. By contrast, the genome size has only a moderate effect on performance.

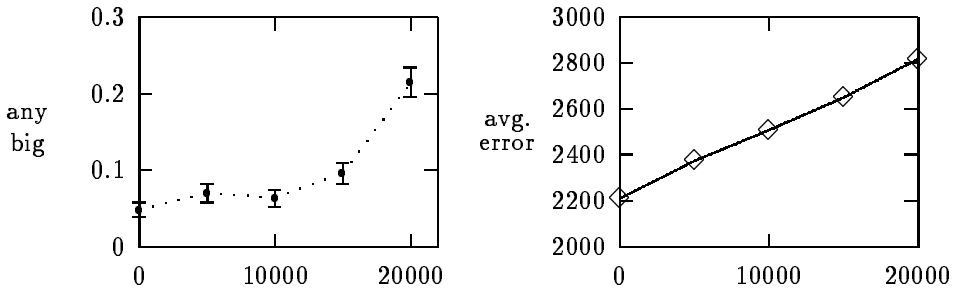

Figure 5: The influence of clone length variability. The x axis represents the maximal variability from the average clone sized length. Clone lengths were uniformly distributed in this range. The graphs show good performance with a variability of up to about 15000 bp, or 37%.

cerevisiae[6], and *H. sapiens*[7]. The lengths of the sequences ranged from about 1.7MB to over 4MB. Two disjoint 1MB sections from each genome (1MB and 730KB for *H. sapiens*) were used for the tests. An additional random DNA sequence was used for comparison.

We use a same-rate Poisson process to model probe hybridizations. The distributions of codons in real DNA is far from random, and we can therefore expect our model to fail when tested on real DNA. We tested this prediction by counting the number of probe hybridizations for all possible 8mers. The results in Figure 6 demonstrate the failure of the same-rate model. One remedy that we considered was to replace the model with a separate Poisson model for each probe. The disadvantages of this idea are that (1) Most probes occur very infrequently, and will therefore provide very little information, and (2) It is difficult to estimate correctly the statistics of rarely occurring probes. A more promising idea proved to be to use the fact that for all tested organisms a significant fraction of 'good' probes falls under the graph of the random DNA, i.e., are in good agreement with the same-rate

[6]Chromosomes iv,vii, and xv taken from the Munich Information Centre for Protein Sequences (MIPS) at http://www.mips.biochem.mpg.de/mips/programs/get_fragment.html.

[7]Chromosome 7 DNA from the University of Washington Genome Center (UWGC), http://chimera.biotech.washington.edu/UWGC. The sequence itself, of length 1050106 bp is available through a link in the above web site into GenBank. Chromosome 17 DNA was taken from the Whitehead Institute/MIT Genome Sequencing Project, http://www-seq.wi.mit.edu/sequencing/public_release/human17.html. The following files were used: L106,L107,L108,L117,L118,L127. The chromosome 7 data was used as the main sequence file. The chromosome 17 DNA contigs were concatenated into one sequence and used as an additional DNA file.

model. Because probe hybridizations are effort and cost intensive it is impractical to make very many experiments and then choose a small subset of "good" probes. However, if probes can be chosen before the actual experiment based on prior knowledge of the organism's typical sequences (e.g., from other sequenced parts of its genome), better results can be achieved. We call this process *probe pre-selection*. Figure 7 demonstrates that a fairly good fit with the same-rate model can be achieved.

In spite of the great improvement that probe pre-selection offers, the distributions in Figure 7 still have very long tails. These tails represent probes which either occur very rarely or else occur almost in all clones. In order to prevent such probes from hindering the algorithm's performance we successfully experimented with *probe post-screening*, namely retaining only those probes which according to their hybridization fingerprints fit well the same-rate model. Because of probe pre-screening most probes can be retained with only a small loss of usable hybridization information. Based on the noise parameters, we compute the desired minimum and maximum number of hybridizations, and retain only the appropriate probes. Alternatively, if noise estimate is unreliable, it is possible to calculate a histogram of the number of hybridizations across all probes, and retain only the probes in the central part of the histogram.

We tested both probe pre-screening on itself, and the combination of pre-screening and post-screening. The results in Figure 8 and Table 1 demonstrate acceptable results even with probe pre-screening alone, and significantly better results when both algorithms are used. These encouraging results suggest it is realistic to expect good physical mapping results in spite of the highly non-random statistics of real DNA sequences.

7 Concluding remarks

The basic simulations demonstrate robust performance in scenarios containing a large genome and highly noisy hybridizations. The algorithm is also robust to considerably more clone length variability than is the case with cosmid clones. With clones of widely variable lengths some length screening is likely to contribute significantly to performance. The basic simulations also show the importance of using many probes. The manufacturing of many probes involves high initial costs, but the same probes can later be used for the entire genome. The per-experiment costs are then much smaller than for an experiment using unique probes.

A key issue emerging from the simulations is probe selection. The basic

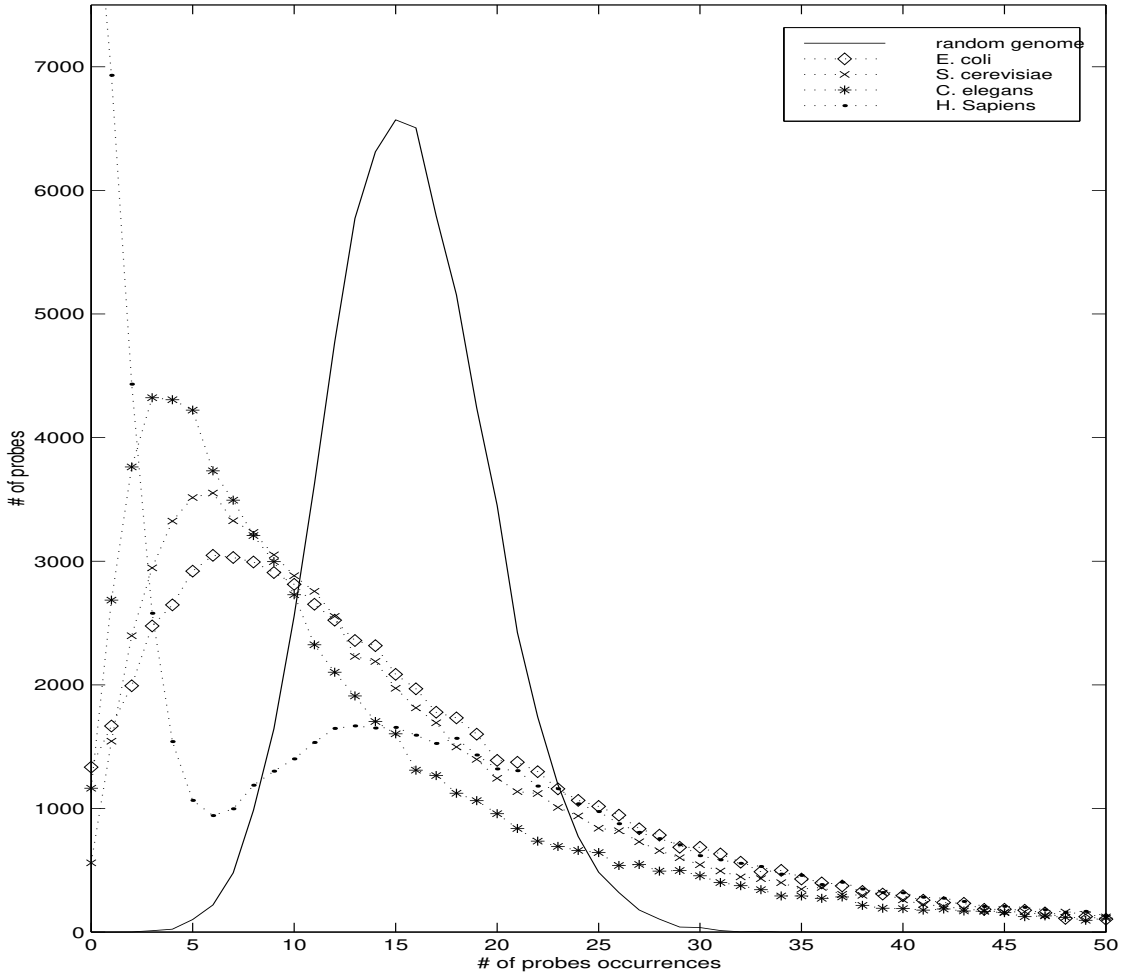

Figure 6: Histogram of the number of 8-mer probe occurrences along a genome section of length 1MB. The number of occurrences for each possible 8-mer was recorded, and the resulting histogram is plotted. The results for *M. tuberculosis* and *B. subtilis* were essentially identical for those of *E. coli* and were omitted for clarity. Other genome samples from the same organisms produced almost identical histograms. In contrast to this intra-organism consistency, the distributions for different organisms are quite different. In particular, probe occurrences on the human DNA are very differently distributed than the other tested organisms. Most importantly for our purposes, probe occurrences on all tested organisms are very different from the Poisson distribution exhibited by the random genome.

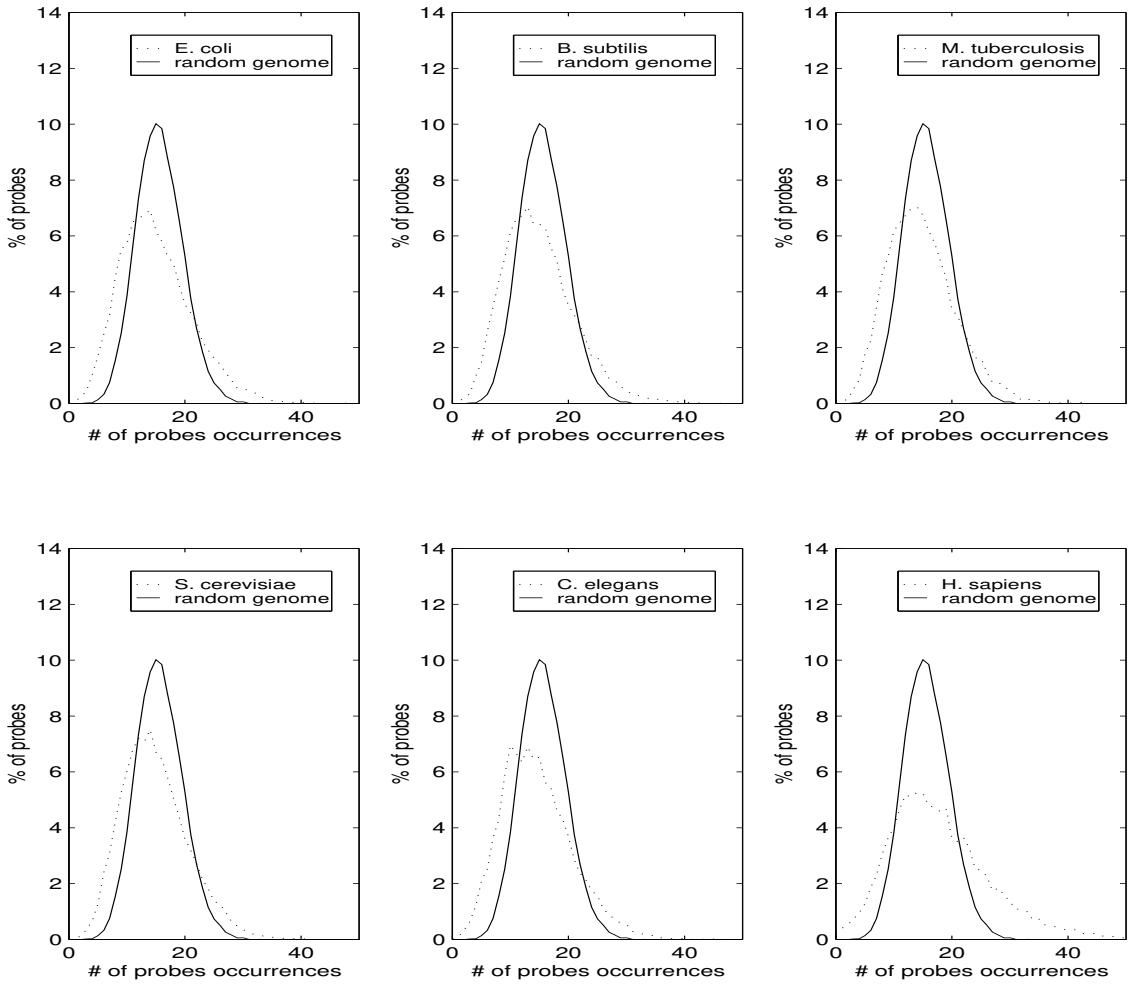

Figure 7: Histograms of the number of probe occurrences when probe pre-selection is used on genome sections of length 1MB. Probes occurring between 0.75 and 1.25 of the average number of occurrences were used, as estimated from a *different* genome section of length 1MB (730KB for human) of the same organism. The x-axis denotes the number of probes occurrences, and the y-axis is the percentage of probes with that many occurrences. The figures also contain the random distribution for reference. A comparison with Figure 6 shows that this type of pre-selection process dramatically improves the fit between the distribution of probe occurrences and the same-rate model.

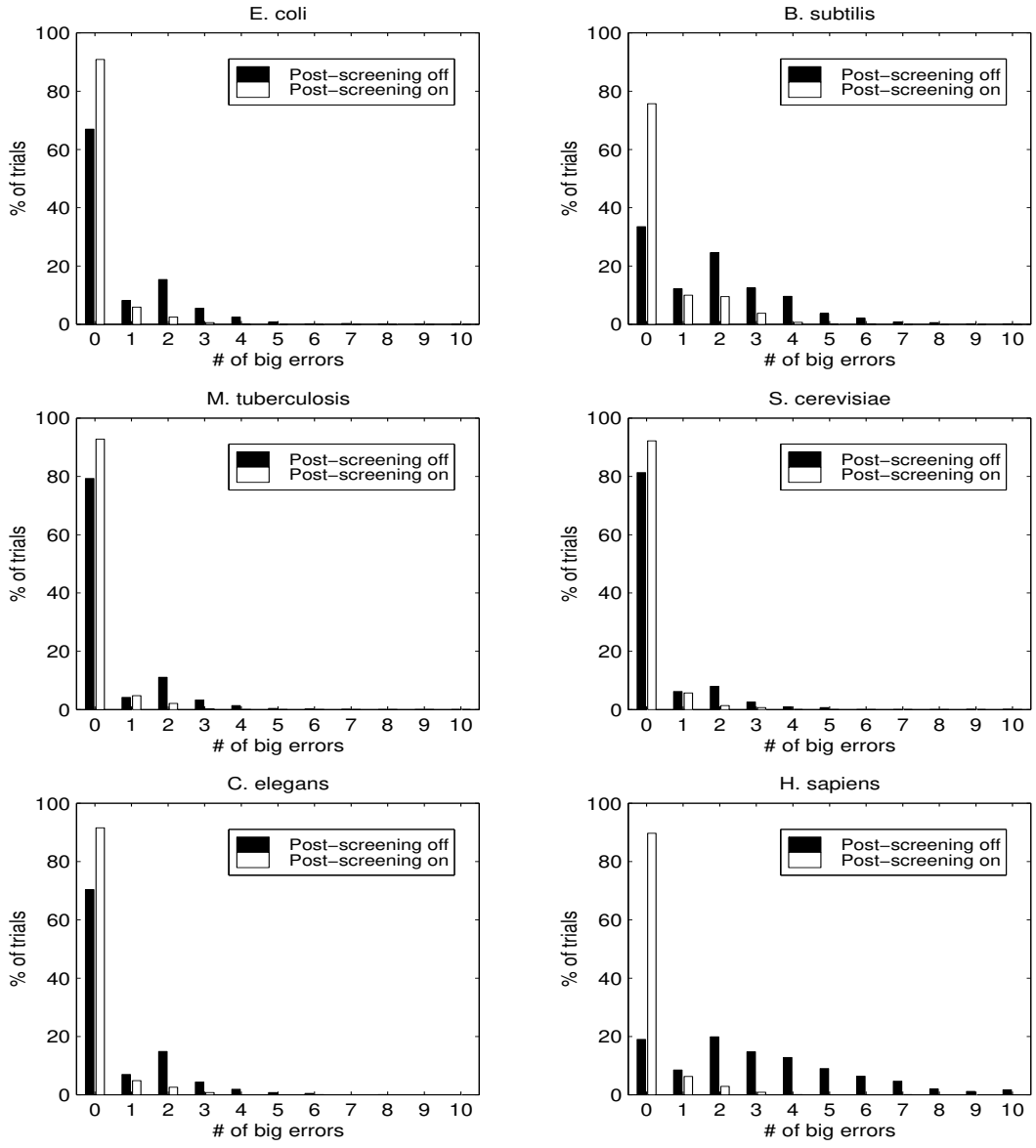

Figure 8: The effect of probe post-screening on the distribution of the number of big errors (1000 trials in each case). Pre-selection was applied in all cases. See also Table 1. Refer to the text for a discussion of these results. Note that even without post-screening the *average* number of big errors is quite small (about 3 for *H. Sapiens*, below 2 for the other species).

Organism	% trials with no big errors post-screening off	% trials with no big errors post-screening on	# of remaining probes after post-screening
E. coli	67.0 ± 1.5	90.9 ± 0.9	299 ± 11
B. subtilis	33.5 ± 1.5	75.7 ± 1.4	306 ± 11
M. tuberculosis	79.3 ± 1.3	92.8 ± 0.8	308 ± 11
S. cerevisiae	81.3 ± 1.2	92.2 ± 0.8	313 ± 11
C. elegans	70.4 ± 1.4	91.6 ± 0.9	297 ± 11
H. sapiens	19.0 ± 1.2	89.7 ± 1.0	259 ± 11

Table 1: The effect of probe post-screening on the probability of achieving an error free map (1000 trials in each case). Probes occurring between 0.75 and 1.25 of the average number of occurrences were used for both pre-selection and post-screening processes. A significant improvement is evident when probe post-screening is used in addition to pre-screening. The effect is particularly dramatic with the *H. sapiens* DNA, and may be the case with other mammalian genomes. Relatively poor results were achieved with the *B. subtilis* genome. Approximately 60% of the probes were retained after probe post-screening. A finer tuning of the post-screening parameters is likely to improve results somewhat.

simulations demonstrate how crucial it is to choose probes with approximately 50% hybridizations. The simulations on real DNA show that random probes are a very poor choice with real DNA. They also show that by using pre-selection based on known sequence statistics specific to the organism, the algorithm could then apply post-screening and still have a sufficient number of probes to work well.

Short hybridization techniques are challenged by the difficulty to predict the strength of hybridization (and hence of the signal) between an oligo and a particular sequence: Factors such as GC content, mismatches and local fold structure influence the actual signal. A very important observation is that accurate prediction is not really necessary for the success of the algorithm. All that is required is that the hybridization strength between any probe and any point (a sequence segment) in the target will be *consistent* in all the clones that contain that point. The actual level of match between the sequence of the oligo and the sequence at that point in the target is immaterial. Any relation whatsoever can exist between the two sequences, provided that consistency is maintained. For example, a probe and a target sometimes hybridize well even when only a partial complementary match

exists between them, but this is not a problem for the algorithm. This type of consistent deviation from a simple hybridization model has to be clearly differentiated from uncorrelated deviations of individual clone and probe pairs. The latter deviations are the false positive and the false negative errors.

The statistical inference approach to analyzing oligonucleotide hybridization data is not limited to physical mapping. In *cDNA clustering* [6], clone pairs can either fully overlap (one is contained in the other), or else come from different genes and have zero overlap. A likelihood ratio of these two possibilities can be used as input to the clustering algorithm. Initial experiments that we have done [10] show that it may give substantially better results than the currently used Euclidean norm or Hamming distance.

Another particularly promising field for applications is preselection of clones for shotgun sequencing. The problem is to select as small a subset as possible of clones covering the target genome, sequence only those first and thereby avoid re-sequencing the same target parts unnecessarily and save time and cost in sequencing [15]. Relatively inexpensive oligonucleotides fingerprinting is performed, and the resulting hybridization data can be used to produce a physical map, from which a near optimal clone subset can be chosen. In this problem the clone lengths are typically of the order of 1200bp, and so 8mers are suboptimal as probes, as their occurrence probability is low. However, it has been suggested (Uwe Radelof—personal communication) that oligo pooling can be used, with a pool of several 8mers serving as one probe, thus making it possible to achieve around 50% hybridization.

A final possible application is to restriction digest mapping. In this problem each clone is completely digested by a restriction enzyme and the input data for mapping is a list of fragment lengths for each clone. Our algorithm can be applied also to this seemingly very different type of data: On an abstract level, all the algorithm needs in order to work is features or landmarks along the DNA, where the appearance or absence of each feature is consistent in all the clones containing the corresponding DNA segment. If restriction fragment lengths serve as the features, the algorithm can be applied to restriction mapping too.

Acknowledgment

This research was supported in part by grants from the G.I.F., the German-Israeli Foundation for Scientific Research and Development, and from the Ministry of Science and Technology, Israel. We thank Hans Lehrach, Steffen

Hennig, Uwe Radelof, John O'Brien, and Ralf Herwig from the Max Planck Institute, Berlin for many discussions and insights on the fingerprinting techniques and their use for mapping.

References

[1] F. Alizadeh, R. M. Karp, L. A. Newberg, and D. K. Weisser. Physical mapping of chromosomes: A combinatorical problem in molecular biology. *Algorithmica*, 13:52–76, 1995.

[2] D.R. Cox, M. Burmeister, R. Price, S. Kim, and R.M. Myers. Radiation hybrid mapping: a somatic cell genetic method for reconstructing high-resolution maps of mammalian chromosomes. *Science*, 250:245–250, 1990.

[3] A. G. Craig, D. Nizetic, D. Hoheisel, G. Zehetner, and H. Lehrach. Ordering of cosmid clones covering the herpes simplex virus type I (HSV-I) genome: A test case for fingerprinting by hybridization. *Nucleic Acids Research*, 18:2653–2660, 1990.

[4] A. Cutticchia, J. Arnold, and W. Timberlake. ODS: ordering DNA sequences - a physical mapping algorithm based on simulated annealing. *CABIOS*, 9:215–219, 1993.

[5] Y.-X. Fu, W. E. Timberlake, and J. Arnold. On the design of genome mapping experiments using short synthetic oligonucleotides. *Biometrics*, 48:337–359, 1992.

[6] E. Hartuv, A. Schmitt, J. Lange, S. Meier-Ewert, H. Lehrach, and R. Shamir. An algorithm for clustering cDNAs for gene expression analysis using short oligonucleotide fingerprints. In *Proceedings Third International Symposium on Computational Molecular Biology (RECOMB 99)*, 1999. to appear.

[7] J.D. Hoheisel, E. Maier, R. Mott, L. McCarthy, A.V. Grigoriev, L.C. Schalkwyk, D. Nizetic, F. Francis, and H. Lehrach. High resolution cosmid and P1 maps spanning the 14 Mb genome of the fission yeast *S. pombe*. *Cell*, 73:109–120, 1993.

[8] E. S. Lander and M. S. Waterman. Genomic mapping by fingerprinting random clones: A mathematical analysis. *Genomics*, 2:231–239, 1988.

[9] H. Lehrach, R. Drmanac, J. Hoheisel, Z. Larin, G. Lennon, A. P. Monaco, D. Nizetic, G. Zehetner, and A. Poustka. Hybridization fingerprinting in genome mapping and sequencing. In K. E. Davies and S. Tilghman, editors, *Genome Analysis: Genetic and Physical Mapping*, pages 39–81. Cold Spring Harbor, New York, 1990.

[10] G. Mayraz and R. Shamir. A Bayesian score for cDNA clustering. Technical report, School of Mathematics, Tel-Aviv University, Tel-Aviv, Israel, 1997.

[11] F. Michiels, A. G. Craig, G. Zehetner, G. P. Smith, and H. Lehrach. Molecular approaches to genome analysis: a strategy for the construction of ordered overlapping clone libraries. *CABIOS*, 3(3):203–210, 1987.

[12] R.F. Mott, A.V. Grigoriev, E. Maier, J.D. Hoheisel, and H. Lehrach. Algorithms and software tools for ordering clone libraries: appplication to the mapping of the genome of *Schizosaccharomyces pombe*. *Nucleic Acids Research*, 21:1965–1974, 1993.

[13] D. M. Platt. and T. I. Dix. Comparison of clone-ordering algorithms used in physical mapping. *Genomics*, 40:490–492, 1997.

[14] A. Poustka, T. Pohl, D.P. Barlow, G. Zehetner, A. Craig, F. Michiels, E. Ehrich, A.M. Frischauf, and H. Lehrach. Molecular approaches to mammalian genetics. *Cold Spring Harb. Symp. Quant. Biol.*, 51:131–139, 1986.

[15] U. Radelof, S. Hennig, P. Seranski, M. Steinfath, J. Ramser, R. Reinhardt, A. Poustka, F. Francis, , and H. Lehrach. Preselection of shotgun clones by oligonucleotide fingerprinting: An efficient and high-throughput strategy to reduce redundancy in large-scale sequencing projects. *Nucleic Acids Res.*, 26(23):5358–5364, 1998.

A The Mathematics of the Bayesian Score

This appendix gives the mathematical details of the Bayesian score used by the construction algorithm and the computation of the big errors used to assess the solution quality. A description of the clone pairs overlap score is followed by that of the contig pairs overlap score, which utilizes the results of the former. The discussion does not depend on the assumption of equal clone lengths. Finally the computation of big errors is presented.

A.1 Clone pairs overlap score

Consider a pair of clones C_a and C_b, viewed as intervals of lengths l_a and l_b respectively. Without loss of generality, assume $l_a \leq l_b$. Let $C'_\gamma = C_a \cap C_b$, $l_\gamma = |C'_\gamma|$. The overlap score uses the hybridization vectors $\vec{B_a}$ and $\vec{B_b}$ to produce a probabilities vector for the overlap length l_γ.

We first calculate $Prob(\vec{B_a}, \vec{B_b} | l_\gamma = t), 0 \leq t \leq l_a$. Let $C'_a = C_a \setminus C_b$, and $C'_b = C_b \setminus C_a$. Define $A_{a,j}$ as the number of occurrences of probe j in C_a, similarly defining $A_{b,j}$. Define $A'_{i,j}$ as the number of occurrences of probe j in C'_i for $i = a, b, \gamma$. Then:

$$
\begin{aligned}
& Prob(B_{a,j}, B_{b,j} | l_\gamma = t) \qquad (1) \\
&= \sum_{k'_a=0}^{\infty} \sum_{k'_b=0}^{\infty} \sum_{k'_\gamma=0}^{\infty} Prob(B_{a,j} | A_{a,j} = k'_a + k'_\gamma) Prob(B_{b,j} | A_{b,j} = k'_b + k'_\gamma) \\
& \quad Prob(A'_{a,j} = k'_a | l_\gamma = t) Prob(A'_{b,j} = k'_b | l_\gamma = t) Prob(A'_{\gamma,j} = k'_\gamma | l_\gamma = t)
\end{aligned}
$$

(Here $A_{a,j} = A'_{a,j} + A'_{\gamma,j}$ and $A_{b,j} = A'_{b,j} + A'_{\gamma,j}$). The calculation of the probabilities inside the summation symbols is straightforward using the statistical model. Since hybridization is a virtual certainty if a probe occurs many times inside a clone, we can limit the summation to small values of k (say $0 \leq k \leq 5$), thereby making feasible the score's computation while introducing only a negligible error.

Considering each probe as an independent source of information, the conditional probability of the vector pair ($\vec{B_a}$,$\vec{B_b}$) is

$$
Prob(\vec{B_a}, \vec{B_b} | l_\gamma = t) = \prod_{j=1}^{n} Prob(B_{a,j}, B_{b,j} | l_\gamma = t) \qquad (2)
$$

Assuming uniform parameters for the probes, the expression inside the product, $Prob(B_{a,j} = x, B_{b,j} = y | l_\gamma = t)$, is independent of j. Therefore, we can

define $P^t_{x,y}$ for $0 \leq x, y \leq 1$ by $P^t_{x,y} = Prob(B_{a,j} = x, B_{b,j} = y|t)$. Denoting by $S_{x,y}$ the set of probes $1 \leq j \leq n$, such that $B_{a,j} = x$ and $B_{b,j} = y$, we can write:

$$Prob(\vec{B_a}, \vec{B_b}|t) = \prod_{x=0}^{1} \prod_{y=0}^{1} P^t_{x,y}{}^{|S_{x,y}|} \tag{3}$$

Having computed $Prob(\vec{B_a}, \vec{B_b}|l_\gamma = t)$ we can use Bayes Theorem:

$$Prob(l_\gamma = t_0|\vec{B_a}, \vec{B_b}) = \frac{Prob(\vec{B_a}, \vec{B_b}|l_\gamma = t_0)Prob(l_\gamma = t_0)}{\sum_{t=0}^{l_a} Prob(\vec{B_a}, \vec{B_b}|l_\gamma = t)Prob(l_\gamma = t)} \tag{4}$$

The actual computation uses logarithms to avoid accuracy and underflow problems. We also use only quantized values for l_γ, so that $l_\gamma = 0, \frac{l_a}{d}, \frac{2l_a}{d}, \ldots, l_a$. Because the probability of each of the four possible hybridization combinations is either increasing with l_γ or else decreasing with l_γ, the maximal error introduced by the quantization is less than the quantization unit length d. The experiments in this paper use a quantization unit size of one tenth of a clone length.

A.1.1 Complexity

With the aid of simple tables, the calculation of the Bayesian score for a pair of clones consists of the following steps:

1. Counting the number of probes with each of the four possible hybridization combinations in the pair $(B_{a,x}, B_{b,y})$ ($|S_{x,y}|$ in equation 3). This calculation requires $O(n)$ steps.

2. Calculating $\log Prob(\vec{B_a}, \vec{B_b}|l_\gamma = t)$ using quantized values of t. The probabilities $P^t_{x,y}$ are calculated in advance and stored in a table. Such a table can be used for variable sized clones as well as for fixed size clones. Using this table the calculation takes $O(d)$ steps.

3. Normalization (Bayes Theorem), taking $O(d)$ steps.

The overall complexity is, thus, is $O(n + d)$, where n is the number of probes, and d is the number of possible quantized values for l_γ. Typically, $d \ll n$, so the total complexity is proportional to the number of probes.

A.2 Contig pairs overlap score

A *contig* is a set of clones covering a contiguous region of the genome together with their relative placements. Consider the problem of estimating the posterior probabilities for the relative placement of two contigs G_a and G_b. Two subsets (B_a, B_b) of the rows of the clone hybridization matrix B correspond to the two contigs. Before applying Bayes Theorem we need to estimate $P(B_a, B_b | p_{a,b} = t)$ for a relative placement $p_{a,b}$ of G_b with respect to G_a (As the probabilities of contigs relative placements depend on their relative orientation, the overlap section length by itself does not determine the relative placement). Consider a pair of clones (C_i, C_j) from G_a and G_b respectively. The relative placement $p_{a,b}$ of the contigs G_a and G_b determines the overlap $o_{i,j}$ between the two clones. As we already know how to calculate $P(\vec{B_i}, \vec{B_j} | o_{i,j})$ we can calculate $P(\vec{B_i}, \vec{B_j} | p_{a,b})$. Going over all the possible relative placements of (G_a, G_b) and normalizing the resulting vector (Bayes Theorem) we get an estimate of $P(p_{a,b} = t | B_a, B_b)$ based on the information from the two clones C_i and C_j. Let $\vec{R}^{i,j}$ stand for this probabilities vector. Going over all the relevant clone pairs we make the following definition:

$$S_t \stackrel{\text{def}}{=} \prod_{(i,j)|C_i \in G_a, C_j \in G_b} R_t^{i,j} \tag{5}$$

$\vec{S}$ is the vector of posterior probabilities: $S_t \approx Prob(p_{a,b} = t | B_a, B_b)$, approximated under the assumption that the information from each of the clone pairs is independent. This assumption is not true, and it tends to sharpen the distribution of the posterior probabilities in comparison with the true distribution. However, the approximation enables the construction algorithm to integrate information from many clone pairs, and arrive at a much more informed contig merger decision.

A.2.1 Complexity

The space taken by the two contigs is at most L, giving a complexity of $O(n + L)$ for calculating the contribution of each clone pair to the contig pairs score. In the worst case there are $m/2$ clones in each contig giving an overall complexity of $O((n + L)m^2)$.

clone no.	1	2	3	4	5	6	7	8	9	10
location in $\mathcal{A}$	0	6	6	13	13	18	19	21	24	32
location in $\mathcal{P}$	6	13	13	18	-3	-10	-9	-9	23	30
location in $\overline{\mathcal{P}}$	24	17	17	12	33	40	39	39	7	0
errors w.r.t $\mathcal{P}$	-	-	-	+	+	-	-	+	-	
errors w.r.t $\overline{\mathcal{P}}$	+	-	+	+	-	-	-	+	+	
big errors	-	-	-	+	-	-	-	+	-	

Table 2: Example of the computation of big errors. The locations of the 10 clones in the algorithmic solution $\mathcal{A}$ is presented together with their location in the correct solution in both orientations $\mathcal{P}$ and $\overline{\mathcal{P}}$. Clone locations in $\overline{\mathcal{P}}$ were defined by $\overline{P}_i = 30 - P_i$, with 30 being an arbitrary constant. The value L is set to 10, the clones length. Errors with respect to $\mathcal{P}$ and $\overline{\mathcal{P}}$ are marked with + signs. The last line in the table marks big errors, as determined from the previous two lines.

B Computing big errors

For simplicity the presentation below is done for fixed length clones, but the computation easily extends to variable lengths clones.

By a *location* of a clone we mean the coordinate of its left endpoint. We number of clones in non-decreasing order of the locations in the algorithmic solution. The solution is thus an ordered sequence $\mathcal{A} = A_1 \leq A_2 \leq \ldots \leq A_n$. Let $\mathcal{P}$ be the correct solution, with P_i the location of clone i, indexed using the same indices as in $\mathcal{A}$. Let $\overline{\mathcal{P}}$ be the reverse of solution $\mathcal{P}$. That is, $\overline{P_i} = S - P_i$ for some constant S.

We say that *an error with respect to* $\mathcal{P}$ occurred at position i of $\mathcal{A}$ if $|(A_{i+1} - A_i) - (P_{i+1} - P_i)| \geq L$, where L is some constant (in practice we used the average clone length). We similarly define an error with respect to $\overline{\mathcal{P}}$.

The determination of big errors is done as follows: We first check for each $1 \leq i \leq n$ whether an error occurred at position i w.r.t. $\mathcal{P}$. A similar check is done for errors w.r.t. $\overline{\mathcal{P}}$. Then, starting from position 1 we compute the position of the next error w.r.t. $\mathcal{P}$ and $\overline{\mathcal{P}}$, and pick the larger position i as the place of the next big error. The process is then repeated starting from position $i + 1$. Table 2 and Figure 9 demonstrate the computation on a small example.

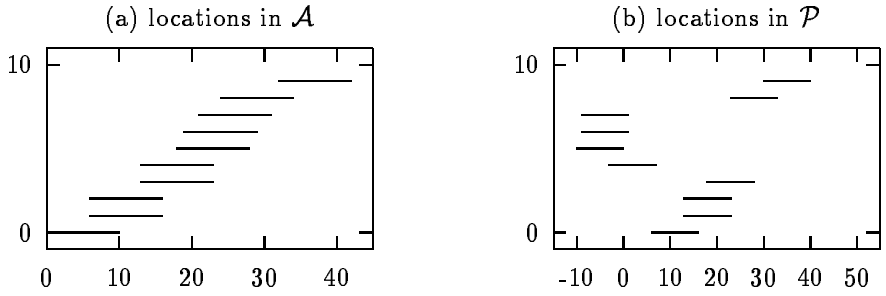

Figure 9: Physical maps for the data in Table 2. In both parts of the figure the y-axis corresponds to the clone indices in the algorithmic solution. In part (a) the x-axis denotes clone locations in algorithmic solution. In part (b) the x-axis denotes clone locations in the correct solution. Note that regardless of the correctness of the algorithmic solution, the clone locations in $\mathcal{A}$ form a non-decreasing series. It is in a graph such as (b), that the quality of the solution can be assessed. In this case, a division into three section is apparent, corresponding to the two big errors. Note also that in the first and third regions the orientation appears to be correct, whereas in the second region a reversed orientation fits $\mathcal{A}$ better.