

Dinucleotide Controlled Null Models for Comparative RNA Gene Prediction

Tanja Gesell^{1,2,3,4} & Stefan Washietl^{5,6*}

¹Center for Integrative Bioinformatics Vienna, Max F. Perutz Laboratories, Dr. Bohr-Gasse 9, A-1030 Vienna, Austria

²University of Vienna, ³Medical University of Vienna, ⁴University of Veterinary Medicine Vienna

⁵Institute for Theoretical Chemistry, University of Vienna, Währingerstraße 17, 1090 Vienna, Austria

⁶EMBL-European Bioinformatics Institute, Wellcome Trust Genome Campus, Hinxton, Cambridge CB10 1SD, UK

Email: Tanja Gesell - tanja.gesell@univie.ac.at; Stefan Washietl* - washietl@ebi.ac.uk;

*Corresponding author

Abstract

Background: Comparative prediction of RNA structures can be used to identify functional noncoding RNAs in genomic screens. It was shown recently by Babak *et al.* [BMC Bioinformatics. 8:33] that RNA gene prediction programs can be biased by the genomic dinucleotide content, in particular those programs using a thermodynamic folding model including stacking energies. As a consequence, there is need for dinucleotide-preserving control strategies to assess the significance of such predictions. While there have been randomization algorithms for single sequences for many years, the problem has remained challenging for multiple alignments and there is currently no algorithm available.

Results: We present a program called SISSId that simulates multiple alignments of a given average dinucleotide content. Meeting additional requirements of an accurate null model, the randomized alignments are on average of the same sequence diversity and preserve local conservation and gap patterns. We make use of a phylogenetic substitution model that includes overlapping dependencies and site-specific rates. Using fast heuristics and a distance based approach, a tree is estimated under this model which is used to guide the simulations. The new algorithm is tested on vertebrate genomic alignments and the effect on RNA structure predictions is studied. In addition, we directly combined the new null model with the RNAalifold consensus folding algorithm giving a new variant of a thermodynamic structure based RNA gene finding program called SISSIz that is not biased by the dinucleotide content.

Conclusions: We presented an efficient algorithm to randomize multiple alignments preserving dinucleotide content. It can be used to get more accurate estimates of false positive rates of existing programs, to produce negative controls for the training of machine learning based programs, or as a null model of a standalone RNA gene finding program. Other applications in comparative genomics that require randomization of multiple alignments can be considered.

Availability : SISSId/SISSIz are available as open source C code that can be compiled for every major platform and downloaded here: www.tbi.univie.ac.at/papers/SUPPLEMENTS/SISSIz.

1 Background

Comparative genome analysis is currently the most widely used strategy to detect and annotate noncoding RNAs (ncRNAs) [1, 2]. In the past few years a series of different algorithms have been developed that predict functional ncRNAs on the basis of conserved secondary structure [3–10]. Some of these methods have been used to predict novel ncRNAs on a genome wide scale [7, 11–14]. In combination with experimental verification (microarray, RT-PCR, Northern blot) these methods could successfully uncover many examples of novel ncRNAs [15–20]. However, in particular in large vertebrate genomes the signal-to-noise ratio of true predictions and false positives is thought to be relatively low [20]. In a recent paper, Babak and colleagues demonstrated that comparative ncRNA gene finders are strongly biased by the genomic dinucleotide content leading to an excess of false predictions [21]. Especially methods that are based on a thermodynamic folding model are sensitive to this effect: In the so-called nearest neighbour model, energies are not assigned to single base-pairs but rather to neighbouring base-pairs that stack on each other. As a consequence, the folding stability of genomic sequences does not only depend on the mononucleotide content but also the dinucleotide content.

To assess the significance of predicted structures, e.g. to estimate the false discovery rate in a genomic screen for ncRNAs, one should therefore compare the genomic predictions to the results obtained on randomized data with the same dinucleotide content. In the case of single sequences, there are well known and widely used algorithms to generate dinucleotide controlled random sequences either by shuffling or first order Markov chain simulation [22, 23]. However, there is currently no algorithm to randomize multiple sequence alignments preserving the dinucleotide content. Babak and colleagues [21] added the conserva-

tion of dinucleotides as an additional constraint to the commonly used (mononucleotide) shuffling algorithm `shuffle-aln.pl` [5] and applied it to pairwise alignments. Their approach corresponds to a heuristic used in reference 24, that is very inefficient as only a small subspace of the whole permutation space is covered¹. But even a more efficient shuffling algorithm will soon run into difficulties on multiple alignments. Unless two neighbouring columns are 100% conserved, there are several different dinucleotide pairs in these columns. It is therefore impossible to exactly preserve the dinucleotide content as in the single sequence case.

In this paper, we address the problem in a different way. In analogy to a first order Markov model for single sequences, we simulate alignments of a given dinucleotide content. We present a substitution model that captures the neighbour dependencies and all other relevant alignment features. We describe a time efficient way to estimate a tree under this model that we use as a guide to simulate alignments of the desired properties. This new control strategy is tested on genomic alignments and the effect on thermodynamic RNA structure predictions is studied.

2 Algorithm

2.1 Requirements for an accurate null model

An optimal null model preserves all the features of the original data with the exception of the signal under question that needs to be removed efficiently. In our case, the data are multiple alignments of homologous sequences and the signal of interest is an evolved RNA secondary structure. Correlations arising from base-pairing patterns need to be removed. Currently, alignments are usually randomized by shuffling the alignment columns (see ref. 5 for a discussion of this method). Although the shuffling approach has its limitations and considering dinucleotides seems difficult, it is an appealing approach because it is relatively simple, fast, and extremely conservative. Changing the order of the columns does not change the mutational patterns within the columns and thus the underlying phylogenetic tree is exactly preserved.

In this paper we attempt to simulate new alignments from scratch. Even the most sophisticated model cannot capture all evolutionary processes and therefore a simulation approach will inevitably change the original data more than shuffling does. So much care has to be taken to preserve all the relevant characteristics of the data. To qualitatively assess the most important parameters that need to be considered in our model, we performed a series of simulation experiments. Using a simple tree with four taxa we simulated alignments under the HKY evolutionary model [25]. We systematically varied model and tree parameters to study how

¹The heuristic exchanges only positions that have the same neighbours left and right. For the short sequence ACAGCCAA for example not a single permutation can be found that way. However, there are 11 such permutations according to the Altschul & Erikson algorithm [22].

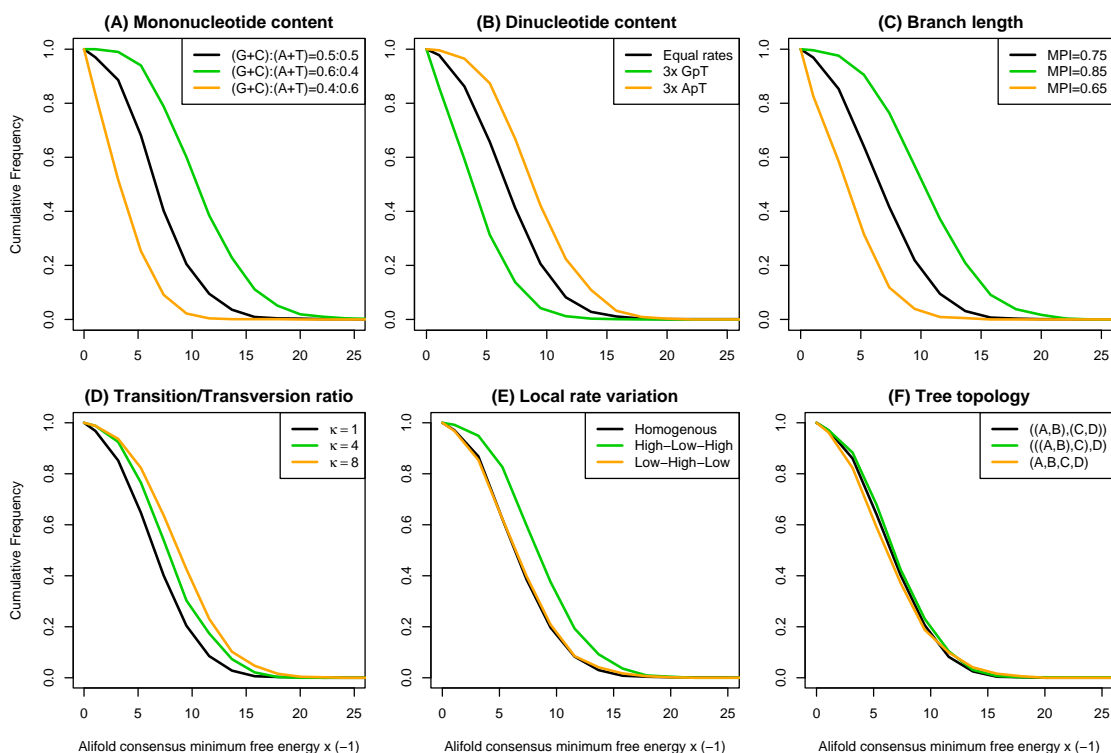


Figure 1: Parameters effecting thermodynamic consensus RNA structure predictions. As a basic parameter set we used equal base frequencies of 0.25, a transition/transversion rate ratio $\kappa = 1$, and the following tree $((A:0.09,B:0.09):0.09,(C:0.09,D:0.09):0.09)$. One parameter was varied at a time while others were kept constant. If necessary branch lengths were adjusted to keep a MPI of 0.75 ± 0.01 . 1000 alignments of length 80 were simulated under each condition. Cumulative histograms for the RNAalifold consensus folding energies are shown. Please note that we plot negative minimum free energies, i.e. higher values correspond to more stable folds. **(A)** Base frequencies were varied to get high and low G+C content. **(B)** Two specific dinucleotide frequencies were elevated 3-fold while the mononucleotide content was kept constant. **(C)** Branch lengths were equally scaled to produce alignments with lower or higher MPI identity than for the basic tree. **(D)** The transition/transversion rate ratio was varied. $\kappa = 1$ means equal rates, while $\kappa > 1$ gives more transition than transversions. **(E)** The alignment of size 80 was divided into a central block of 40 and two flanking regions of 20. We set 100% conservation in the central block and low conservation in the flanks (rate “high-low-high”) and the other way round (“low-high-low”). The total average MPI was always 0.75. **(F)** We tested all possible topologies of this 4 taxa tree and adjusted the branch lengths to give a MPI of 0.75. For one given topology, all the branch lengths were of the same length.

they affect thermodynamic RNA consensus structure predictions in the alignment. We used `RNAalifold` [26] to predict consensus secondary structures which is the the basis of the `AlifoldZ` [5] and `RNAz` [6] gene finders.

Not surprisingly, base composition is one of the parameters affecting the predicted folding energies strongest (Fig. 1A). High G+C content leads to more stable RNA predictions, while high A+T content gives less stable predictions. As mentioned in the introduction and in fact the main motivation of this paper, also dinucleotide content affects folding energies. We used our simulation algorithm that is described below to simulate alignments of the same mononucleotide content but varying dinucleotide content. Fig. 1B shows for example that a three times enriched ApT content lead to more stable predictions. The excess of some other dinucleotides like for example GpT can cause the opposite effect leading to less stable predictions.

Another major parameter that needs to be controlled is the sequence diversity of the alignment. Variation of the branch lengths of the tree gives alignments with different sequence diversity which we usually measure as the mean pairwise sequence identity (MPI)². High diversity (i.e. low MPI) makes it difficult to predict a consensus structure if there is no selection pressure for it. On the other hand, almost perfectly conserved sequences fold readily in some random structure even if there is no natural RNA structure present. Therefore we observe a strong dependency on the MPI (Fig. 1C).

One well known characteristic of natural mutation processes are the different rates for transitions and transversions [27]. Interestingly, this also affects the consensus structure predictions. A model with equal transition/transversion rates (parameter $\kappa = 1$ in the HKY model) gives less stable predictions than a model with more realistic rates (e.g $\kappa = 4$, Fig. 1D). This parameter affects the type of column patterns observed in the simulated alignments which in turn affects how well they can form consensus base pairs.

Natural mutation processes are not homogeneous across all sites, in particular in functional genomic regions. It was observed previously that mutation patterns within an alignment can affect structure predictions [5]. For example, an alignment containing a 100% conserved block with low mutation rate that is flanked by highly divergent regions of high mutation rate can have different folding energies compared to an alignment with homogeneous rates but the same overall MPI (Fig. 1E). The same is true for patterns of insertions and deletions which was also already discussed in reference 5 and which we do not show here explicitly again.

We also tested the effect of different tree topologies, but did not find a significant influence of this parameter at least in our four taxa example.

²We found it convenient to use both identity and distance measures at different places in this paper although this might complicate the notation at some points.

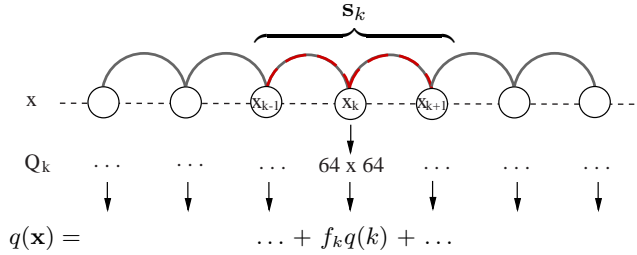


Figure 2: Site dependencies for overlapping dinucleotides (red-gray): The substitution process of a given nucleotide x_k at site k by another one depends on the states x_{k-1}, x_k, x_{k+1} , the subsequence s_k . Q_k has the dimension 64×64 , where only one mutation is allowed at the current site k . The substitution rate for the whole sequence $q(\mathbf{x})$ is the sum of each rate $q(k) = Q_k(s_k, s_k)$ multiplied with a site-specific scaling factor f_k , with $k = 1, \dots, l$.

Taken together, an accurate randomization procedure needs to generate alignments that preserve (i) mono- and dinucleotide content, (ii) mean pairwise sequence identity, (iii) transition/transversion rate ratio (iv) site-specific mutation rates, and (v) gap patterns.

In the next section we describe a model that is capable of simulating alignments under these constraints.

2.2 Model

Sequence evolution is usually described by a time-continuous Markov process [27, 28]. The most commonly used models assume that all sites of a sequence evolve independently from each other rendering it impossible to model dinucleotide dependencies between neighbouring pairs. Various evolutionary models have been proposed in the past years to overcome this limitation [29–36]. We make use of the recently introduced framework called SISSI (SImulating Site-Specific Interactions). SISSI allows to define site dependencies of arbitrary complexity in the form of a “neighbourhood system” that also may include overlapping dependencies [37]. Given the requirements of our specific problem, we extended and simplified several aspects of SISSI as necessary.

Following the general framework of SISSI, we introduce a site-specific rate matrix Q_k for every site $k = 1, \dots, l$ in a sequence $\mathbf{x} = (\mathbf{x}_1, \dots, \mathbf{x}_l)$. This matrix defines the substitution process at site k , where the substitution of a given nucleotide $x_k \in \mathcal{A} = \{A, C, G, U\}$ by another one depends on the states x_{k-1}, x_k, x_{k+1} (Fig. 2).

Thus, the instantaneous rate matrix Q_k has the dimension $|\mathcal{A}|^3 \times |\mathcal{A}|^3 = 64 \times 64$. The stationary distribution of Q_k determines the equilibrium dinucleotide content of our system (see the next section for how the required trinucleotide frequencies of Q_k are calculated from the dinucleotide frequencies).

To be able to control the transition/transversion rate ratio and the site-specific mutation rates, we have to add two additional parameters. Let $\mathbf{s}_k = (x_{k-1}, x_k, x_{k+1})$ represent the current triplet of sequence \mathbf{x} and $\mathbf{y} = (y_1, y_2, y_3)$ another triplet in \mathcal{A}^3 . First, we introduce a general parameter $r_{(s_k, y)} \geq 0$ to incorporate the additional mechanistic rates. Second, we introduce a site-specific scaling factor f_k with $k = 1, \dots, l$, such that:

$$\frac{1}{l} \cdot \sum_1^l f_k = 1. \quad (1)$$

We impose the usual restriction, that only one substitution per unit time is admissible [38,39]. Moreover, Q_k only allows for substitutions at site k . The diagonal elements of our instantaneous rate matrix Q_k are defined by the mathematical requirement that the sum of each row is zero.

The entries of Q_k are thus given by

$$Q_k(\mathbf{s}_k, \mathbf{y}) = f_k \cdot \begin{cases} r_{(s_k, y)} \cdot \pi_k(\mathbf{y}) & \text{if } H(\mathbf{s}_k, \mathbf{y}) = 1 \text{ and } x_k \neq y_2 \\ - \sum_{\substack{\mathbf{z} \in \mathcal{A}^3 \\ \mathbf{z} \neq \mathbf{s}_k}} Q_k(\mathbf{s}_k, \mathbf{z}) & \text{if } H(\mathbf{s}_k, \mathbf{y}) = 0 \\ 0 & \text{otherwise} \end{cases} \quad (2)$$

where $\pi_k(\mathbf{y})$ is the stationary frequency of \mathbf{y} and the Hamming distance $H(\mathbf{s}_k, \mathbf{y})$ counts the number of differences between the sites of the triplets \mathbf{s}_k and \mathbf{y} .

In principle, we can choose any rate for the parameter $r_{(s_k, y)}$. However, based on the requirement that we want to define the counted dinucleotide content as the stationary distribution, we restrict here the parameter $r_{(s_k, y)}$ to be reversible. Any parameter of the commonly used independent nucleotide substitution models, like HKY [25] or the general time-reversible model GTR [40] can be chosen for $r_{(s_k, y)}$. For our application, we use a transition-transversion rate ratio and set $r_{(s_k, y)} = \kappa$ for transitions and $r_{(s_k, y)} = 1$ for transversions.

The restriction that a substitution is only possible at site k leads to sparse rate matrices. Q_k has only $|\mathcal{A}|^4$ non-zero entries. Hence, we can write Q_k in the form of 16 submatrices, which describe the admissible substitutions for site k depending on the left y_1 and right y_3 neighbours,

$$\begin{array}{c} \text{A} | y_1, y_3 \\ \text{C} | y_1, y_3 \\ \text{G} | y_1, y_3 \\ \text{U} | y_1, y_3 \end{array} \begin{pmatrix} \text{A} | y_1, y_3 & \text{C} | y_1, y_3 & \text{G} | y_1, y_3 & \text{U} | y_1, y_3 \\ \left(\begin{array}{cccc} * & \pi_{\text{C} | y_1, y_3} & \kappa \pi_{\text{G} | y_1, y_3} & \pi_{\text{U} | y_1, y_3} \\ \pi_{\text{A} | y_1, y_3} & * & \pi_{\text{G} | y_1, y_3} & \kappa \pi_{\text{U} | y_1, y_3} \\ \kappa \pi_{\text{A} | y_1, y_3} & \pi_{\text{C} | y_1, y_3} & * & \pi_{\text{U} | y_1, y_3} \\ \pi_{\text{A} | y_1, y_3} & \kappa \pi_{\text{C} | y_1, y_3} & \pi_{\text{G} | y_1, y_3} & * \end{array} \right) \end{pmatrix} \quad (3)$$

Finally, we scale Q_k such that the number of substitutions d_k equals 1:

$$d_k = - \sum_{\mathbf{z} \in \mathcal{A}^3} \pi_k(\mathbf{z}) \cdot Q_k(\mathbf{z}, \mathbf{z}) = 1. \quad (4)$$

and thus the total instantaneous substitution rate for a sequence \mathbf{x} can be written as the sum over each rate $Q_k(\mathbf{s}_k, \mathbf{s}_k)$ multiplied with the site-specific scaling factor f_k , with $k = 1, \dots, l$ (Fig. 2),

$$q(\mathbf{x}) = - \sum_{k=1}^l f_k \cdot Q_k(\mathbf{s}_k, \mathbf{s}_k). \quad (5)$$

Without dependencies on the neighbours, Q_k is of dimension 4×4 and the model reduces essentially to a HKY model with a specific rate for each site. We use this mononucleotide variant later in this paper for testing and comparison to the dinucleotide model.

2.3 Simulation

For the simulation process, we essentially used the same algorithm described previously [37] with some modifications. During the simulation process, we pick a site k with the relative mutability

$$P(k) = \frac{|f_k \cdot Q_k(\mathbf{s}_k, \mathbf{s}_k)|}{q(\mathbf{x})}, \quad (6)$$

and for the chosen site k , the nucleotide x_k will be replaced by a new nucleotide $y_2 \in \mathcal{A}$ from the corresponding triplet \mathbf{y} with probability:

$$P(x_k \rightarrow y_0) = \frac{f_k \cdot Q_k(\mathbf{s}_k, \mathbf{y})}{|f_k \cdot Q_k(\mathbf{s}_k, \mathbf{s}_k)|} = \frac{Q_k(\mathbf{s}_k, \mathbf{y})}{|Q_k(\mathbf{s}_k, \mathbf{s}_k)|} \quad (7)$$

In the most general SISSI framework Q_k needs to be updated for all k sites every time one nucleotide in \mathbf{x} is substituted. However, in our special case we can use the same instantaneous rate matrix Q_k for each site with special conditions for $r_{(\mathbf{s}_k, \mathbf{y})}$. As a consequence, we can fix $q(\mathbf{x})$ and do not need to sum over each rate of the site, which improves the running time of the algorithm.

2.4 Parameter estimation

The idea of our randomization procedure is to estimate a tree under the model described in the previous section and simulate sequences along this tree. Ideally, all parameters are estimated simultaneously within a maximum likelihood framework. One problem is the high number of parameters since we want to estimate

a specific rate for each site. A more fundamental issue is, however, that our model includes overlapping dependencies which breaks the independence assumption necessary for basic maximum likelihood estimation. Other possible techniques like Markov chain Monte Carlo in a Bayesian framework are not a viable alternative either. Speed is a critical issue as the algorithm is meant to be applied to data on a genome wide scale.

Facing these difficulties, we have developed heuristic approximations to estimate the parameters and use a distance based approach to estimate the tree. The method is fast and yet surprisingly accurate for our application.

2.4.1 Equilibrium frequencies

The stationary frequencies of our model are set in a way that in equilibrium we obtain a dinucleotide frequency that is the same as the dinucleotide content of the alignment to be randomized. To this end, we first count the dinucleotide frequencies as an average of all sequences in the original alignment (see Methods on how gaps are treated). Then, we calculate the corresponding trinucleotide frequencies needed for Q_k as a function of the single and dinucleotide frequencies as follows [30, 32],

$$\pi(\alpha\beta\gamma) = \frac{\pi(\alpha\beta)\pi(\beta\gamma)}{\pi(\beta)} \quad (8)$$

where $\pi(\alpha\beta\gamma)$ are the trinucleotide frequencies, $\pi(\alpha\beta)$ and $\pi(\beta\gamma)$ the counted dinucleotide frequencies and $\pi(\beta) = \sum_{\alpha} \pi(\alpha\beta) = \sum_{\gamma} \pi(\beta\gamma)$ with $\alpha, \beta, \gamma \in \{A, C, G, U\}$.

Fig. 3A shows an example of the dinucleotide frequency distribution of 1000 simulated alignments. We counted the dinucleotide frequencies of an alignment of 7 5.8 rRNA sequences and set the trinucleotide parameters of our model accordingly. On average, we get the same dinucleotide frequencies in the simulated alignments as in the original one.

2.4.2 Distances and tree construction

To build a distance based tree, we first have to estimate the number of substitutions that have taken place between two sequences. In other words, we have to estimate the genetic or evolutionary distance d from the Hamming distances p under our model. Both distances are different because back mutations have taken place that are not directly visible. To estimate the relationship between d and p , we simulate sequence pairs separated by different branch lengths d and calculate the corresponding Hamming distances p (Fig. 3B). We fit an exponential function to this curve:

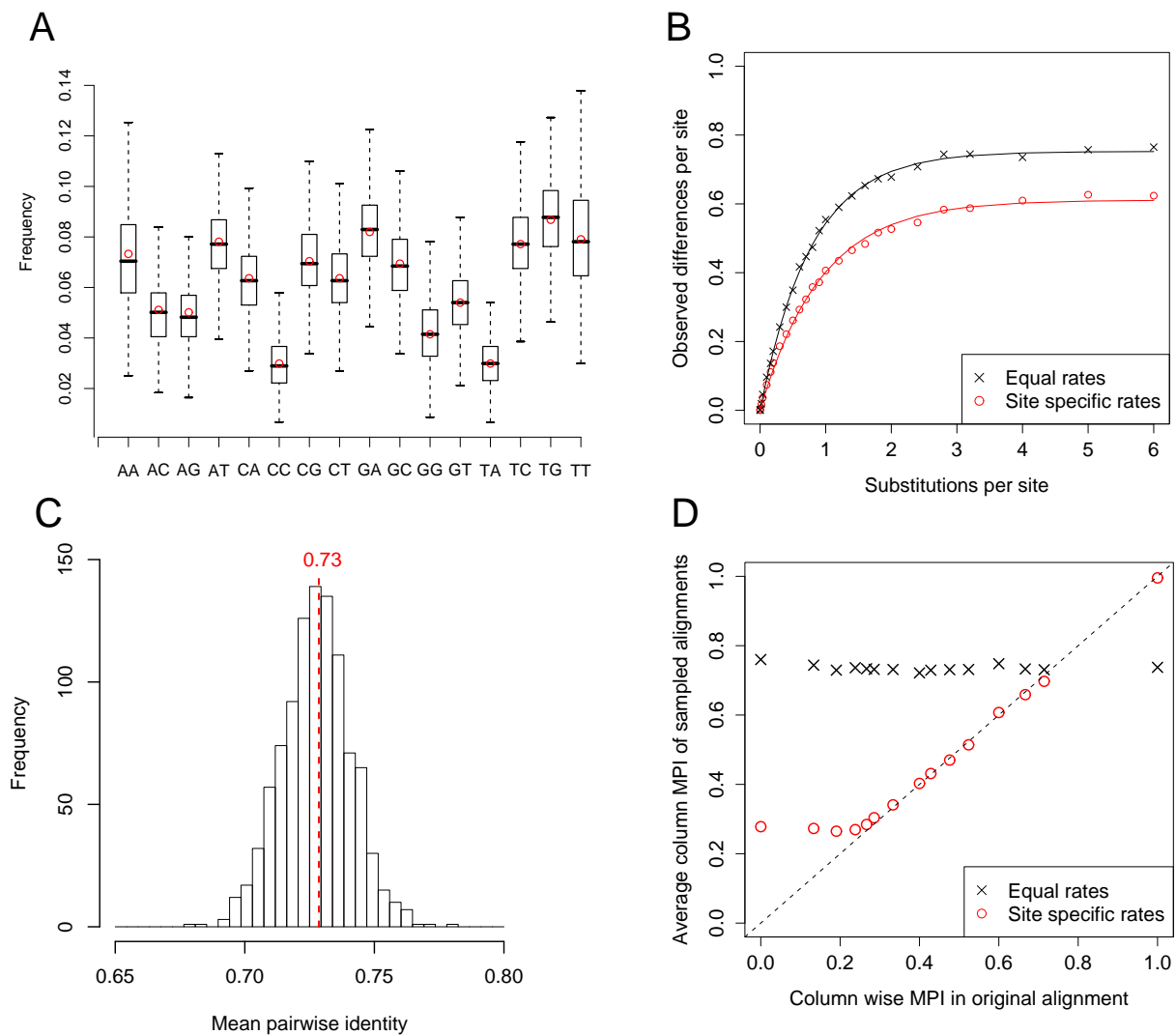


Figure 3: Key concepts of the algorithm shown on an example alignment of 5.8S rRNA. **(A)** Distribution of dinucleotide frequencies of 1000 simulated alignments are shown. Red circles indicate the frequencies observed in the original alignment. **(B)** Relationship between the number of substitutions and observed differences empirically determined by sampling of 25 points. Each point shows the average of 10 simulations. Note that the short distances are sampled more densely. These setting are the default values and used throughout the paper. **(C)** Distribution of mean pairwise identities for 1000 random samples. The MPI of the original alignment is shown in red. **(D)** Comparison of site-wise MPIs in the original alignment and the average of the corresponding sites of 1000 random alignments.

$$p = \hat{a} \cdot (1 - e^{\hat{b} \cdot d}) \quad (9)$$

Using this function, all pairwise distances are calculated for the sequences in the original alignment. From this distance matrix a tree is constructed using the BIONJ algorithm [41]. BIONJ is a variant of the well known neighbour joining algorithm and currently one of the most accurate algorithms for distance based tree building.

Given that the distances and the tree is accurately estimated, we observe on average the same mean pairwise identity in the simulated alignment as in the original one. Fig. 3C shows the distribution of MPIs of 1000 simulations of our example rRNA alignment. The average MPI of the simulations is exactly the same as the MPI 0.73 of the original alignment.

2.4.3 Site-specific rates

Setting different mutation rates at different sites gives us the possibility to preserve natural mutation patterns of the original alignment. The problem of finding accurate site-specific rates is illustrated in Fig. 3D. For each site in the alignment, the MPI of this site is plotted against the average MPI observed in the simulated alignments on the same site. If we consider equal rates for all sites, each site will have the same average MPI which is of course equal to the overall MPI of 0.73 of the whole alignment. Ideally, the average MPI for each simulated site is the same as the original MPI at this site. In this case, the points in the plot are on a diagonal indicating that we have found accurate estimates for the rates.

Simple estimates for site-specific rates in combination with distance based trees have been described previously [42]. The method includes fits to a gamma distribution which requires data of at least 1000 nucleotides and 30 sequences to get reasonable results. Here we use a different approach that also gives good results for smaller alignments.

The substitution rate at a site is of course related to the observed sequence diversity at this site. If a site is highly conserved the rate is low, whereas high sequence diversity indicates a high mutation rate. So in a first step, we calculate the average number of pairwise differences $\langle p_k \rangle$ for each site k in the alignment with n sequences:

$$\langle p_k \rangle = \frac{2}{n(n-1)} \sum_i^n \sum_{j>i}^n \delta_{ij}^k; \quad \text{with } \delta_{ij}^k = \begin{cases} 1 & \text{if nucleotides in sequences } i, j \text{ differ at site } k \\ 0 & \text{otherwise} \end{cases} \quad (10)$$

$\langle p_k \rangle$ are observed differences ignoring multiple substitutions. If we naively choose our rates proportional to $\langle p_k \rangle$ we would underestimate high rates while overestimating low rates. We therefore use the relationship in equation 9 to correct for this effect and calculate estimates \hat{f}_k for the the rates at site k as follows:

$$\hat{f}_k = \frac{1}{\hat{b}} \cdot \ln \left(1 - \frac{\langle p_k \rangle}{\hat{a}} \right) \quad (11)$$

It must be pointed out that the site-specific rates change the relationship between genetic distance and observed differences (Fig. 3B). For correcting the site-specific rates we use the estimates for \hat{a} and \hat{b} from our model *without* site-specific rates. So this is only an approximation and one could think about iteratively refining the estimates. However, we found that this approach already yields accurate rates within one step as can be seen in Fig. 3D. Using the model with site-specific rates, the simulated alignments have on average almost exactly the same site-wise MPI as the original one.

The reader will notice that the first three points deviate from the diagonal. This illustrates a limitation of our method. With our simulation procedure we can on average only reach the level of saturation even if we use very high rates. It is possible, however, that the original data contains sites below the level of saturation. For example in a four way alignment a column can be ACGT, i.e. MPI=0. However, we cannot simulate on average columns with MPI=0, since the MPI is bounded below by zero and our simulations will always contain columns with MPI > 0. In practice this does not seem to cause any obvious problems in particular when we have many sequences where it is unlikely to see columns below saturation.

2.4.4 Gaps

Gaps have been ignored completely so far. There are evolutionary models including gaps [43–46] and it would be possible to include this in our simulation algorithm. However, this does not appear practical in our case. Alignment programs produce gap patterns that do not necessarily reflect phylogenetically reasonable insertion/deletion events [47] and thus cannot always be captured by an idealized model that is motivated by evolutionary processes and ignores algorithmic idiosyncrasies of alignment programs. Moreover, it is difficult to estimate reasonable gap model parameters on relatively short alignments which hold only little information.

So we follow here a very pragmatic strategy that has also been used previously [5]: We keep exactly the same gap pattern in our randomized alignments as in the original alignment. To this end, we simply treat gaps as missing data and simulate nucleotide characters for the gapped positions. This is done in a way that

the overall characteristics are not changed when they are replaced with gaps again at the end (see Methods for details).

2.4.5 Transition/transversion rate ratio

The transition/transversion rate ratio κ is a parameter in our model that cannot be simply counted as in the case of the dinucleotide frequencies, or empirically determined like the branch lengths. Given that the influence of this parameter is not that critical as for example the branch length or base composition (see Fig. 1), one possibility might be to use a fixed transition/transversion ratio if a reasonable average value is known for the genome at hand. Alternatively, we found that a good estimate can be obtained by using maximum likelihood on an independent mononucleotide model. We used here the HKY model with Γ -distributed rates which is closest to our dinucleotide model.

2.5 Putting it together

Fig. 4 gives a short outline over the whole randomization procedure. We start by parametrization our model: we count the dinucleotides and calculate the corresponding stationary trinucleotide frequencies. A transition/transversion rate ratio for the alignment is estimated using maximum likelihood under a HKY+ Γ model. Having set these parameters, we empirically estimate the relationship between substitutions and observed differences with equal rates for each site. This first estimate is used to calculate the site-specific rates, which are then used for the second estimation. In the next step, the pairwise distances between all sequences are calculated. For the calculation of the site-specific rates and the pairwise distances gap characters are treated in a special way as missing data (see Methods). From the distance matrix a tree is built using the BIONJ algorithm. An ancestral sequence is sampled from a first order Markov model parametrized according to the dinucleotide frequency in the original alignment. This is used as a starting sequence for the simulation that is guided by the tree. Finally, the gap pattern of the original alignment is introduced into the simulated one. Fig. 5 shows our rRNA example and two randomized versions obtained by this procedure.

3 Implementation

We implemented our method in ANSI C in a program called `SISSId`. The source code is freely available under the the GNU Public Licence. The version that was used for this paper is available here: www.tbi.univie.ac.at/papers/SUPPLEMENTS/SISSIz. Future versions of `SISSId` will be hosted here: www.cibiv.

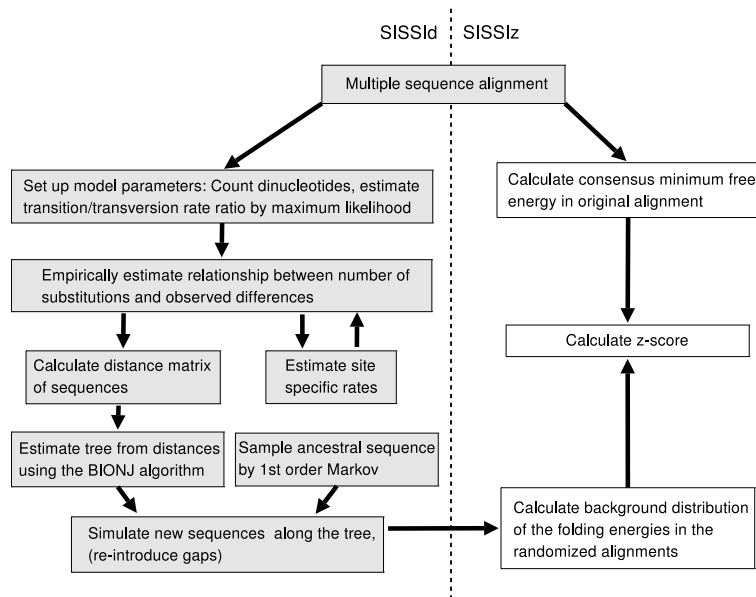


Figure 4: Overview of the algorithm. Left: the steps of the randomization procedure are shown. This algorithm is implemented in SISSId. Right: In combination with RNAalifold consensus folding the randomization procedure can be used to calculate z -scores and to predict significant RNA structures (SISSIz). See text for details.

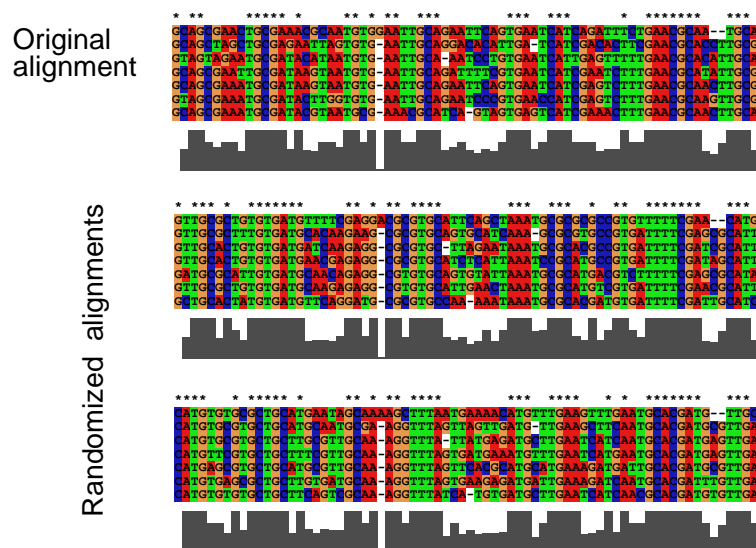


Figure 5: Example of randomized alignments. Part of the example alignment used in Fig. 3 are shown. The grey bars indicate the level of local conservation. Exactly conserved sites are marked by asterisks.

at/software/sissi.

In addition we have implemented a program `SISSIZ` that predicts RNA consensus secondary structures using the `RNAalifold` algorithm allowing to calculate z -scores as described in more detail below. `SISSIZ` can be used as a stand-alone RNA gene prediction program similar to the Perl program `AlifoldZ` [5]. It is available with the supplement of this paper and will be included in an upcoming release of the `RNAz` software package that is available here: www.tbi.univie.ac.at/~wash/RNAz.

Some words on running time: One might suspect that the randomization algorithm including two times the sampling procedure to estimate the parameters of equation 9 and the maximum likelihood estimation of the transition transversion rate ratio is relatively slow. Indeed, it is much slower than for example randomization by shuffling, but still very fast. To build the model for our example of 7 rRNAs of 158 length takes 0.2 seconds on a modern Intel Core 2 Quad CPU at 2.4 GHz. To simulate 1000 alignments using this model takes another 0.6 seconds.

4 Testing

4.1 Randomizing vertebrate genomic alignments

We tested our randomization method on vertebrate genomic alignments. In a setting similar to recent genomic screens in vertebrates [11,20], we extracted `Multiz` [48] alignment blocks from human chromosome 1. We randomly selected 1000 alignment blocks between 70 and 120 in length and between 4 and 10 sequences without considering annotation information of any sort. These alignments are meant to represent an unbiased “genomic background” that may also contain functional elements like coding exons or structured RNAs depending on their frequency in the genome.

The alignments were randomized using our new simulation procedure with both the dinucleotide and the mononucleotide model. In addition, we shuffled the alignments using `shuffle-aln.pl`. The global distribution of dinucleotides for the original and randomized data is shown in Fig. 6. As expected, the shuffling approach and the mononucleotide simulation give the same results. The dinucleotide distribution obtained by these methods, however, differs from the distribution in the native alignments. One can see for example the well known under-representation of CpGs in the native genomic data. Using our dinucleotide based model, we obtain simulated alignments which are statistically indistinguishable from the native data in terms of their average dinucleotide content.

Also the observed sequence diversity of the simulated alignments closely follows the original data as shown in Fig. 7. 98.7% of the simulated alignments are within a range of ± 0.05 mean pairwise identity compared

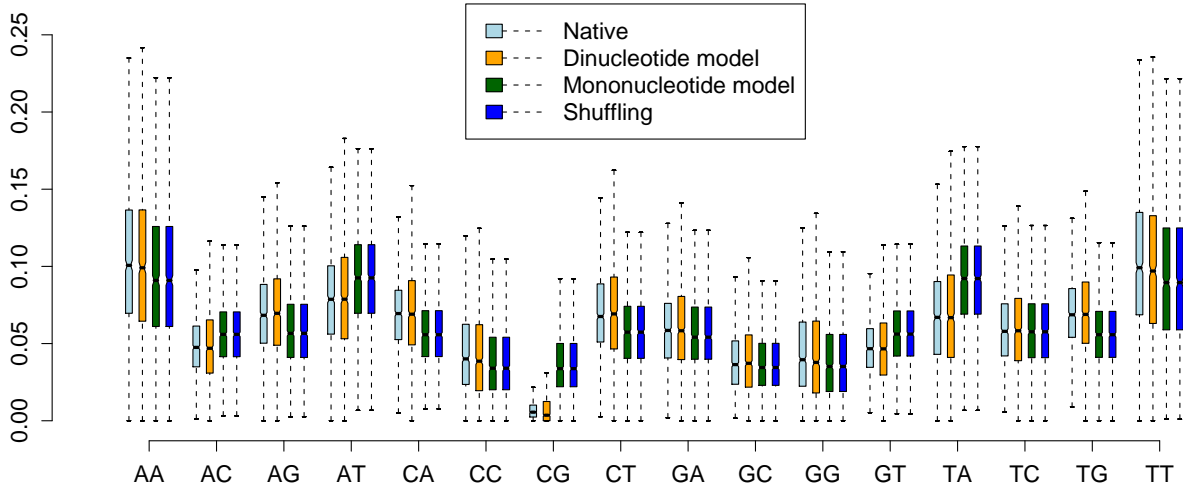


Figure 6: Dinucleotide frequencies of genomic alignments. 1000 vertebrate genome alignments were randomized using three different methods. The dinucleotide frequency of the native and randomized data is shown as box-plots.

to the original alignments. It must be noted, that the distribution in Fig. 7 has a mean of $+0.007$ which indicates a subtle bias of the simulations towards higher MPIs. We suspect that this is an indirect result of the way we estimate site-specific rates and related to the issue of sites below saturation discussed before. However, this deviation does not have any practical consequences since it represents a conservative bias in the context of RNA folding controls and, more importantly, seems to be too small to have any noticeable effect at all.

4.2 Influence of randomization procedure on RNA predictions

The main motivation of this paper is to provide dinucleotide based controls for comparative RNA gene predictions. Therefore, we ran *RNAalifold* and *RNAz* on the alignments to demonstrate how different randomization procedures affect the results. Fig. 8A shows the distribution of *RNAalifold* consensus MFEs on the genomic alignments and their different randomizations. One can see that the genomic alignments show the most stable structures. There is a clear difference between the native genomic alignments and the shuffled and mononucleotide simulated ones. However, the folding energies of the dinucleotide simulated alignments are much closer to the native data. This difference between the di- and mononucleotide simulations reflects the bias caused by the genomic dinucleotide content. The difference between the native and the dinucleotide

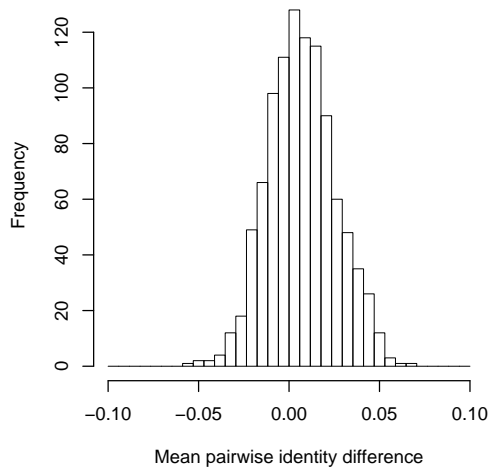


Figure 7: Mean pairwise identity in randomized genomic alignments. The distribution of the difference of the mean pairwise identity between the original genomic alignments and the simulated ones (dinucleotide model) is shown.

controls indicates the existence of RNA signals in the genome or, alternatively, another as yet unidentified bias.

Clearly, the differences shown here in these cumulative histograms might appear very subtle. The results for the **RNAz** predictions, however, show that such differences can strongly affect the statistics of RNA gene predictions (Fig. 8B). On this particular test set, **RNAz** predicts RNA signals in 4.3% of the native alignments. Using the conventional shuffling strategy or mononucleotide based simulation one would estimate a false positive rate of 0.8% or 0.7%, respectively. Using the more conservative dinucleotide based model the estimate would be 2.1%, i.e. three times higher. This is consistent with the results obtained by Babak *et al.* using their dinucleotide shuffling approach on pairwise alignments.

4.3 Calculating z-scores

We can directly assess the significance of a predicted RNA by calculating a z -score. The folding energy of the native data m and the mean μ and standard deviation σ of randomized data is calculated. The significance of the native fold can then be expressed as $z = (m - \mu)/\sigma$, i.e. the number of standard deviations from the mean. This score has been repeatedly used on single sequences applying mono- or dinucleotide shuffling or simulation using a zero or first order Markov model [23, 24]. Using shuffled alignments as null model, this approach is implemented in **AlifoldZ** [5]. The same strategy can be used in combination with our new dinucleotide base randomization strategy without any further modifications (Fig. 4).

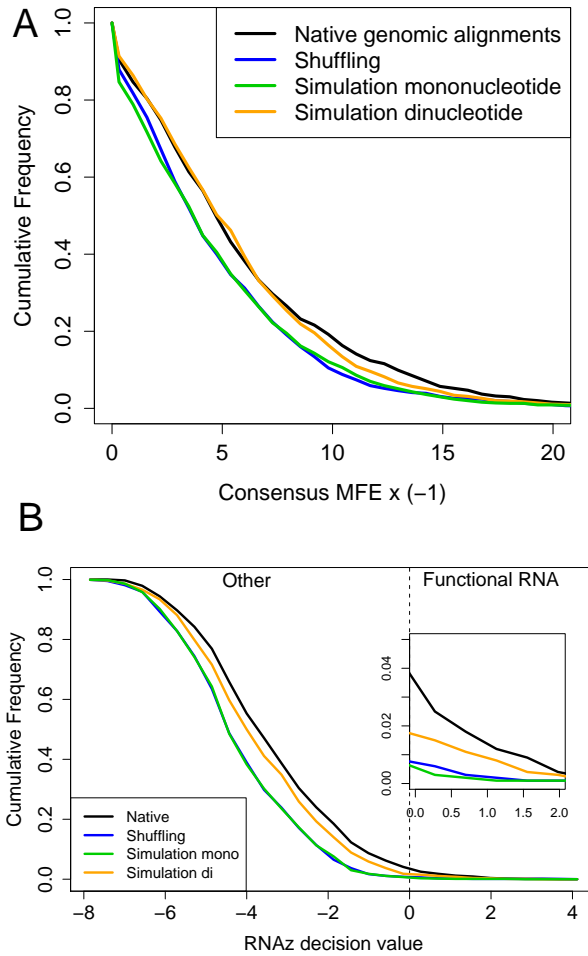


Figure 8: Influence of the randomization procedure on RNA predictions. **(A)** Cumulative frequency distribution of RNAalifold consensus folding energies for the native and randomized alignments. **(B)** Cumulative frequency distribution of RNAz scores. The “decision-value” is the result of the support vector machine classification. Positive values indicate a potential functional RNA while negative values indicate no significant fold. The positive tail is magnified.

We calculated z -scores for the genomic background alignments of chromosome 1 and three sets of well annotated structural RNAs (H/ACA snoRNAs, miRNAs and tRNAs). A sample size of 1000 randomizations was used. The results are summarized in Tab. 1. Using mononucleotide based randomization the z -scores of the genomic background are approximately half a standard deviation from zero (-0.45 and -0.54 , for shuffling and mononucleotide simulation respectively). Albeit the signal does not vanish completely, the dinucleotide based z -scores are much closer to zero (-0.12). As a result, the fraction of predicted RNAs is lower. For the two z -score cutoffs of -2 and -3 we observe three times fewer predictions, consistent with the RNAz results discussed before.

For the structural RNAs, we also observe that mononucleotide simulated z -scores are lower than the dinucleotide simulated ones. Here, a dinucleotide content that favors stable RNA structures is clearly not only a general background effect of the genomic base composition but a feature of structural RNAs. However, this signal is lost if the more conservative dinucleotide based null model is used.

Interestingly, on the structural RNAs the z -scores obtained by shuffling are the most significant and generally below the z -scores obtained by mononucleotide simulation. This is remarkable because on structural RNAs shuffling has the disadvantage that the signal from compensatory mutations is not entirely removed. The number of “compatible” columns that can form a consensus base-pair remains unchanged. Therefore, the average consensus MFE is closer to the original one as can be seen for example in the case of the snoRNAs that have many compensatory mutations. This should result in a conservative z -score closer to zero, the opposite of what we observe. The reason is the different standard deviations which is lower for shuffling than for simulation.

This shows that there are many effects that have to be taken into account. To test how the overall classification performance of a z -score based RNA gene prediction approach depends on the null model, we generated receiver operating characteristic curves for the different approaches (Fig. 9). We ask how well the true ncRNAs can be distinguished from the background alignments from chromosome 1. Indeed, the dinucleotide based model shows a better overall classification performance than mononucleotide simulation or shuffling. However, in the regions of high specificity which are the only relevant regions for practical applications, we do not see a marked difference between the methods in terms of sensitivity. At a false positive rate of 0.05 we obtain sensitivities of 0.84, 0.84 and 0.88 for mononucleotide shuffling, mononucleotide simulation and dinucleotide simulation, respectively. At a false positive rate of 0.01 all methods essentially perform equally well (see zoomed ROC curve in Fig. 9).

Table 1: z-scores statistic of genomic background and structured RNAs

Data type	Null model	Consensus MFE ¹			z^1	Fraction below cutoff		
		Native	Mean	Stdv.		-2	-3	-4
Background	Shuffling	-5.50	-4.55	1.92	-0.45	0.09	0.03	0.01
	SISSI mono	-5.50	-4.19	2.16	-0.54	0.09	0.03	0.01
	SISSI di	-5.50	-5.09	2.46	-0.12	0.03	0.01	0.00
snoRNA	Shuffling	-34.69	-19.41	3.12	-4.97	0.86	0.77	0.61
	SISSI mono	-34.69	-17.13	3.84	-4.72	0.90	0.78	0.65
	SISSI di	-34.69	-17.99	4.09	-4.17	0.82	0.73	0.52
miRNA	Shuffling	-30.84	-13.85	2.75	-6.18	0.94	0.92	0.85
	SISSI mono	-30.84	-13.04	3.15	-5.80	0.95	0.90	0.83
	SISSI di	-30.84	-14.19	3.38	-5.05	0.94	0.85	0.74
tRNA	Shuffling	-29.80	-23.16	3.15	-2.12	0.49	0.13	0.00
	SISSI mono	-29.80	-22.98	3.50	-1.95	0.49	0.07	0.00
	SISSI di	-29.80	-23.24	3.66	-1.80	0.36	0.04	0.00

¹The average of all examples in the test set is shown.

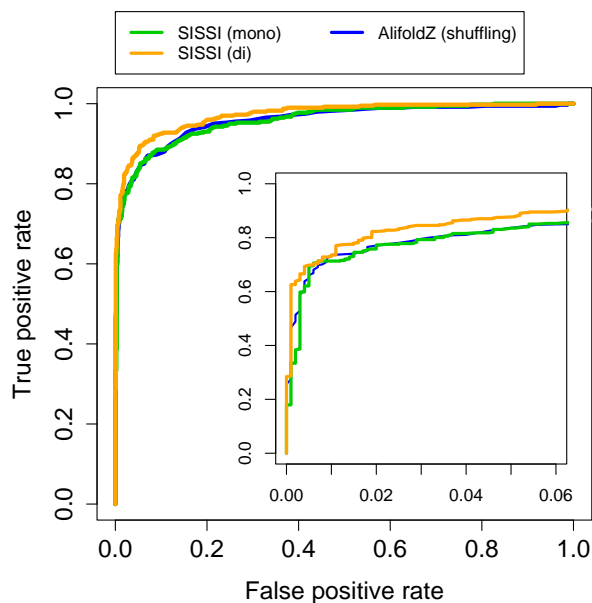


Figure 9: Accuracy of z-score based classification of structured RNAs. As a positive examples we used tRNAs, H/ACA snoRNAs, and miRNAs. As negative examples we used alignments randomly chosen from the genome. ROC curves are shown in dependence on the null model used. The relevant high specificity area is magnified.

5 Discussion

Any experiment is only as good as its controls. What is true for experimental biology clearly also holds in the field of computational biology. The value of even the most sophisticated algorithm remains unclear if the significance of the results cannot be assessed properly. In this paper we addressed the problem of finding an adequate control strategy for comparative noncoding RNA predictions, which are started to get widely used for genome annotation.

Babak *et al.* demonstrated that currently used null models based on mononucleotide shuffling lead to an underestimation of the false positive rate in such screens. Although single opinions may be different [49], it is generally accepted that in the context of RNA gene prediction one should consider dinucleotide content as “background” rather than “signal”. However, while there have been dinucleotide controlled randomization algorithms for single sequences for more than 20 years, it is a non-trivial problem in the case of multiple sequence alignments.

Here we devised a simulation procedure that produces alignments that have on average a given dinucleotide frequency and sequence diversity (globally and locally). The corresponding model needs to be relatively complex including overlapping dependencies and site-specific rates. Clearly, this model with a high number of parameters would not be a reasonable choice for use in phylogenetic analysis, but it turned out to be a good choice for this specific application.

We have to use heuristics and simplifications to estimate the tree and parameters for this model in reasonable time. The accuracy of our approach is measured in terms of how well the simulations reflect the properties of the original data. In this respect, we found that our strategy performs very well. Again, phylogenetic analysis was not the goal here, but some of the techniques introduced here might be of interest in this context. For example, we found that in the mononucleotide case our estimations for site-specific rates are surprisingly competitive when compared to the currently best maximum likelihood methods (data not shown).

The influence of the null model for genomic RNA predictions was found to be remarkable. Consistent with Babak and colleagues’ findings on pairwise alignments, we observed three times more false positives using dinucleotide controls than using mononucleotide controls. This clearly shows that the new approach should be the method of choice to get more sensible estimates of the significance of comparative RNA predictions.

Following the idea of `AlifoldZ`, we can use our new null model to directly predict significant RNA structures. The z -scores calculated this way are not biased by the dinucleotide content. `SISSIZ` is thus the first comparative gene finding program, that explicitly corrects for this effect. However, by using

this conservative null model we also lose part of the signal in true structured RNAs. This might be the main reason, why we observed only minor improvements in the overall classification performance compared to `AlifoldZ`. This does not necessarily generalize to other gene finders. `AlifoldZ` generally gives very significant results on well conserved RNAs as used in the test here and is relatively specific. On this test set which is relatively easy to classify (sensitivities higher than 80% at reasonable specificities) the choice of the null model might not be that critical. However, in more difficult cases with only weak RNA signals a dinucleotide model could give more improvements. In such cases the `RNAz` algorithm is preferable. However, there is still no `RNAz` version based on a dinucleotide model, mainly for two reasons. First, it is still an unsolved question how to compute dinucleotide based z -scores efficiently without shuffling. Second, so far there was no way to produce a dinucleotide controlled negative test set that is necessary for training of the support vector machine [6]. With the method presented here, at least the second problem is now solved.

Finally, we want to add that our randomization algorithm is not only of interest in the context of RNA gene prediction. It can be used for other comparative genomics applications whenever random alignments are needed as control. One could consider other applications in the context of RNA structures (e.g. prediction of conserved miRNA target sites) but also in different context (e.g. conserved sequence motifs). Currently our software implements a mono- and dinucleotide model which should be sufficient for many applications. In principle, however, it is also possible to consider higher order correlations within this framework.

6 Methods

6.1 Treating gaps

Gapped positions are treated as missing data. When counting the dinucleotide content, dinucleotides including a gap (N-, -N, --) are ignored. During simulation, gap positions are filled with nucleotides and gaps are re-introduced afterwards. Note that this way, if two nucleotides N_1 and N_2 are separated by a gap (e.g. $N_1\text{---}N_2$) the dinucleotide N_1N_2 is not in equilibrium. Depending on how gaps are treated in the downstream analysis this might be or might not be of concern. In any case, since not every gap position but only every gap *opened* is affected, this (potential) error is generally very small for reasonable alignments. So we did not consider correcting for this effect which would require reconstructing the gap history and setting lineage specific neighbourhood systems.

For calculating the site-specific rates, we also treat gaps as missing data and calculate $\langle p_k \rangle$ in eq. 10 only over non-gap positions. After the simulation, the whole column has on average $\langle p_k \rangle$ estimated from the non-gap positions that does not change when originally gapped positions are masked again. For calculating

the observed differences p between two sequences we set positions that includes gaps to the average $\langle p_k \rangle$ at this site.

6.2 Distances above the level of saturation

When calculating genetic distances between two sequences the problem may occur that the observed number of differences is higher than the level of saturation. We found that this problem becomes severe when considering site-specific rates that generally lead to much lower levels of saturation (cf. Fig. 3B). We use a simple trick to overcome this limitation. We add additional sites during the simulation with site-specific rates that correspond to the average of the whole alignment (i.e. $\langle p_k \rangle$ is set to 1-MPI in eq. 11 for all these additional sites). They act as “buffer sites” that reduces the number of mutation events that repeatedly hit the same sites of high rate leading to many double substitutions. As a consequence, the overall level of observed differences is higher and we do not run into problems building the distance matrix. In the end, the sites are removed again and since the relative rate ratios between the sites remained unchanged, we get the desired site-specific mutation patterns.

6.3 Limiting base composition variation

During the testing of the influence of the randomization procedure on RNA folding, we made an interesting observation. As expected, the variance of the folding energies of randomized data is higher with simulation than with shuffling. However, we also observed that there is difference in the mean. Simulation leads to slightly higher (i.e. less stable) folding energies than shuffling. We observed this behaviour not only on multiple alignments but also on single sequences using shuffling vs. first order Markov simulation. We suspect that extreme deviations in the base composition that can occur in simulated data do not symmetrically lead to the same deviations of the folding energies but preferentially impair the formation of RNA structures. To compensate for this effect, we have introduced an option in our software that only outputs simulated alignments, that are within a specific range of mononucleotide frequencies. We can thus limit our simulations to mononucleotide frequencies that are almost exactly as in the original data. As a distance measure we use the Euclidean distance $\sum_{\alpha \in A,G,C,T} \sqrt{(\pi_{\alpha}^s - \pi_{\alpha})^2}$ with π_{α} the desired frequency of nucleotide α in the original alignment, and π_{α}^s the observed frequency in the simulation. For all the data shown in Figs. 8,9 and Tab. 1 we used simulations with this cutoff set to 0.05.

6.4 Software

For the simulations in Figs. 1 we used `seq-gen` version 1.3.2 (<http://tree.bio.ed.ac.uk/software/seqgen>, [50]) Mononucleotide shuffling was carried out using `shuffle-aln.pl` with option “`--mode conservative2`”. Together with `alifoldz.pl` it is available here: <http://www.tbi.univie.ac.at/papers/SUPPLEMENTS/Alifoldz>. For the tests in Figs. 1 and 8 and we used `RNAalifold` from the Vienna RNA package version 1.6.1. with options “`-nc 0 -cv 0`” (www.tbi.univie.ac.at/~ivo/RNA, [51]) and `RNAz` version 1.0 with standard parameters (www.tbi.univie.ac.at/~wash/RNAz). For implementation of our software we used a series of third party C-code that is available as open source: `levmar` by Manolis Lourakis (<http://www.ics.forth.gr/~lourakis/levmar>) for least squares fitting, `BIONJ` by Olivier Gascuel (<http://www.crt.umontreal.ca/~olivierg/bionj.html>, [41]), `PHYML` by Stéphane Guindon and Olivier Gascuel (<http://atgc.lirmm.fr/phyml>, [52]) for maximum likelihood estimation of the transition/transversion rate, Vienna RNA package by Ivo L. Hofacker and others for consensus folding in `SISSIZ`.

6.5 Genomic data

The genomic example alignments were extracted from Multiz 17-way vertebrate alignments available at the UCSC genome browser (genome.ucsc.edu, [53]). We used the `rnazWindow.pl` script from the `RNAz` software package (www.tbi.univie.ac.at/~wash/RNAz, [54]) to get typical alignment blocks as used previously in genomic ncRNA screens (e.g. [20] or [14]). For set of structural RNAs we extracted annotated H/ACA snoRNAs, miRNAs and tRNAs (excluding any pseudogene annotations). We chose this set of RNAs because it is currently the only reliable annotation of known ncRNAs with a reasonable number of examples and relatively different characteristics. It should be noted, that the tRNAs are poorly represented in the alignments used (either no homologous sequences at all, or very closely related species are aligned). Therefore they only give relatively insignificant z -scores in Tab. 1 in contrast to previous studies [5].

The test alignments are available for download here: www.tbi.univie.ac.at/papers/SUPPLEMENTS/SISSIZ.

7 Authors contributions

Both authors contributed equally.

8 Acknowledgments

We thank Arndt von Haeseler, Ivo Hofacker, and Nick Goldman for discussions and Roland Fleißner for comments on the manuscript. We acknowledge funding from the Austrian GEN-AU projects “noncoding RNA” and “Bioinformatics Integration Network” and Financial support to the CIBIV institute from the Wiener Wissenschafts-, Forschungs- and Technologiefonds (WWTF).

References

1. Griffiths-Jones S: **Annotating noncoding RNA genes**. *Annu Rev Genomics Hum Genet* 2007, **8**:279–298.
2. Backofen R, Bernhart SH, Flamm C, Fried C, Fritzsich G, Hackermuller J, Hertel J, Hofacker IL, Missal K, Mosig A, Prohaska SJ, Rose D, Stadler PF, Tanzer A, Washietl S, Will S: **RNAs everywhere: genome-wide annotation of structured RNAs**. *J Exp Zool B Mol Dev Evol* 2007, **308**:1–25.
3. Rivas E, Eddy SR: **Noncoding RNA gene detection using comparative sequence analysis**. *BMC Bioinformatics* 2001, **2**:8–8.
4. Coventry A, Kleitman DJ, Berger B: **MSARI: multiple sequence alignments for statistical detection of RNA secondary structure**. *Proc Natl Acad Sci U S A* 2004, **101**(33):12102–12107.
5. Washietl S, Hofacker IL: **Consensus folding of aligned sequences as a new measure for the detection of functional RNAs by comparative genomics**. *J Mol Biol* 2004, **342**:19–30.
6. Washietl S, Hofacker IL, Stadler PF: **Fast and reliable prediction of noncoding RNAs**. *Proc Natl Acad Sci U S A* 2005, **102**(7):2454–2459.
7. Pedersen JS, Bejerano G, Siepel A, Rosenbloom K, Lindblad-Toh K, Lander ES, Kent J, Miller W, Haussler D: **Identification and classification of conserved RNA secondary structures in the human genome**. *PLoS Comput Biol* 2006, **2**(4).
8. Yao Z, Weinberg Z, Ruzzo WL: **CMfinder—a covariance model based RNA motif finding algorithm**. *Bioinformatics* 2006, **22**(4):445–452.
9. Uzilov AV, Keegan JM, Mathews DH: **Detection of non-coding RNAs on the basis of predicted secondary structure formation free energy change**. *BMC Bioinformatics* 2006, **7**:173.
10. Torarinsson E, Sawera M, Havgaard JH, Fredholm M, Gorodkin J: **Thousands of corresponding human and mouse genomic regions unalignable in primary sequence contain common RNA structure**. *Genome Res* 2006, **16**(7):885–9.
11. Washietl S, Hofacker IL, Lukasser M, Hüttenhofer A, Stadler PF: **Mapping of conserved RNA secondary structures predicts thousands of functional noncoding RNAs in the human genome**. *Nat Biotechnol* 2005, **23**(11):1383–1390.
12. Missal K, Rose D, Stadler PF: **Non-coding RNAs in *Ciona intestinalis***. *Bioinformatics* 2005, **21 Suppl 2**:77–78.
13. Missal K, Zhu X, Rose D, Deng W, Skogerbo G, Chen R, Stadler PF: **Prediction of structured non-coding RNAs in the genomes of the nematodes *Caenorhabditis elegans* and *Caenorhabditis briggsae***. *J Exp Zool B Mol Dev Evol* 2006, **306**(4):379–392.
14. Rose D, Hackermueller J, Washietl S, Reiche K, Hertel J, Findeiss S, Stadler PF, Prohaska SJ: **Computational RNomics of *Drosophilids***. *BMC Genomics* 2007, **8**:406.
15. Axmann IM, Kensche P, Vogel J, Kohl S, Herzog H, Hess WR: **Identification of cyanobacterial non-coding RNAs by comparative genome analysis**. *Genome Biol* 2005, **6**(9).
16. Weile C, Gardner PP, Hedegaard MM, Vinther J: **Use of tiling array data and RNA secondary structure predictions to identify noncoding RNA genes**. *BMC Genomics* 2007, **8**:244–244.
17. del Val C, Rivas E, Torres-Quesada O, Toro N, Jiménez-Zurdo JI: **Identification of differentially expressed small non-coding RNAs in the legume endosymbiont *Sinorhizobium meliloti* by comparative genomics**. *Mol Microbiol* 2007, **66**(5):1080–1091.

18. Mourier T, Carret C, Kyes S, Christodoulou Z, Gardner PP, Jeffares DC, Pinches R, Barrell B, Berriman M, Griffiths-Jones S, Ivens A, Newbold C, Pain A: **Genome-wide discovery and verification of novel structured RNAs in *Plasmodium falciparum***. *Genome Res* 2007.
19. Sandmann T, Cohen SM: **Identification of Novel *Drosophila melanogaster* MicroRNAs**. *PLoS ONE* 2007, 2(11).
20. Washietl S, Pedersen JS, Korbelt JO, Stocsits C, Gruber AR, Hackermüller J, Hertel J, Lindemeyer M, Reiche K, Tanzer A, Ucla C, Wyss C, Antonarakis SE, Denoeud F, Lagarde J, Drenkow J, Kapranov P, Gingeras TR, Guigó R, Snyder M, Gerstein MB, Reymond A, Hofacker IL, Stadler PF: **Structured RNAs in the ENCODE selected regions of the human genome**. *Genome Res* 2007, 17(6):852–864.
21. Babak T, Blencowe BJ, Hughes TR: **Considerations in the identification of functional RNA structural elements in genomic alignments**. *BMC Bioinformatics* 2007, 8:33.
22. Altschul SF, Erickson BW: **Significance of nucleotide sequence alignments: a method for random sequence permutation that preserves dinucleotide and codon usage**. *Mol Biol Evol* 1985, 2(6):526–538.
23. Clote P, Ferré F, Kranakis E, Krizanc D: **Structural RNA has lower folding energy than random RNA of the same dinucleotide frequency**. *RNA* 2005, 11(5):578–591.
24. Workman C, Krogh A: **No evidence that mRNAs have lower folding free energies than random sequences with the same dinucleotide distribution**. *Nucleic Acids Res* 1999, 27(24):4816–4822.
25. Hasegawa M, Kishino H, Yano T: **Dating of the Human–Ape Splitting by a Molecular Clock of Mitochondrial DNA**. *J Mol Evol* 1985, 22:160–174.
26. Hofacker IL, Fekete M, Stadler PF: **Secondary structure prediction for aligned RNA sequences**. *J Mol Biol* 2002, 319(5):1059–1066.
27. Felsenstein J: *Inferring Phylogenies*. Sunderland, Massachusetts: Sinauer Associates 2004.
28. Tavaré S: **Some probabilistic and statistical problems on the analysis of DNA sequences**. *Lec. Math. Life Sci.* 1986, 17:57–86.
29. Jensen J, Pedersen AM: **Probabilistic models of DNA sequence evolution with context dependent rates of substitution**. *Adv Appl Prob* 2000, 32:499–517.
30. Duret L, Galtier N: **The covariation between TpA deficiency, CpG deficiency, and G+C content of human isochores is due to a mathematical artifact**. *Mol Biol Evol* 2000, 17(11):1620–1625.
31. Pedersen AM, Jensen J: **A dependent rates model and MCMC based methodology for the maximum likelihood analysis of sequences with overlapping reading frames**. *Mol Biol Evol* 2001, 18:763–776.
32. Arndt PF, Burge CB, Hwa T: **DNA sequence evolution with neighbor-dependent mutation**. *J Comput Biol* 2003, 10:313–322.
33. Robinson DM, Jones DT, Kishino H, Goldman N, Thorne JL: **Protein Evolution with Dependence Among Codons Due to Tertiary Structure**. *Mol Biol Evol* 2003, 20:1692–1704.
34. Siepel A, Haussler D: **Phylogenetic Estimation of Context-Dependent Substitution Rates by Maximum Likelihood**. *Mol Biol Evol* 2004, 21:468–488.
35. Lunter G, Hein J: **A nucleotide substitution model with nearest-neighbour interactions**. *Bioinformatics* 2004, 20:i216–i223.
36. Christensen OF: **Pseudo-likelihood for non-reversible nucleotide substitution models with neighbor dependent rates**. *Stat. Appl. Genet. Mol. Biol.* 2006, 5:1–29.
37. Gesell T, von Haeseler A: **In silico sequence evolution with site-specific interactions along phylogenetic trees**. *Bioinformatics* 2006, 22:716–722.
38. Schöniger M, von Haeseler A: **A Stochastic Model for the Evolution of Autocorrelated DNA sequences**. *Mol Phylogenet Evol* 1994, 3:240–247.
39. Schöniger M, von Haeseler A: **Simulating efficiently the evolution of DNA sequences**. *Comput Appl Biosci* 1995, 11:111–115.
40. Lanave C, Preparata G, Saccone C, Serio G: **A new method for calculating evolutionary substitution rates**. *J Mol Evo* 1984, 20:86–93.

41. Gascuel O: **BIONJ: an improved version of the NJ algorithm based on a simple model of sequence data.** *Mol Biol Evol* 1997, **14**(7):685–695.
42. Van de Peer Y, Baldauf SL, Doolittle WF, Meyer A: **An updated and comprehensive rRNA phylogeny of (crown) eukaryotes based on rate-calibrated evolutionary distances.** *J Mol Evol* 2000, **51**(6):565–576.
43. Thorne J, Kishino H, Felsenstein J: **An Evolutionary Model for Maximum Likelihood Alignment of DNA Sequences.** *J Mol Evol* 1991, **33**:114–124.
44. Thorne J, Kishino H, Felsenstein J: **Inching toward reality: An improved likelihood model of sequence evolution.** *J Mol Evol* 1992, **34**:3–16.
45. Metzler D: **Statistical alignment based on fragment insertion and deletion models.** *Bioinformatics* 2003, **19**:490–499.
46. Fleißner R, Metzler D, von Haeseler A: **Simultaneous Statistical Alignment and Phylogeny Reconstruction.** *Syst Biol* 2005, **54**:548–561.
47. Löytynoja A, Goldman N: **Phylogeny-aware alignments change perception of the roles of insertions and deletions in sequence evolution.** *Submitted* 2008.
48. Blanchette M, Kent WJ, Riemer C, Elnitski L, Smit AF, Roskin KM, Baertsch R, Rosenbloom K, Clawson H, Green ED, Haussler D, Miller W: **Aligning multiple genomic sequences with the threaded blockset aligner.** *Genome Res* 2004, **14**(4):708–715.
49. Forsdyke DR: **Calculation of folding energies of single-stranded nucleic acid sequences: conceptual issues.** *J Theor Biol* 2007, **248**(4):745–753.
50. Rambaut A, Grassly NC: **Seq-Gen: an application for the Monte Carlo simulation of DNA sequence evolution along phylogenetic trees.** *Comput Appl Biosci* 1997, **13**(3):235–238.
51. Hofacker IL, Fontana W, Stadler PF, Bonhoeffer LS, Tacker M, Schuster P: **Fast folding and comparison of RNA secondary structures.** *Monatsh Chem* 1994, **125**:167–188.
52. Guindon S, Gascuel O: **A simple, fast, and accurate algorithm to estimate large phylogenies by maximum likelihood.** *Syst Biol* 2003, **52**(5):696–704.
53. Karolchik D, Kuhn RM, Baertsch R, Barber GP, Clawson H, Diekhans M, Giardine B, Harte RA, Hinrichs AS, Hsu F, Kober KM, Miller W, Pedersen JS, Pohl A, Raney BJ, Rhead B, Rosenbloom KR, Smith KE, Stanke M, Thakkapallayil A, Trumbower H, Wang T, Zweig AS, Haussler D, Kent WJ: **The UCSC Genome Browser Database: 2008 update.** *Nucleic Acids Res* 2007.
54. Washietl S: **Prediction of Structural Noncoding RNAs With RNAz.** *Methods Mol Biol* 2007, **395**:503–526.