JOURNAL OF COMPUTATIONAL BIOLOGY
Volume 16, Number 6, 2009

Pp. 845–858
DOI: 10.1089/cmb.2008.0180

Statistical Models for Local Occurrences of RNA Structures

NIELS RICHARD HANSEN

ABSTRACT

We develop in this article the necessary statistical theory for computing, for instance, E-values when searching long sequences for the occurrences of local RNA-structures. We show in particular how the theory can be used for estimating scoring parameters with the purpose of optimizing the discriminative performance of the algorithm. The results are implemented in the program *StemSearch*, which can search for stem loop structures that are formed by, for example, micro RNA precursors. We illustrate the use of the estimation method in practice by considering three miRNA target datasets from *Human*, *Arabidopsis*, and *C. elegans* and by optimizing three penalty parameters in *StemSearch*. We show that the optimization can improve the discriminative performance considerably when using a first order Markov model as null-distribution. Finally, we compare the output from *StemSearch* with that of *RNALfold*, and we discuss some notable differences that are primarily due to fundamental differences in the choice of parameters.

Key words: discriminative estimation, RNA structures, statistics, stem loops, stemsearch.

1. INTRODUCTION

Almost any algorithm used in biological sequence analysis relies on a choice of parameters. In practice, the parameters are set by using combinations of experience, expert knowledge and estimation based on data. The popular hidden Markov models used, for instance, for protein family profiles provide examples where parameters are typically estimated using maximum-likelihood. For local alignments, a good choice of parameters is a little more delicate—in particular, when it comes to the choice of penalty parameters. We know, however, that to obtain truly local alignments in, for example, BLAST, one needs to choose gap open and gap extension parameters sufficiently large and negative.

RNA folding (the computational prediction of the structure of RNA molecules from its sequence) is another important algorithmic problem. Single molecule predictions are typically based on minimizing the free energy in a more or less physically realistic model. In most cases the focus is on predicting the secondary structure only and not the entire three dimensional structure with atomic coordinates. The secondary structure is the combinatorial description of the nucleotides that form hydrogen bonds. Each possible combination is assigned a free energy, which is then minimized using dynamic programming as implemented in,

Department of Mathematical Sciences, University of Copenhagen, Copenhagen, Denmark.

for example, *RNAfold* (Hofacker et al., 1994). The parameters used are based on physical binding energies between the nucleotides.

In recent years, there has been a large interest in small non-coding RNA molecules like micro RNA (miRNA) and the computational search for such molecules. One characteristic of miRNA's—or rather the slightly larger molecules called the miRNA precursors—is that they form stem loop structures (Zhang et al., 2007). Therefore, a computational search may among other things search for bits and pieces of the genome that are capable of forming stem loops. The challenge here is then both algorithmic as well as statistical. How should we computationally fold a long sequence locally and how do we distinguish "random folds" from true folds—or more specifically in the case of miRNA precursors, how do we distinguish random stem loops from true stem loops?

One procedure for computationally folding entire genomes locally is to choose a fixed window size L and then, while scanning along the genome, fold subsequences of length L. An efficient implementation of this procedure is *RNALfold* (Hofacker et al., 2004). The implementation does not, however, deal with the statistical aspect. The purpose of this paper is to develop the statistical theory for local folding and illustrate how it can be applied in the search for local RNA structures in genomes. In particular we show how to use the theory to tune some of the parameters used for the algorithm for optimal performance. The development is partly based on the theoretical results in Hansen (2007). For the results to be of practical use, we have developed the program, *StemSearch*, that searches long sequences for the occurrence of local stem loops and reports a list of essentially different local stem loops together with a statistical evaluation in terms of an E-value. We discuss how to define "essentially different" in Section 2.3.

It is evident that the computational search for small regulatory RNA molecules like miRNA has intensified over the past 5–7 years. For recent reviews of hitherto used computational techniques for miRNA discovery, see Lindow and Gorodkin (2007) and Yoon and Micheli (2006). The computational pipelines that are in use all combine several ideas such as evolutionary conservation of miRNA, matching of miRNA with mRNA-targets, secondary structure constraints, etc. Moreover, the order of such computational filters for singling out likely miRNA candidates are also to some extent interchangeable. One intended use of *StemSearch* is therefore as a first filter in a computational pipeline for miRNA searching.

As such a filter, it is of utmost importance to reduce the number of false positives while retaining a reasonable specificity. The most important contribution of this paper is therefore to show how the statistical theory not only offers an computation of E-values but actually offers a natural method for *discriminative estimation* of parameters. That is, a method for selecting the scoring parameters in the algorithm that yields an optimal performance in terms of reducing the number of false positives. On the basis of the statistical setup, we derive a natural class of objective functions to optimize and as an illustration we apply this method to miRNA datasets from the three different organisms *Human*, *Arabidopsis*, and *C. elegans*.

Throughout the article, we consider algorithms that are based on maximizing a score function over the set of allowed secondary structures, and we then need to understand the distribution of the maximal score for a random sequence, that is, a sequence that does not contain parts that are supposed to fold and form structures. For the energy based methods the score would be *minus* the free energy. A theoretical analysis of the distribution of the score for a sequence of independent and identically distributed random nucleotides was carried out in Xiong and Waterman (1997). With S_n denoting the maximal score over all contiguous subsequences of a sequence of random nucleotides of length n it is shown in Xiong and Waterman (1997) that S_n scales *linearly* or *logarithmically* in n depending upon the scoring parameters used. If we let μ_n denote the expected score obtained for the entire length n sequence, it is, moreover, shown in Xiong and Waterman (1997) that μ_n/n converges to μ, say, for $n \to \infty$, and the S_n scales linearly if $\mu > 0$ and logarithmically of $\mu < 0$. What happens for $\mu = 0$ is not known precisely, though a qualified guess based on the theory of random walks is that the scaling is like $\sqrt{n}$. This divides the parameter space used for the algorithm into two parts called the linear and the logarithmic phase, respectively. The division is quite easy to interpret; if $\mu > 0$ the score grows linearly with the length of the sequence, thus the maximal score is obtained for a sequence of length roughly n, but for $\mu < 0$ the largest score is obtained for a truly local part of the sequence.

For parameters in the logarithmic phase, it is possible to develop a satisfying statistical theory, and we show how to optimize the choice of parameters over that part of the space. In addition, we make a comparison of our program *StemSearch* to the algorithmically similar program *RNALfold*, for which the parameters are in the linear phase of the parameter space. In particular, we illustrate the differences in the output from the two programs due to the use of parameters from the different phases.

To avoid making the algorithm and scoring parameters overly complicated—and to aid the presentation of the main ideas on the statistics—a relatively simple scoring scheme based on a first order Markov chain models is set up. This is also what is currently used in *StemSearch*. We derive the bulk of the scoring parameters from maximum-likelihood estimates of the Markov chain models, but three penalty parameters remain. We show how to use the discriminative estimation procedure for the estimation of these parameters. It may be the case that higher order models or other modifications can yield even better results in concrete cases. Also the choice of focusing on stem loops is restrictive. In practice more general secondary structures could be allowed if needed but stem loops or hairpins are still the primary structural elements. A review of a number of different hairpin RNA functions can be found in Svoboda and Cara (2006). The role of RNA structures and in particular RNA stem loops in splicing is reviewed in Buratti and Baralle (2004). In any case, the main idea of discriminative estimation can be applied if one can justify that the statistical theory extrapolates as well.

2. METHODS

We consider a sequence $\mathbf{x} = x_1, \ldots, x_n$ of length n of letters from the DNA/RNA alphabet, which we denote E. A (local) stem structure is a set of coordinate pairs

$$\mathbf{z} = \{(i_1, j_1), \ldots, (i_m, j_m)\}$$

fulfilling that $1 \leq i_m < \cdots < i_1 < j_1 < \cdots < j_m \leq n$. In the implementation *StemSearch*, we require $j_1 - i_1 \geq 3$. Let $\mathcal{Z}$ denote the set of such coordinate pairs, that is, the set of stem structures. With $S(\mathbf{z},\mathbf{x})$ the score of the stem given by $\mathbf{z}$ for the sequence $\mathbf{x}$, we search for the maximal score over the set of stems,

$$S(\mathbf{x}) = \max_{\mathbf{z} \in \mathcal{Z}} S(\mathbf{z}, \mathbf{x})$$

and let

$$\hat{\mathbf{z}} = \arg\ \max_{\mathbf{z} \in \mathcal{Z}} S(\mathbf{z}, \mathbf{x})$$

denote the stem where the optimal score is attained. It may not be unique in general.

2.1. Stack scoring and loop penalties

We consider a rather simple way of scoring the structure, where the main part consists of a score for stacks of hydrogen bonded nucleotides. Thus, for each quadruple of nucleotides x,y,v,w we have a score $s(xy,vw)$ interpreted as the score for forming the hydrogen bond between v and w when x,y,v,w are organized as shown on Figure 1.

The sum of the stack scores in the stem $\mathbf{z}$ is denoted

$$S_0(\mathbf{z}, \mathbf{x}) = \sum_{(i,j) \in \mathbf{z}} s(x_{i+1}x_{j-1}, x_i x_j),$$

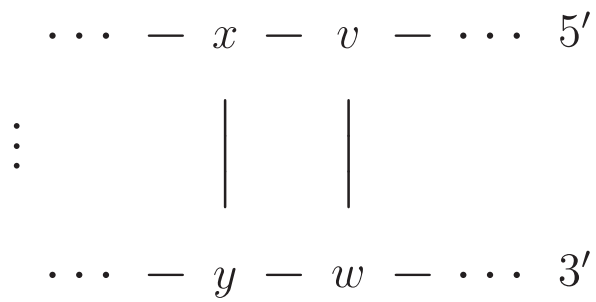

FIG. 1. The score $s(xy, vw)$ quantifies the formation of a hydrogen bond between v and w when "stacked" on top of xy—reading from the hairpin-loop and outwards.

which is the contribution to the total score of the stem $\mathbf{z}$ for the sequence $\mathbf{x}$ that comes from the stacks formed. We will usually then add a penalty term depending on the number and length of internal loops and bulges and on the length of the hairpin loop. We will only consider sequence independent penalty terms, and we will treat internal loops and bulges identically. Let $r_{\mathbf{z}}$ and $l_{\mathbf{z}}$ denote the total number and total length respectively of the loops and bulges for $\mathbf{z}$, and let $h_{\mathbf{z}}$ denote the length of the hairpin loop. Then define the total score of the structure $\mathbf{z}$ for the sequence $\mathbf{x}$ to be

$$S(\mathbf{z}, \mathbf{x}) = S_0(\mathbf{z}, \mathbf{x}) + (\alpha - \beta) r_{\mathbf{z}} + \beta l_{\mathbf{z}} + g(h_{\mathbf{z}})$$

where $g : \mathbb{N}_0 \to (-\infty, 0]$ is a function and α, $\beta < 0$ are some constants.

2.2. Log-likelihood ratio scores based on Markov chains

The stack scores as well as the loop penalties as introduced above could be based on energy considerations as in the common RNA–structure prediction programs. As an alternative, we present here a way to compute stack scores based on a first order Markov chain model. Essentially, we suggest the usual minus-log-likelihood ratio scores based on the log-probability-ratio of a null model of independent nucleotides versus a model for the occurrence of independent hydrogen bonded pairs. However, we want to retain the stacking idea that the occurrence of a bonding pair may be affected by the pair it is stacked upon. This naturally suggests that we use a first order Markov chain models. This will complicate the presentation slightly as one strand in a stem is in the reverse direction of the Markov chain under the null.

As a null model for the x's we consider a first order Markov chain with $P = (P(x, y))_{x, y \in E}$ the matrix of transition probabilities. We will assume that $P(x,y) > 0$ for all $x,y \in E$, which guarantees the existence of an invariant probability distribution $\pi = (\pi(x))_{x \in E}$ fulfilling that

$$\sum_{x \in E} \pi(x) P(x, y) = \pi(y).$$

The matrix P provides a model of the transitions in the sequence from the 5′ to the 3′ end. The transitions in the other direction, from the 3′ end to the 5′ end, are given by the reversed transition probabilities

$$\overleftarrow{P}(x, y) = \frac{\pi(y) P(y, x)}{\pi(x)}.$$

Let $(Q(xy,vw))_{x,y,v,w \in E}$ denote a matrix of transition probabilities on $E \times E$. We imagine that for a given stem $\mathbf{z}$ the letter pairs in a stack are build sequentially from the inner most hydrogen bonded pair of letters by drawing conditionally on that pair another pair of hydrogen bonded letters according to Q. Thus, the matrix Q represents the transition probabilities for building up a stack as a first order Markov chain from the hairpin-loop and outwards. In this framework a natural choice of $s(xy,vw)$ is the log-likelihood ratio

$$s(xy, vw) = \log \frac{Q(xy, vw)}{\overleftarrow{P}(x, v) P(y, w)}. \tag{1}$$

Making this choice of scores the total score, $S_0(\mathbf{z}, \mathbf{x})$, will be the minus-log-likelihood ratio of the Markov chain model of $\mathbf{x}$ given by P (the null model) against the alternative where all paired letters in $\mathbf{z}$ are drawn conditionally, as described above, according to Q and where the remaining letters are drawn according to P.

In the implementation of *StemSearch*, as discussed below, we have used the log-likelihood ratio approach to compute the stack scores $s(xy, vw)$ with P estimated from genome data using maximum likelihood and Q likewise estimated using maximum likelihood from (predicted) miRNA stem structures. That is, the number of stack transitions are counted in miRNA hairpins, that were predicted by *RNAfold*.

2.3. Algorithms and implementation

A standard dynamic programming algorithm can be implemented for finding the maximal score using the score function S defined above. As a starting point, we define an upper triangular score matrix, V, with $V_{i,j}$ for $i + 3 \leq j$ being the maximal score when (i, j) closes the stem. In addition, a loop penalty matrix W is needed and the recursions read

$$V_{i,j} = \max \begin{cases} V_{i+1,j-1} + s(x_{i+1}x_{j-1}, x_i x_j) \\ W_{i+1,j-1} + s(x_{i+1}x_{j-1}, x_i x_j) \\ g(i-j+1) \end{cases}$$

and

$$W_{i,j} = \max \begin{cases} W_{i+1,j} + \beta \\ W_{i,j-1} + \beta \\ V_{i+1,j} + \alpha \\ V_{i,j-1} + \alpha \\ g(i-j+1) \end{cases}$$

We will typically band limit V (and W), which effectively means that we restrict attention to stems contained in a sliding window of size L. The optimal stem is found by a subsequent traceback. The traceback starts in the V matrix at the position where the maximal score is found and the traceback terminates when it reaches an entry (i, j) in V with $V_{i,j} = g(i-j+1)$.

The time complexity as well as the memory complexity for the band-limited algorithm is $O(nL)$. In the implementation *StemSearch* a memory efficient algorithm with memory complexity $O(L^2)$ is employed. Consequently tracebacks are carried out "on the fly."

Typically, one would be interested in finding essentially different, suboptimal stems in addition to the highest scoring local stem. A version of the so-called island method, as presented in Altschul et al. (2001) for local alignments, can solve this problem. One defines a matrix I such that $I_{i,j}$ points to the entry in V where the traceback from position (i,j) in V will terminate. All entries in I that point to the same terminating entry are said to belong to the same island. Note that this provides a partition of the matrix V such that entries located in different islands correspond to trace-backs of stems that share no pairs. The maximal score in an island is called the island score, and the island score together with the corresponding stem is regarded as a representative for the island. These scores and stems are referred to as the island scores, and due to the non-sharing of paired letters the islands and island scores may arguably be regarded as essentially different suboptimal stems. We refer to Altschul et al. (2001) for a thorough discussion of the island method in the alignment setup.

Throughout we will only consider the use of *StemSearch* with a penalty on the size of the loop, and the statistical theory below is developed from this point of view. We can also imagine using *StemSearch* to locate putative stems, whose arms are separated by a very large loop-region. Whether the loop-region is actually a loop or form some other structure plays no role. To locate such non-local stems we would of course need to take the bandwidth sufficiently large, but to find non-local stems we should also choose the loop length penalty function g to be constantly equal 0. This feature is implemented in *StemSearch* together with a corresponding statistical evaluation, though this is not discussed further in this paper.

2.4 Statistics

We present here a null model of local stem scores when $\mathbf{x}$ is a realization of a "random DNA sequence" that does not contain structural elements. For the suggested statistical model of the scores to be valid we do not need to assume that the letters in the random DNA sequence are i.i.d. or form a Markov chain, say. But we will use a Markov chain model in several concrete computations. The null model we consider specifies that the number of islands with a score exceeding a given threshold $t > 0$ follows a Poisson distribution and that the excesses above t of the island scores are independent and exponentially distributed with intensity parameter $\lambda > 0$. We refer to Hansen (2007) for some theoretical justification.

It is necessary to assume that $g \to -\infty$ "sufficiently fast". In the case $\alpha = -\infty$, it is shown in Hansen (2007) that if

$$\sum_{k=1}^{\infty} \exp(\lambda g(k)) < \infty \tag{2}$$

then $g \to -\infty$ sufficiently fast. Note that this condition is always fulfilled if $g(n) \sim \gamma n$ for some $\gamma < 0$. We expect that this condition is also sufficient when $\alpha > -\infty$.

We will also make the additional assumption that the scores $s(\mathbf{z}, \mathbf{x})$ are not lattice valued, that is, they are not a fixed multiple of integers. This is because we will use the exponential distribution. If the scores are lattice valued, a similar treatment is possible using the geometric distribution (Altschul et al., 2001).

We let I_t for $t \geq 0$ denote the set of (declumped) stems in $\mathcal{Z}$ produced by the island method with a score exceeding level t. Then with $N_t = |I_t|$ the number of excesses we use the Poisson distribution to model N_t with the mean

$$\mathbb{E}(N_t) = nK \exp(-\lambda t) \tag{3}$$

for two parameters λ, $K > 0$. The excesses,

$$S(\mathbf{z}, \mathbf{x}) - t; \quad \mathbf{z} \in I_t,$$

are assumed independent and identically exponentially distributed with mean λ^{-1}. The exponential excess and the Poisson mean value structure are seen to be in concordance. If we take N_{t_0}, say, to be Poisson distributed with mean $nK \exp(-\lambda t_0)$ and let $S_1, \ldots, S_{N_{t_0}}$ be i.i.d. exponentially distributed with mean λ^{-1} then for any $t > t_0$

$$N_t = \sum_{i=1}^{N_{t_0}} 1(S_i > t - t_0)$$

is Poisson distributed with mean $nK \exp(-\lambda t)$ and the excesses above $t - t_0$ are exponential with mean λ^{-1}.

These model assumptions are only supposed to hold approximately for sufficiently large n, L and t, which are chosen such that $n \simeq \exp(\lambda t)$ and $\log(n) = o(L)$. It is under such assumptions that the theoretical results in Hansen (2007) are obtained. In practice this reminds us that there are limitations to the applicability of the model, and what the nature of these limitations is. It will also be reflected in the estimators considered below, where we need to take t sufficiently large to get reliable estimators.

The parameters λ and K do not depend upon n or L but capture the role of the scoring parameters, as given by the stack scores s, the internal loop and bulges penalty parameters α, β, and the hairpin loop penalty function g, together with the distribution of the DNA-sequence. We introduce the normalized scores

$$\overline{S}(\mathbf{z}, \mathbf{x}) = \lambda S(\mathbf{z}, \mathbf{x}) - \log K$$

for $\mathbf{z} \in \mathcal{Z}$, and we call $\overline{S}(\mathbf{z}, \mathbf{x})$ the *nat-scores*. We note that if we change α, say, then the distribution of the raw scores changes, and the parameters λ and K change as well, but the distribution of the resulting *nat-scores* does (approximately) not change. Thus, we get comparability in the null distribution across different choices of scoring and penalty parameters.

2.5. *Optimization of penalty parameters*

We introduce in this section a method for choosing penalty parameters with the purpose of minimizing the expected number of false positives. We take $g(n) = \gamma(n-2)$, $n \geq 2$, for $\gamma < 0$. Let $v = (\alpha, \beta, \gamma)$ denote the vector of penalty parameters and, to emphasize the influence of v on the optimal score, let

$$S_v(\mathbf{x}) = \max_{\mathbf{z} \in \mathcal{Z}} \{S_0(\mathbf{z}, \mathbf{x}) + (\alpha - \beta) r_{\mathbf{z}} + \beta l_{\mathbf{z}} + \gamma(h_{\mathbf{z}} - 2)\} \tag{4}$$

Let Y denote a random sequence, which we think of as the RNA-sequence we search for with a preference for forming a stem-loop structure. With $q \in (0, 1)$ and with $t(v, q)$ denoting the $(1-q)$-quantile for the distribution of $S_v(Y)$, a local folding with threshold $t(v, q)$ has sensitivity[1] q. Fixing q to a desired level we would

[1]When a sequence x that form a stem is embedded in a longer sequence, the search procedure will locate a segment with a score $\geq S_v(x)$. We can have strict inequality due to the surroundings of the embedded sequence, which in principle can increase the sensitivity slightly.

like to minimize the expected number of false positives, which is the number $N_{t(v,q)}$ of islands with a score $> t(v, q)$ under the null model. Minimizing the expected number of false positives, as given by (3), over the parameter v gives

$$\begin{aligned} v_{\text{opt}}(q) &= \arg\min_v \mathbb{E}(N_{t(v,q)}) \\ &= \arg\min_v K(v)\exp(-\lambda(v)t(v,q)) \\ &= \arg\max_v \lambda(v)t(v,q) - \log K(v). \end{aligned} \tag{5}$$

We write $\lambda(v)$ and $K(v)$ to emphasize that these two parameters depend on the penalty parameter vector v. It is seen that minimizing the expected number of false positives with a fixed sensitivity q is equivalent to maximizing the $(1-q)$-quantile for the *nat-score* $\lambda(v)S_v(Y) - \log K(v)$. Alternatively, we argued that the distribution of the nat-scores *under the null model* is independent of the parameter v—the dependence is encoded entirely in $\lambda(v)$ and $K(v)$. Consequently we should aim at moving the entire distribution of $\lambda(v)S_v(Y) - \log K(v)$ towards the higher values. A simple quantitative measure to maximize is the expected *nat-score*. Another possibility, as given by (5), is to fix the sensitivity at the desired level q and maximize the $(1-q)$-quantile of the distribution of $\lambda(v)S_v(Y) - \log K(v)$. Note that neither of the resulting optimal v's depend upon the length n of the sequence we search in nor on the bandwidth L we have chosen.

2.6. Estimation of λ and K

Since we have no analytic expressions of how λ and K varies, we must rely on estimation—either from simulations or from real DNA-sequences. We can do that as follows. If $S_1,\ldots, S_m$ denote m island scores (obtained by fixing a suitably small threshold), we let

$$S_{1:m} < S_{2:m} < \cdots < S_{m-1:m} < S_{m:m}$$

denote the scores sorted in increasing order. The null model gives for suitable $t \geq 0$ a natural estimator

$$\tilde{\lambda} = \left(\frac{1}{N_t} \sum_{i=1}^{N_t} S_{(m-i+1):m} - t \right)^{-1} \tag{6}$$

where N_t is the number of excesses above t. An estimator of K can be derived from (3) by plugging in $\tilde{\lambda}$. As the model is only supposed to fit the data for sufficiently large t, n and L, the estimators are only going to work well for sufficiently large t, n and L. This method for estimating λ and K, known as Peaks Over a Threshold (POT) in the literature on extreme value statistics (de Haan and Ferreira, 2006; Embrechts et al., 1997), was also used in Altschul et al. (2001) in the context of local alignment. Since we want to plug in the estimates in (5), we will have to look at λ and K varying as functions of the scoring and/or penalty parameters. A slight variation of the estimators is then more appropriate. Instead of fixing the threshold t, we fix the number N_t of excesses. Thus for $N \in \{1, \ldots, M\}$ take the threshold to be $t = S_{(m-N):m}$. Note that this makes the threshold, t, and not N a random variable. It gives the alternative estimators

$$\hat{\lambda} = \left(\frac{1}{N} \sum_{i=1}^{N} S_{(m-i+1):m} - S_{(m-N):m} \right)^{-1} \tag{7}$$

and

$$\hat{K} = N \exp(\hat{\lambda} S_{(m-N):m}) / n. \tag{8}$$

A notable property of the latter estimator—which is not shared by the former for fixed t—is that it preserves the scaling properties of the parameters. The estimator $\hat{\lambda}$ is known in the literature as the *Hill estimator*[2] (de Haan and Ferreira, 2006).

[2]One mostly encounters the Hill estimator in the context of approximate power law distributions, which is what we get by taking the exponential of our scores.

For the estimation of the quantile $t(v, q)$, we consider a training dataset $Y_1, \ldots, Y_l$ consisting of l sequences that form stems, and we let $\hat{t}_l(v, q)$ denote the $(1-q)$-quantile of $S_v(Y_1), \ldots, S_v(Y_l)$. Plugging this estimator and the estimators (7) and (8) into (5) yields that the empirically optimal penalty parameter vector is given by

$$\begin{aligned}\hat{\upsilon}_{\text{opt}}(q) &= \arg\max_{\upsilon} \hat{\lambda}(\upsilon)(\hat{t}_l(\upsilon, q) - S_{m-N:m}) \\ &= \arg\max_{\upsilon} \frac{\hat{t}_l(\upsilon, q) - S_{m-N:m}}{\frac{1}{N}\sum_{i=1}^{N} S_{m-i+1:m} - S_{m-N:m}} \end{aligned} \tag{9}$$

This result is obtained by disregarding the terms in $\log \hat{K}(v)$ that do not depend upon v. The resulting objective function that we try to maximize has a quite intuitive interpretation—we try to maximize the excess of the q-quantile above the high level $S_{m-N:m}$ relative to the average excess above that level under the null model. Replacing the empirical quantile $\hat{t}_l(v, q)$ with the empirical mean of $S_v(Y_1), \ldots, S_v(Y_l)$ results in maximizing the empirical mean of the *nat-scores* instead.

2.7. Data analysis

We considered data from *C. elegans*, *Arabidopsis,* and *Human*. Initially, we estimated the first order transition probabilities, P, based on the entire genomes for each species. A dataset of miRNA precursors was generated from Rfam, (Griffiths-Jones et al., 2005), for each species with 114 miRNA precursors from *C. elegans*, 118 from *Arabidopsis*, and 462 from *Human*. The *Human* miRNA dataset was, furthermore, randomly split into a training set and a test set of equal size. The miRNA precursor secondary structure was predicted with *RNAfold* with default parameters. These datasets of predicted structures were used to estimate the matrix Q of transition probabilities in a stack. Finally, the log-likelihood ratio scores as given by (1) were computed for each species. These scores were used subsequently to investigate the statistical theory through simulations and to optimize over the penalty parameters. For *Human*, we use only the training set.

3. RESULTS

To investigate the statistical model for the stem scores with parameters in the logarithmic phase, we conducted a simulation study. For the simulation study we used the parameters estimated from the *Human* data, and we simulated a first order Markov chain with transition probabilities P. We choose $\alpha = -4.0$, $\beta = -2.1$ and $\gamma = -0.4$. This choice of penalty parameters comes from the optimization on the *Human* miRNA training data (Table 1). Taking as initial threshold $t = 4$ we did 1000 replications with $n = 5000$ and $L = 200$ resulting in a total of $m = 959075$ excesses. Figure 2 shows the empirical distribution of N_t taking $t = 15, 18, 21$ and 24. The variance-to-mean ratios for the empirical counts are 1.136 (p-value 0.0017), 1.085 (p-value 0.031), 1.087 (p-value 0.028), and 1.052 (p-value 0.12), respectively. The p-values are computed using the $\chi^2/999$-approximation with 999 degrees of freedom. In conclusion, the empirical counts are mildly over-dispersed as compared to the approximating Poisson distribution for the lower thresholds, but the

Table 1. Resulting Optimal Values of α, β, and γ, Where We, for Each of the Three Species, Maximized the Objective Function in (9) for $q = 0.2, 0.5, 0.8$ Based on the Species-Specific miRNA Training Datasets*

	Human				*Arabidopsis*				*C. elegans*			
	Sensitivity (q)				*Sensitivity (q)*				*Sensitivity (q)*			
	0.2	*0.5*	*0.8*	*Mean*	*0.2*	*0.5*	*0.8*	*Mean*	*0.2*	*0.5*	*0.8*	*Mean*
α	−4.6	−4.0	−3.7	−4.6	−3.7	−3.0	−2.8	−4.1	−2.9	−2.9	−2.8	−3.2
β	−2.3	−2.1	−1.9	−2.1	−1.0	−1.4	−1.4	−0.7	−1.6	−1.4	−1.4	−1.4
γ	−0.2	−0.4	−1.3	−0.2	−2.3	−2.2	−2.3	−2.0	−2.0	−1.6	−2.1	−1.5
λ	0.39	0.35	0.31	0.39	0.57	0.49	0.45	0.53	0.46	0.43	0.41	0.48
$\log(K)$	0.34	−0.31	−0.97	0.29	−0.77	−0.94	−0.98	−0.87	−0.92	−0.93	−1.04	−0.84

*The table also shows the optimal parameters computed with the use of the empirical mean in (9) instead. The resulting parameters λ and $\log(K)$ were estimated from simulations.

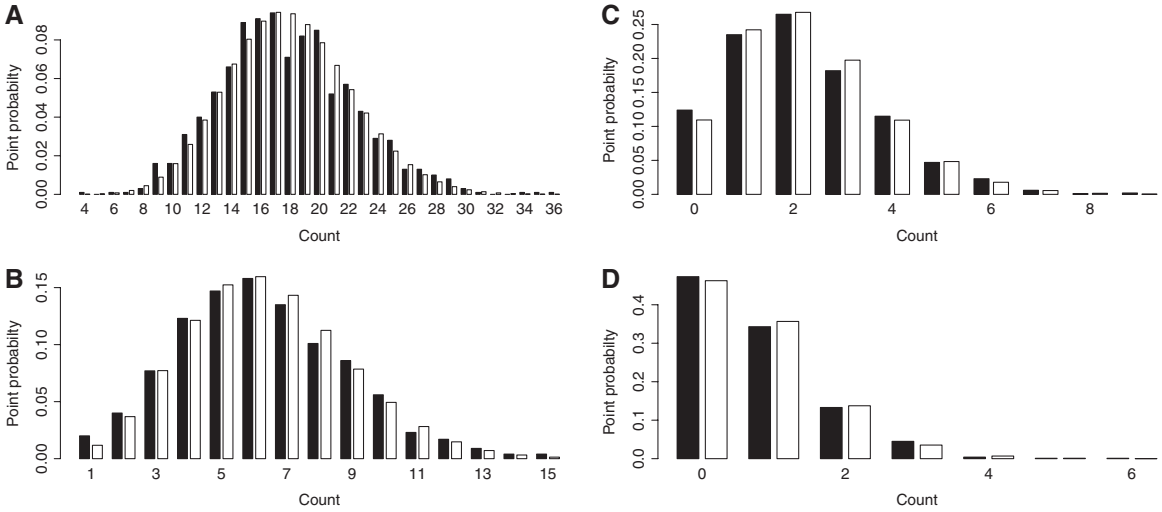

FIG. 2. Empirical point probabilities (black bar) and theoretical Poisson point probabilities (white bar) for a simulation study using *StemSearch* with $n = 5000$, $L = 200$ and $v = (-4, -2.1, -0.4)$ on sequences generated by a first order Markov chain with *Human* genome transition probabilities. We simulated 1000 sequences and used the thresholds $t = 15$ (A), $t = 18$ (B), $t = 21$ (C), and $t = 24$ (D). The variance-to-mean ratio for the empirical counts are 1.136 (A), 1.085 (B), 1.087 (C), and 1.052 (D), which is also seen as a mild over-dispersion of the empirical data compared to the Poisson distribution for the lowest thresholds.

approximation improves for the higher thresholds. The excesses (not shown) fit an exponential distribution well. Figure 3 shows the logarithm of the average number of excesses over the threshold as a function of the threshold for $t = 15, \ldots, 30$. The theory predicts that these points should fall on a straight line. Figure 3 includes the least squares fitted straight line with slope -0.352 and intercept 8.18. This corresponds to an estimate of λ as 0.352 and of $\log(K)$ as $8.18 - \log(5000) = -0.342$. Using the Hill estimator instead with $N = 2000$ provides the estimates $\hat{\lambda} = 0.350$ and $\log(\hat{K}) = -0.377$ (the corresponding random threshold is $S_{(959075-2000):959075} = 21.3$).

Next, we optimize over the penalty parameters (α, β, γ). For each species, we do a numerical maximization over v of the objective function in (9) for $q = 0.2, 0.5, 0.8$. The numerical method used was Nelder-Mead as implemented in the function optimize in R. Empirical studies of the objective function that was optimized showed that the function was well behaved with apparently no local maxima. For the computation of the empirical objective function, we compute $\hat{t}_l(v, q)$ for each species using the species specific miRNA dataset. The computation of $S_{(m-N):m}$ and the excesses were based on simulations using the species-specific transition probabilities, $n = 100000$, $L = 200$, and $N = 1500$. The results are shown in Table 1. Subsequently, the parameters $\lambda(v)$ and $K(v)$ were re-estimated using (7) and (8) with $n = 5 \times 10^6$, $L = 200$ and $N = 4000$. The results are also shown in Table 1.

Figure 4 shows the fraction of miRNA's with a *nat-score* exceeding the threshold t as a function of t for each species using the optimal penalty parameters for $q = 0.5$. That is, Figure 4 shows the empirical

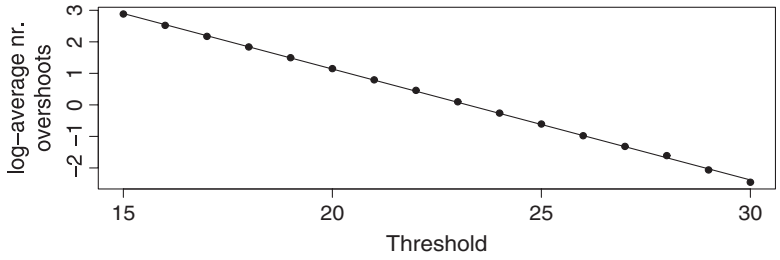

FIG. 3. The long-average number of excesses as a function of the threshold for a simulation study using *StemSearch* with $n = 5000$, $L = 200$, and $v = (-4, -2.1, -0.4)$ on sequences generated by a first order Markov chain with *Human* genome transition probabilities. The line is the least squares fit to the points with slope -0.352 and intercept 8.18.

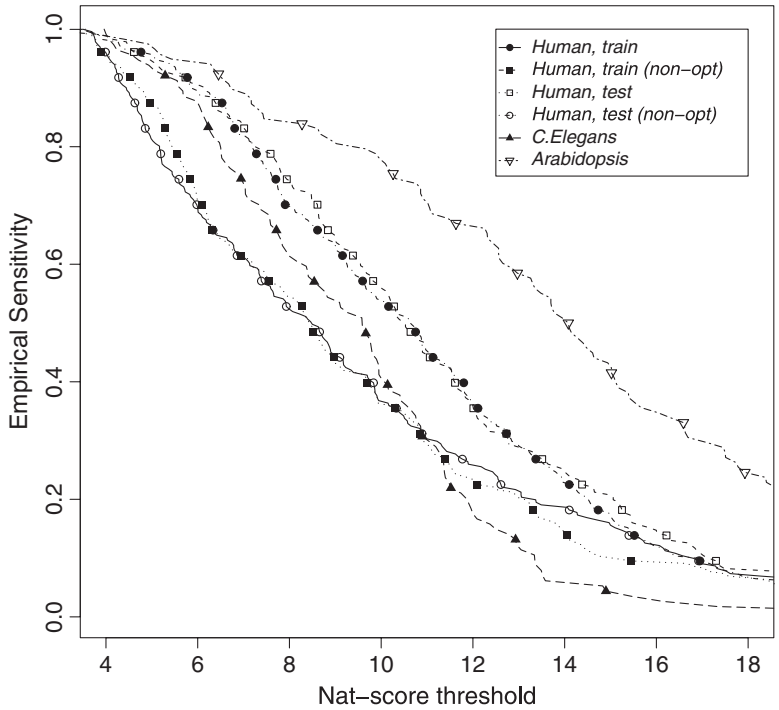

FIG. 4. The empirical sensitivity for each of the three species-specific miRNA training datasets using the penalty parameters from Table 1 with $q = 0.5$. For *Human*, the empirical sensitivity for the test dataset is also shown using both the optimized penalty parameters from Table 1 and the non-optimal choice of $(\alpha, \beta, \gamma) = (-4, -4, -2)$. If we aim for a sensitivity of 60%, say, the threshold should be changed from 9.5 when using the optimized penalty parameters to 7.1 for the suboptimal choice—considering the *Human* miRNA test dataset—resulting in an increase of the expected number of false positives with a factor of approximately 11.

sensitivity based on the miRNA training dataset as a function of the *nat-score* threshold. The figure has a close resemblance to a ROC-curve. In our context it seems, however, to be difficult to come up with a sensible definition of the specificity as there is no well defined total number of false cases—though the number of false positives for a given threshold is well defined. Thus, the use of ROC-curves to illustrate the relationship between sensitivity and specificity is not directly applicable. However, the plot in Figure 4 has an interpretation just like an ROC-curve. The interpretation of the figure is that the further to the north-east the curve is the better. A choice of q in the objective function (9) corresponds to moving the point on the curve giving sensitivity q as far to the right as possible. Note that the area under a curve in Figure 4 equals the average *nat-score* for the miRNA dataset with the given penalty parameters. For *Human*, Figure 4 also shows the empirical sensitivity for the test dataset and for the test dataset with one choice of non-optimal penalty parameters. We observe that the empirical sensitivity curves for the *Human* test and train dataset follow each other very closely, but that there is a notable difference between the curves from the optimal choice of parameters to the non-optimal choice.

We next illustrate the differences in the output from the free energy based *RNALfold* program and *StemSearch*. A simulation study (not shown) reveals that the default parameter set for *RNAfold* indeed belongs to the linear phase of the parameter space. First the precursor for the miRNA mir-37 from *C. elegans* was embedded into a sequence of random DNA. The sequence of random nucleotides is generated from a first order Markov chain model using the transition probabilities estimated for *C. elegans*. The total length of the resulting sequence is 1000. The results of running either program on this artificial sequence can be seen in Figure 5. As long as we take the window size large enough ($L > 94$ will do), *StemSearch* produces almost[3] the same 12 highest scoring stems as output. On the contrary, *RNALfold* produces a highly different output depending upon whether $L = 100$ or $L = 200$. Only when $L = 100$ does *RNALfold* rank the segment

[3]A few, longer, suboptimal segments are found when we raise the window size to 200, say.

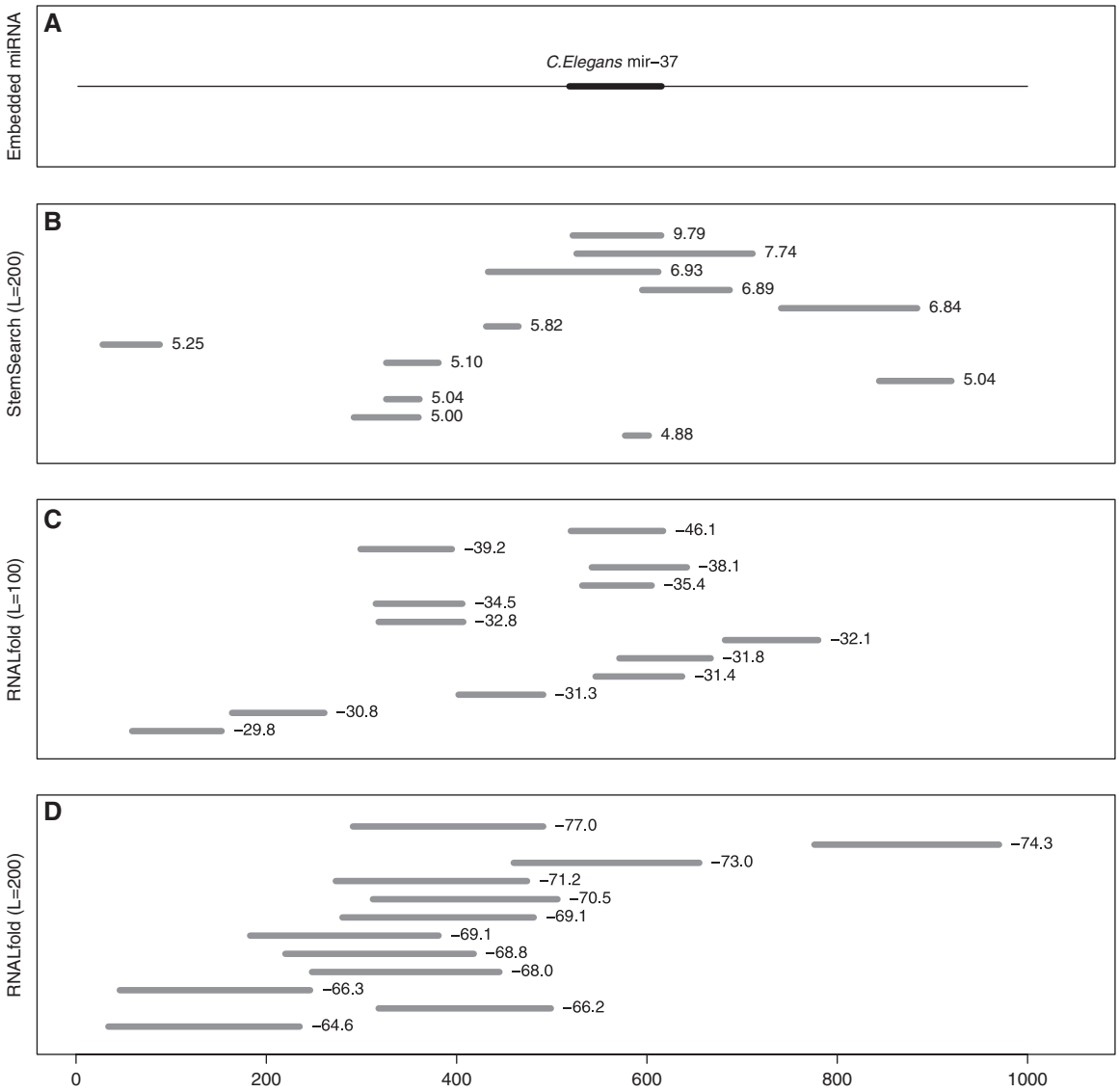

FIG. 5. The *C. elegans* mir-37 miRNA precursor consisting of 98 nucleotides is embedded in a sequence of random DNA of a total length 1000 (**A**). *StemSearch* locates almost the same high-scoring segments *taking any bandwidth* as long as $L > 94$ (here $L = 200$), and the figure shows the position and score of the twelve highest scoring segments (**B**). Using *RNALfold* with window sizes $L = 100$ (**C**) or $L = 200$ (**D**) produces completely different results, and only with $L = 100$ does *RNALfold* identify the mir-7 precursor.

corresponding to the embedded mir-37 precursor as number 1. For $L = 200$, the mir-37 precursor drowns in noise. Note in addition that the size of the high-scoring segments from *RNALfold* are all close to the actual window size, which is what we would expect from the fact that the energy parameters are in the linear phase of the parameter space. Although Figure 5 only shows a single miRNA, the picture is a generic representation of the differences between the output from *StemSearch* and *RNALfold*.

To further illustrate this difference on real data, we ran both programs on two 20kb regions of the *Human* genome that each contain a mir-cluster—one containing mir-17 and one containing mir-106a—and which contain a total of 12 miRNA's. Figure 6 shows the final results. What was first observed was that both regions contain an annotated CpG-island. As a consequence of the strong CG-bias in the CpG-islands both programs produce a large number of high-scoring hits in the CpG-islands and they are thus masked before further analysis. Likewise, both regions were masked using RepeatMasker. What is shown in Figure 6 is the result after the masking. We may note that there are still traces of the CpG-islands at the boundaries of the

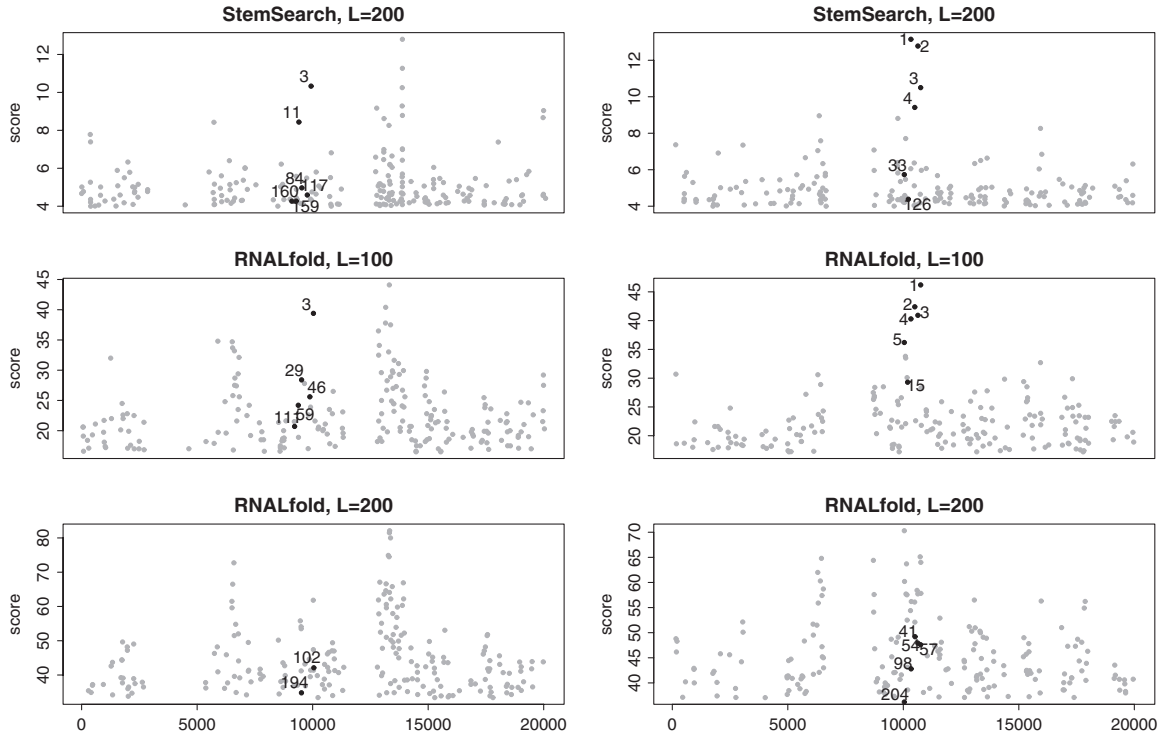

FIG. 6. High scoring structures from *StemSearch* and *RNALfold* on two 20kb regions of the *Human* genome containing the mir-17 miRNA cluster **(left)** and the mir-106a cluster **(right)**. Both clusters contain six annotated miRNA's. Both regions contain a CpG-island annotation, and both programs produce a larger number of high scores on the borders of the masked CpG-island. The hits from either program that matches the annotated miRNA's best are shown together with the ranking of the hit among the hits plotted.

annotations—especially for the mir-17 region. To define which of the hits from either program that correspond to the annotated miRNA we computed for each hit the distance to the annotation as the sum of the distance between the starting position of the annotation and the hit and the ending position of the annotation and the hit. Then we selected the hit with the smallest distance as being the "score for the annotated miRNA." These scores are shown on Figure 6 together with the rank of the hit among those hits plotted. In terms of locating the miRNA's as high-scoring hits, *StemSearch* with $L = 200$ and *RNALfold* with $L = 100$ show comparable performance with *RNALfold* being perhaps slightly better for the mir-106a region. However, the performance for *RNALfold* degrades when we set $L = 200$. It may be argued that because *RNALfold* operates in the linear phase, one should normalize the scores by dividing with the length of the hit. Doing so (not shown) improves the performance of *RNALfold* for $L = 200$ but degrades the performance for $L = 100$ leaving no clear picture whether one should normalize or not.

4. DISCUSSION

We have shown how a statistical theory for the occurrence of local RNA structures can be formulated and how the effect of the scoring parameters enters through the two parameters λ and K. The statistical model is incorporated in the corresponding implementation *StemSearch*, which is a program for local folding of longer sequences. The program is a dedicated datamining tool, and the implementation is sufficiently fast to be able to scan entire genomes for stem-loops. If we search a sequence of length n, *StemSearch* provides a ranked list of occurrences of local stem-loop structures, a *nat-score* s and an E-value. The *nat-score* is an intrinsic quantification of the corresponding stem loop, and the E-value expresses the expected number of random stem loops with a *nat-score* exceeding s in a random sequence of length n. The computation of the *nat-score* is based on two parameters λ and K, which come from the

statistical model, and which depend upon the scoring parameters, the penalty parameters and the null model.

An interesting point is that the length n of the sequence we search in enters in the formula (3) in a multiplicative way and the bandwidth L does not enter at all. Consequently, the suggested method for optimization of the penalty parameters by minimization of the expected number of false positives is unaffected by the length n as well as the bandwidth L, and we can estimate the parameters using a dataset—simulated, shuffled or real sequences—that is much smaller than typical genomes. This reduces the computational costs dramatically as compared to brute-force simulations to compute estimates of the expected number of false positives (for a given threshold), say, and it becomes practical to optimize such a quantitative measure of discriminative performance over the penalty parameters.

We also showed how standard estimators from extreme value statistics can be used to estimate λ and K. The data used for the estimation can be the output from running *StemSearch* on simulated sequences, or if possible running *StemSearch* on a valid ''null model'' dataset of sequences that do not contain stem loops. Using simulated sequences, based on the species specific first order Markov chain, we illustrated how the statistical theory and in particular the *nat-score* normalization can be utilized to optimize the measure of discriminative power over the penalty parameters.

In this paper we have focused on optimizing over the three penalty parameters. From Table 1 the results seem fairly robust to the precise choice of objective function. It also seems that there are notable differences between the species, but we emphasize that we provide no evidence that the differences are statistically significant. We also illustrated that there can be a considerable gain in choosing optimal penalty parameters as compared to non-optimal penalty parameters. The stack score parameters (in total $6\times16=96$ free parameters as we allow for canonical and GU/UG pairs only) were derived as log-likelihood ratios. As an alternative we could minimize the expected number of false positives over the entire $96+3$-dimensional space of stack score and penalty parameters. This does, however, raise a number of difficulties. The dimensionality of the parameter space makes this a non-trivial numerical optimization problem, and either the run-time of *StemSearch* must be lowered or the computations must be parallelized to make this a realistic endeavor. There is also a real chance that such an optimization will overfit the parameters dramatically to the specific training dataset used.

Finally we compared the output from *StemSearch* with the output from *RNALfold*. We showed that due to the fact that *RNALfold* uses parameters from the linear phase, it is rather sensitive to the choice of the bandwidth L. *StemSearch* is less sensitive to the precise choice of L, and with $L=200$ its ability to locate miRNA's, whose sizes are in the range of 68–96 bases, in two 20-kb regions of the *Human* genome containing two miRNA-clusters, was comparable to that of *RNALfold* with $L=100$ and superior to *RNALfold* with $L=200$. As a final remark we observed that neither of the programs was able to clearly discriminate all 12 real miRNA's from the remaining genome sequence. This is most likely a consequence of the general difficulty in discriminating RNA-genes from the bulk genome based on structural scores alone. With the statistical theory behind *StemSearch* we have provided a novel method for optimizing the discriminative performance over the parameter space. Future improvements may be obtained by combinations of (1) extentions of the scoring scheme in *StemSearch* in the direction of the more flexible scheme in *RNALfold*, (2) optimization over larger parts of the (extended) parameter space, and (3) optimization against more realistic null-models, for instance by using empirical sequence data as null models.

ACKNOWLEDGMENTS

Thanks are due to Anders Krogh and Peter Arctander for introducing me to the subject of miRNA, and to Morten Lindow and Paul Gardner for discussions. This work was supported by the Danish Natural Science Research Council (grant 272-06-0442). Part of this work was carried out while the author was Assistant Professor at the Bioinformatics Centre, University of Copenhagen.

DISCLOSURE STATEMENT

No competing financial interests exist.

REFERENCES

Altschul, S.F., Bundschuh, R., Olsen, R., et al. 2001. The estimation of statistical parameters for local alignment score distributions. *Nucleic Acids Res.* 29, 351–361.

Buratti, E., and Baralle, F.E. 2004. Influence of RNA secondary structure on the pre-mRNA splicing process. *Mol. Cell Biol*. 24, 10505–10514.

de Haan, L., and Ferreira, A. 2006. *Extreme Value Theory*. Springer, New York.

Embrechts, P., Klüppelberg, C., and Mikosch, T. 1997. *Modelling Extremal Events. Volume 33 of Applications of Mathematics*. Springer-Verlag, Berlin.

Griffiths-Jones, S., Moxon, S., Marshall, M., et al. 2005. Rfam: annotating non-coding RNAs in complete genomes. *Nucleic Acids Res*, 33, D121–D124.

Hansen, N.R. 2007. Asymptotics for local maximal stack scores with general loop penalty function. *Adv. Appl. Probabil.* 39, 776–798.

Hofacker, I.L., Fontana, W., Stadler, P.F., et al. 1994. Fast folding and comparison of RNA secondary structures. *Monatshefte f. Chemie* 125, 167–188.

Hofacker, I.L., Priwitzer, B., and Stadler, P.F. 2004. Prediction of locally stable RNA secondary structures for genome-wide surveys. *Bioinformatics* 20, 186–190.

Lindow, M., and Gorodkin, J. 2007. Principles and limitations of computational microRNA gene and target finding. *DNA Cell Biol*. 26, 339–351.

Svoboda, P., and Cara, A.D. 2006. Hairpin RNA: a secondary structure of primary importance. *Cell Mol Life Sci*. 63, 901–908.

Xiong, M., and Waterman, M.S. 1997. A phase transition for the minimum free energy of secondary structures of a random RNA. *Adv. Appl. Math.* 18, 111–132.

Yoon, S., and Micheli, G.D. 2006. Computational identification of microRNAs and their targets. *Birth Defects Res. C Embryo Today* 78, 118–128.

Zhang, B., Wang, Q., and Pan, X. 2007. MicroRNAs and their regulatory roles in animals and plants. *J. Cell. Physiol*. 210, 279–289.

Address reprint requests to:
Dr. Niels Richard Hansen
Department of Mathematical Sciences
University of Copenhagen
Universitetsparken 5
2100 Copenhagen O, Denmark

E-mail: richard@math.ku.dk