A PROBABILISTIC METHOD FOR SMALL RNA FLOWGRAM MATCHING

VLADIMIR VACIC[1], HAILING JIN[2],
JIAN-KANG ZHU[3], STEFANO LONARDI[1]
[1] *Computer Science and Engineering Department,* [2] *Department of Plant Pathology,* [3] *Department of Botany and Plant Sciences,*
University of California, Riverside

The 454 pyrosequencing technology is gaining popularity as an alternative to traditional Sanger sequencing. While each method has comparative advantages over the other, certain properties of the 454 method make it particularly well suited for small RNA discovery. We here describe some of the details of the 454 sequencing technique, with an emphasis on the nature of the intrinsic sequencing errors and methods for mitigating their effect. We propose a probabilistic framework for small RNA discovery, based on matching 454 flowgrams against the target genome. We formulate flowgram matching as an analog of profile matching, and adapt several profile matching techniques for the task of matching flowgrams. As a result, we are able to recover some of the hits missed by existing methods and assign probability-based scores to them.

1. Introduction

Historically, the chain termination-based Sanger sequencing[17] has been the main method to generate genomic sequence information. Alternative methods have been proposed, among which a highly parallel, high-throughput pyrophosphate-based sequencing (pyrosequencing)[16] is one of the most important. 454 Life Sciences has made pyrosequencing commercially available[11] and the resulting abundance of 454-generated sequence information has prompted a number of studies which compare 454 sequencing with the traditional Sanger method (see, e.g., [3,6,8,12,20]).

454 pyrosequencing. In the 454 technology, the highly time-consuming sequence preparation step which involves production of cloned shotgun libraries has been replaced with much faster PCR microreactor amplification. Coupled with the highly parallel nature of 454 pyrosequencing, this novel technology allows 100 times faster[8] and significantly less expensive sequencing. A detailed step by step breakdown of time required to complete the process using both methods can be found in Wicker *et al.*[20]

Recent studies by Goldberg *et al.*[8] on sequencing six marine microbial genomes and by Chen *et al.*[3] on sequencing the genome of *P. marinus* report that 454's ability to sequence throughout the regions of the genome with strong secondary structure and the lack of cloning bias represent a comparative advantage. However, the 454's shorter read lengths (100 bp on average compared to 800-1000 bp of Sanger) make it very hard if not impossible to span long repetitive genomic elements. Also, the lack of paired end reads (mate pairs) limits the assembly to contigs separated by coverage gaps. As a consequence, both studies conclude that at the present stage 454 pyrosequencing used alone is not a feasible method for *de novo* whole genome sequencing, although these two issues are being addressed in the new 454 protocol. Another problem inherent to pyrosequencing is accurate determination of the number of incorporated nucleotides in homopolymer runs, which we discuss in Section 2.

Small RNA. Since its discovery in 1998[4], gene regulation by RNA interference has received increasing attention. Several classes of non-coding RNA, typically much shorter than mRNA or ribosomal RNA, have been found to silence genes by blocking transcription, inhibiting translation or marking the mRNA intermediaries for destruction. Short interfering RNA (siRNA), micro RNA (miRNA), tiny non-coding RNA (tncRNA) and small modulatory RNA (smRNA) are examples of classes of small RNA that have been identified to date[13]. In addition to differences in genesis, evolutionary conservation, and the gene silencing mechanism they are associated with, different classes of small RNA have distinct lengths: 21-22 bp for siRNA, 19-25 bp for miRNA and 20-22 bp for tncRNA.

The process of small RNA discovery typically involves (1) sequencing RNA fragments, (2) matching the sequence against the reference genome to determine the genomic locus from which the fragment likely originated, and (3) analyzing the locus annotations in order to possibly obtain functional characterization. In this paper we focus on the second step.

Our contribution. 454 pyrosequencing appears to be particularly well-suited for small RNA discovery. The limited sequencing read length does not pose a problem given the short length of non-coding RNAs, even if we take into account lengths of adapters which are ligated on both ends on the small RNA prior to sequencing. Also, paired end reads are not required, as there is no need to assemble small RNA into larger fragments. Several projects have already used 454 to sequence non-coding RNA (see, e.g., [7,14]).

However, to the best of our knowledge, the issue of handling sequencing errors has not been addressed so far for short reads which occur in small RNA discovery. Observe that this problem could be mitigated in a scenario where an assembly step was involved – which is not the case when sequencing small RNA. In the following sections we describe the 454 sequencing model and the typical sequencing errors it produces. We propose a probabilistic matching method capable of locating some of the small RNA which would have been missed if the called sequences were matched deterministically. We adapt the enhanced suffix array[2] data structure to speed up the search process. Finally, we evaluate the proposed method on four libraries obtained by sequencing RNA fragments from stress-treated *Arabidopsis thaliana* plants and return 26.4% to 28.8% additional matches.

2. The 454 Pyrosequencing Method

In the 454 sequencing method, DNA fragments are attached to synthetic beads, one fragment per bead, and amplified using PCR to approximately 10 million copies per bead[11]. The beads are loaded into a large number of picolitre-sized reactor wells, one bead per reactor, and sequencing by synthesis is performed in parallel by cyclically flowing reagents over the DNA templates. Depending on the template sequence, each cycle can result in extending the strand complementary to the template by one or more nucleotides, or not extending it at all. Nucleotide incorporation results in the release of an associated pyrophosphate, which produces an observable light signal. The signal strength corresponds to the length of the incorporated homopolynucleotide run in the given well in that cycle. The resulting signal strengths are reported as pairs (nucleotide, signal strength), referred to as *flows*. The end result of 454 sequencing is a sequence of flows in the T, A, C, G order called a *flowgram*. Terms *positive flow* and *negative flow* denote, respectively, that at least one base has been incorporated, or that the reagent flowed in that cycle did not results in a chemical reaction, and hence that a very weak signal was observed. Every full cycle of negative flows would be called as an N, because the identity of the nucleotide could not be determined. Positive flow signal strengths for a fixed homopolynucleotide length l are reported to be normally distributed with the mean $0.98956 \cdot l + 0.0186$ and standard deviation proportional to l, while the negative flow signal strengths follow a log-normal distribution[11]. To the

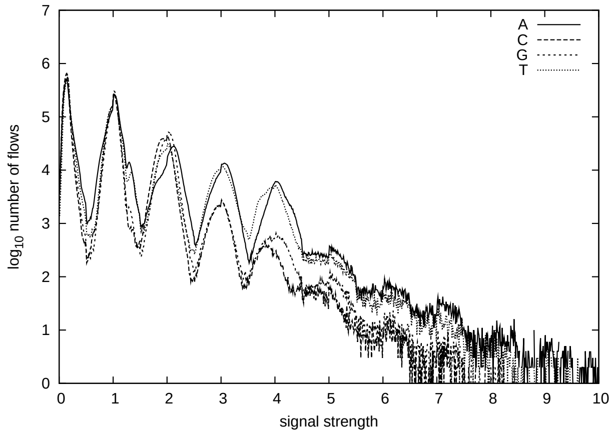

Figure 1. Distribution of signals for the *A. thaliana* pyrosequencing dataset

best of our knowledge, the other parameters of the normal and log-normal distributions have not been reported in the literature. In Section 7 we are estimating the remaining parameters from the available data.

Figure 1 shows the distribution of signal strengths for the *A. thaliana* dataset (50 million flows). Distributions of signal strengths for two additional sequencing projects performed at UC Riverside are given as Supplementary Figure 1, available on-line at http://compbio.cs.ucr.edu/flat. The overlaps between Gaussians for different polynucleotide lengths are responsible for over-calling or under-calling the lengths of incorporated nucleotide runs.

When sequencing small RNA, the 454-provided software employs a maximum likelihood strategy to call a homopolynucleotide length, with cut-off point at $l \pm 0.5$ for polynucleotide length l. This results in, for example, flows (T,2.52) and (T,3.48) both being called as `TTT` even though the proximity of the cut-off points indicates that the former one may have in fact come from `TT` and the latter one from `TTTT`. This could be alleviated to a degree by allowing approximate matches, where insertions or deletions would address under-calling and over-calling. However, without the knowledge of the underlying signal strengths any insertion or deletion would be arbitrary.

Also, according to the 454 procedure, a flow with signal intensity 0.49 will be treated as a negative, even though it is very close to the cut-off point for a positive flow. Consider the following example: sequence of flows (C,0.92)(G,0.34)(T,0.49)(A,0.32)(C,0.98) will be called as `CNC` and all information about which nucleotide was most likely to be in the middle will be lost.

These examples illustrate the intuition behind our approach: we use signal strengths to estimate probabilities of different lengths of homopolymer runs that may have induced the signal. The target genome conditions the probabilities, and the most probable explanations are returned as potential matches. The following section formally introduces the notion of *flowgram matching*.

3. Flowgram Matching

Let F be a flowgram obtained by pyrosequencing a genomic fragment originating from genome Γ, and let G be a flowspace representation of Γ derived by run-length encoding (RLE)[a] of Γ and padding the result with appropriate zero-length negative flows in a manner which simulates flowing nucleotides in the T,A,C,G order, as illustrated in Supplementary Figure 2. Let flowgram $F = \{(b_0, f_0), (b_1, f_1), \ldots, (b_{m-1}, f_{m-1})\}$ be a sequence of m flows, where b_i is the nucleotide flowed and f_i is the resulting signal strength. Let n be the length of G. Under the assumption that the occurrences of lengths of homopolynucleotide runs are independent events, the probability that a flowgram F matches a segment in G starting at position k can be expressed as

$$P(F \sim G_{k..k+m-1}) = \prod_{i=0}^{m-1} P_{b_i}(L = g_{k+i} | S = f_i) \qquad (1)$$

where L is a random variable denoting the length of the homopolynucleotide run in Γ, S is a random variable associated with the induced signal strength in the flowgram, and g_{k+i} is the length of the run at position i from the beginning of the match. For example, if the flowgram (A,0.98)(C,0.14)(G,1.86)(T,0.24)(A,3.12) is matched against `AGGAAA`, the run lengths for the genomic sequence are $g = \{1, 0, 2, 0, 3\}$, and the probability of matching would be $P_A(L = 1|S = 0.98) \cdot P_C(L = 0|S = 0.14) \cdot P_G(L = 2|S = 1.86) \cdot P_T(L = 0|S = 0.24) \cdot P_A(L = 3|S = 3.12)$. One of the benefits of casting the genome in flowspace is that a flowgram of length m will correspond to a segment of length m in G, whereas the corresponding segment

[a] We say that that a sequence w is a *run* of length k if $w = c^k$, where $c \in \{A, C, G, T\}$. In this case, the run-length encoding (RLE) of w is (c, k).

in Γ would have context-dependent length. Also, once the starting flows are aligned in terms of nucleotides, the remaining $m-1$ flows will be aligned as well. Using the Bayes' theorem we can rewrite equation (1) as

$$P(F \sim G_{k..k+m-1}) = \prod_{i=0}^{m-1} \frac{P_{b_i}(S = f_i | L = g_{k+i}) \cdot P_{b_i}(L = g_{k+i})}{P_{b_i}(S = f_i)} \tag{2}$$

where $P_{b_i}(L = g_{k+i})$ is the probability of observing a b_i homopolynucleotide of length g_{k+i} in Γ, and $P_{b_i}(S = f_i | L = g_{i+k})$ and $P_{b_i}(S = f_i)$ depend on the 454 sequencing model and can be estimated from the data through a combination of the called sequences and the underlying flowgrams (see Section 7).

If we assume a *null* model where homopolynucleotide runs are assigned the probabilities obtained by counting their frequencies in G, the log-odds score of the match is

$$Score(F \sim G_k) = \log \frac{P(F \sim G_{k+m-1})}{P_{null}(G_{k..k+m-1})} = \log \frac{\prod_{i=0}^{m-1} P_{b_i}(L = g_{k+i} | S = f_i)}{\prod_{i=0}^{m-1} P_{b_i}(L = g_{k+i})}$$

Rewriting the numerator using the Bayes' theorem allows us to cast flowgram matching as an analog of *profile matching* (see e.g. [2,5,19,21]), with the scoring matrix M defined as

$$M_{i,j} = \log P_{b_i}(S = f_i | L = j) - \log P_{b_i}(S = f_i)$$

The log-odds score can then be expressed as a sum of the matrix entries

$$Score(F \sim G_k) = \sum_{i=0}^{m-1} M_{i,j} \tag{3}$$

A brute-force approach for matching a flowgram F would be to align F with all m flow long segments in G and report the best alignments. This algorithm runs in $O(mn)$ time per flowgram. With typical sequence library sizes in the hundreds of thousands, flowgrams up to 100 bp and genomes in the order of billion bp, this approach is computationally not feasible.

4. Enhanced Suffix Arrays

Recently, Beckstette *et al.*[2] introduced the *enhanced suffix array* (ESA), an index structure for efficient matching of position specific scoring matrices (PSSM) against a sequence database. While providing the same functionality as suffix trees[1], enhanced suffix arrays require less memory and once precomputed they can be easily stored into a file or loaded from a file into main memory. An enhanced suffix array can be constructed in $O(n)$ time[2,9].

We employ enhanced suffix arrays to index the database of genomic sequences, with two adjustments. An ESA indexes the search space of positive flows, in the order determined by the underlying genome. To provide the view of the genomic sequences as observed by the 454 sequencer, positive flows are padded with intermediate dummy negative flows, as illustrated in Supplementary Figure 2 (available at http://compbio.cs.ucr.edu/flat). This padding does not interfere with searching for the complement of the flowgram because `CGTA`, the reverse complement of the order `TACG`, is a cyclic permutation of the original order with offset 2. Consequently the reverse complements of the dummy flows would exactly match the dummy flows inserted if the reverse complement of the RNA fragment was sequenced.

When a flowgram is being aligned along the "branches" of the suffix array, the branches are run-length encoded and negative flows are inserted where appropriate. This amounts to on-the-fly branch by branch flowspace encoding of the underlying sequence database, without sacrificing the compactness of the suffix array representation. The score of the alignment is calculated using equation (3).

The first adjustment solves the problem of intermediate negative flows. However, it can happen that the flowgram corresponding to the RNA fragment starts or ends with one or more negative flows. The second adjustment creates variants of the indexed database subsequence, where combinations of starting and ending negative flows are allowed, as illustrated in Supplementary Figure 3 (available at http://compbio.cs.ucr.edu/flat).

5. Lookahead Scoring

Flowgram matching using the index structure described in the previous section can be stopped early if the alignment does not appear to be promising. More precisely, given a threshold score t which warrants a good match of the flowgram against the sequence database, and the maximum possible score for each flow, we can discard low-scoring matches early by establishing intermediate score thresholds th_i. The final threshold for the whole flowgram, th_m, is equal to t, and the intermediate thresholds are given by $th_{i-1} = th_i - max_j(M_{i,j})$. This method, termed *lookahead scoring*, was introduced in Wu *et al.*[21], and was combined with the enhanced suffix arrays in Beckstette *et al.*[2]. The threshold score t can be estimated using statistical significance of the match (see Section 6).

Although lookahead scoring gives the same asymptotic worst case running time, in practice, it results in significant speed-ups by pruning the subtrees which start with low-scoring prefixes in the database.

6. Statistical Significance of Scores

Intuitively, a higher raw score obtained by matching a flowgram F against a segment of the sequence database should correspond to a higher likelihood that F was generated by pyrosequencing the matched genomic segment. One way to associate a probability value p with a given raw score is to compute the cumulative distribution function (cdf) over the range of scores that can be obtained by matching F against a flowspace-encoded random genomic segment. Formally, if T is a random variable denoting the score, t is the observed score, and f_T is the probability mass function, the p-value p associated with t is $P(T \geq t) = \sum_{i \geq t} f_T(i)$. The probability mass function can be computed using a dynamic programming method described in Staden *et al.*[18] and Wu *et al.*[21], using a profile matching recurrence relation adjusted for the task of flowgram matching:

$$f_T^i(t) = \sum_l f_T^{i-1}(t - M_{i,l}) \cdot P_{b_i}(L = l)$$

An improvement to this method, described in Beckstette *et al.*[2], is based on the observation that it is not necessary to compute the whole cdf, but only the part of the cdf for scores higher than or equal to the observed score t. Values of the probability mass function are computed in decreasing order of achievable scores, until threshold score t for which the sum of probabilities is greater than p is reached. Modified recurrence relation is as follows:

$$f_T^i(t) = \sum_{l \in \{l | M_{i,l} \geq max_i - d\}} f_T^{i-1}(t - d - M_{i,l}) \cdot P_{b_i}(L = l)$$

For a user specified statistical significance threshold p, this method gives a score threshold t which can be used to perform statistical *significance filtering* of the matches. The threshold score t can be used in conjunction with previously described lookahead scoring to speed up the search. In addition, a correspondence between obtained scores and p-values allows for indirect comparison between scores obtained by matching different flowgrams across different sequence databases.

The expected number of matches in a random sequence database of size n, generally known as the E-value, can be calculated as $p \cdot n$.

7. Parameter Estimation for Probability Distributions

The output of a 454 sequencer is given as a set of three files: (1) a collection of called sequences in FASTA format, (2) accompanying per-called base quality scores which are a function of the observed signal and the conditional distributions of signal strengths[11], and (3) the raw flowgram files. The 454 flowgrams start with the first observed positive flow, and signals are reported with 0.01 granularity.

We combined (1) and (3) to obtain four sets (one per nucleotide) of conditional distributions for different called lengths. Using the maximum likelihood method, we estimated means and standard deviations of the normal distributions for positive flows. Only the conditionals for $l \leq 4$ were used, as data for higher lengths becomes noisy (see Figure 1 and Supplementary Figure 1). We fit a line through the observed values for σ, and use this as an estimate for σ_l.

The signals for the negative flows are distributed according to a distribution which resembles the log-normal, but which exhibits a markedly different behavior in the tails. Most notably, as the signal intensities approach 0, the number of observed signals should also approach 0, and the observed frequencies are significantly higher. Because we have a large number of negative flow signals (no less than 3.5 million flows per library per nucleotide), we decided to use histograms for the distribution of negative flow signals on the $[0, 0.5]$ interval, and extrapolate it using an exponential function on $(0.5, \infty)$.

8. Experiments

We coded a prototype implementation of our method in C++; we called this program FLAT (for FLowgram Alignment Tool). The suffix array index was built using `mkvtree`[10].

We compared FLAT to two methods which could be used for matching small RNA against the target genome: (1) exact matching using a suffix array and (2) BLAST(version 2.2.15) with parameters optimized for finding short, near identical oligonucleotide matches (seed word size 7, E-value cutoff 1000). FLAT is matching flowgrams, whereas the other two methods are matching sequences obtained by base calling the same flowgrams which were returned by 454 Life Sciences. In all three cases, adaptors enclosing the sampled small RNA inserts were trimmed before the search. The flowgram dataset was obtained by pyrosequencing four small RNA libraries constructed from *A. thaliana* plants exposed to abiotic stress conditions: A) cold (61,685 raw flowgrams), B) drought and ABA (74,432), C) NaCl and copper (51,894), and D) heat and UV light (33,320). Reference *A. thaliana* sequences were downloaded from TAIR[15]. We matched small RNA against whole chromosome sequences as well as AGI Transcripts (cDNA, consisting of exons and UTRs) datasets, because small RNA could have been sampled before or after splicing.

All three methods were run on a 64 bit 1,594 MHz Intel Xeon processor. Searching for matches of the first library against *Arabidopsis* chromosome 1 (30.4 million bp), for example, took 6 hours 46 minutes for FLAT, 2 hours 9 minutes for BLAST and 14 minutes for exact matching using highly efficient suffix array implementation.

Results. The number of matches returned by the three methods are summarized in Figure 2. Relatively small numbers of matches compared to the sizes of the libraries is due to the high percentage (59.3-62.8%) of raw flowgrams which were shorter than 18bp once the adaptor sequences were trimmed, and hence too short to belong to a known class of small RNA.

Exact matching is the most stringent and most reliable method of the three; however, due to the number of short inserts which cannot be interpreted as small RNA candidates and due to the nature of the sequence base calling method, only a small fraction (16.0-23.9%) of the original flowgrams match the target genome.

Allowing probabilistic matching using FLAT or tolerating insertions and deletions using BLAST increases the number of matches at the expense of reliability. It is difficult to compare FLAT and BLAST directly, as they were designed with different goals in mind; furthermore, an approximate BLAST match has no grounding in the underlying flowgram signals and unlike FLAT with respect to this is completely arbitrary. However, the number of matches they return and the number of returned matches which appear also in the exactly matched dataset, given as a function of the E-value, provide an intuition about FLAT's behavior. For E-value of 10^{-3}, which in our experiments provided the best balance between the number of matches and false positives, in all four libraries, FLAT consistently returns 98.0% to 98.4% of the exact matches, while returning additional 26.4% to 28.8% matches not found exactly. At higher E-values, the relaxed matching conditions mean that less probable matches would also be included in the output.

BLAST returns nearly all exact matches at E-value 10^{-2}, at which point it returns the number of additional matches comparable to FLAT for the

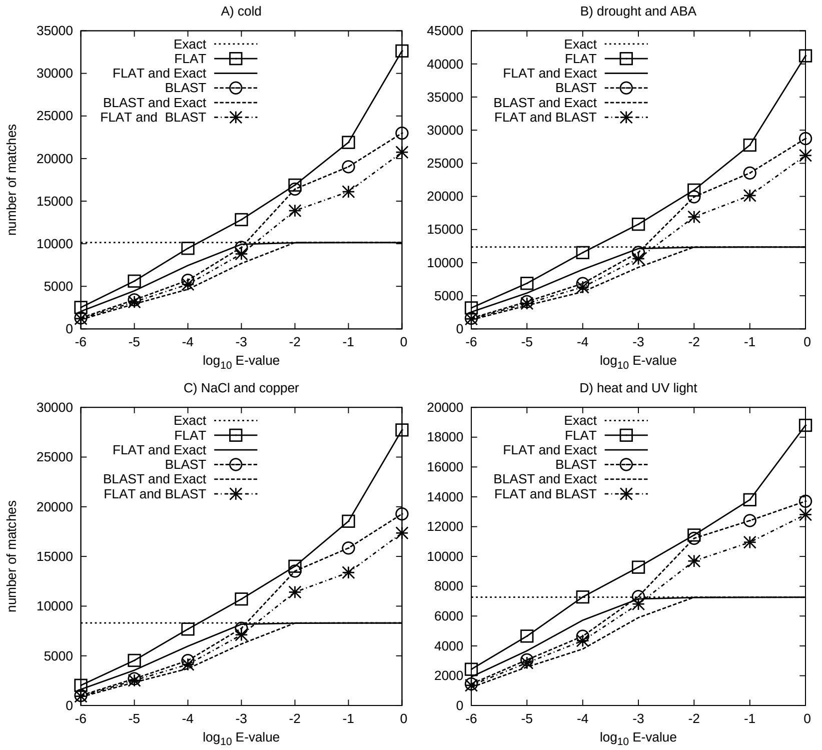

Figure 2. Comparison between the number of matches found for the four stress-induced *A. thaliana* small RNA libraries: A) cold, B) drought and AB, C) NaCl and copper, and D) heat and UV light.

same E-value. It is of interest to note that even though the number of matches is similar, not all of them are found by both methods (the dot-dashed line with star markers in Figure 2).

To illustrate some of the additional matches returned by FLAT and missed by BLAST, consider the flowgram (C,2.06)(G,1.02)(T 0.23)(A 1.53)(C 2.22)(G 0.23)(T 1.99) (A 1.13)(C 0.33)(G 0.96)(T 0.39)(A 0.19)(C 0.96)(G 0.19)(T 0.93)(A 0.10) (C 1.15)(G 0.10)(T 0.26)(A 0.90)(C 0.18)(G 1.02)(T 2.03)(A 0.22)(C 0.12) (G 2.32) for which the maximum likelihood base-called sequence is `CCGAACCTTAGCTCAGTTGG`, which does not occur in the genome. However, if we allow the first `A` flow with the intensity 1.53 to come from `A` and not `AA` we get an alternative base-called sequence `CCGACCTTAGCTCAGTTGG`, which occurs in a number of tRNA genes.

9. Discussion

In this paper, we described a procedure which makes use of the flow signal distribution model to efficiently match small RNA flowgrams against the target genome in a probabilistic framework. Depending on the user-specified statistical significance threshold, additional matches missed by exact matching of the called flowgram sequences are returned.

In principle, evaluating the biological significance as a function of the statistical significance is a challenging task. When analyzing the additional matches, most would agree that calling a flow (A,1.53) as either `A` or `AA` would make sense. However, calling a flow (A,0.20) as `A`, however less probable, is still possible under the model provided in Marguiles *et al.*[11], if less probable matches are allowed by increasing the threshold statistical significance. FLAT provides several output and filtering options which allow the user to focus on the analysis of the non-exact matches or their subset. Most promising matches, in terms of their functional analysis after the tentative genomic loci have been determined, would require additional post-processing and ultimately biological verification.

References

1. M.I. Abouelhoda et al. *Journal of Discrete Algorithms*, 2:53–86, 2004.
2. M. Beckstette et al. *BMC Bioinformatics*, 7:389, 2006.
3. F. Chen et al. In *PAG XIV Conference*, January 2006.
4. A. Fire et al. *Nature*, 391:806–11, 1998.
5. R. Fuchs. *Comput. Appl. Biosci.*, 9:587–91, 1994.
6. B. Gharizadeh et al. *Electrophoresis*, 27(15):3042–7, 2006.
7. A. Girard et al. *Nature*, 442:199–202, 2006.
8. S.M. Goldberg et al. *Proc. Natl. Acad. Sci. USA*, 203(30):11240–5, 2006.
9. J. Karkkainen and P. Sanders. In *13th ICALP*, pages 943–55, 2003.
10. S. Kurtz. http://www.vmatch.de.
11. M. Margulies et al. *Nature*, 435(7075):376–80, 2005.
12. M.J. Moore et al. *BMC Plant Biol.*, 6(17), 2006.
13. C. D. Novina and P. A. Sharp. *Nature*, 430:161–4, 2004.
14. R. Rajagopalan et al. *Genes Dev.*, 20(24):3407–25, 2006.
15. S. Rhee et al. *Nucleic Acids Research*, 31(1):224–8, 2003.
16. M. Ronaghi et al. *Anal Biochem*, 242(1):84–9, 1996.
17. F. Sanger et al. *Proc. Natl. Acad. Sci. USA*, 74:5463–7, 1977.
18. R. Staden. *Comput. Applic. Biosci.*, 5:193–211, 1989.
19. J.C. Wallace and S. Henikoff. *Comput. Applic. Biosci.*, 8:249–254, 1992.
20. T. Wicker et al. *BMC Genomics*, 7(275), 2006.
21. T. Wu et al. *Bioinformatics*, 16(3):233–44, 2000.