STATISTICAL ANALYSIS OF RNA BACKBONE

By

Guillermo Sapiro

Eli Hershkovitz

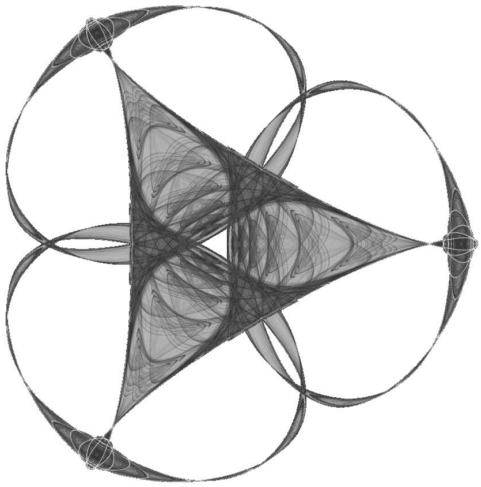
Allen Tannenbaum

and

Loren Dean Williams

IMA Preprint Series # 1964

(February 2004)



INSTITUTE FOR MATHEMATICS AND ITS APPLICATIONS

UNIVERSITY OF MINNESOTA

514 Vincent Hall
206 Church Street S.E.
Minneapolis, Minnesota 55455–0436

Phone: 612/624-6066 Fax: 612/626-7370
URL: http://www.ima.umn.edu

Statistical Analysis of RNA Backbone*

Guillermo Sapiro†
Eli Hershkovitz and **Allen Tannenbaum**‡
Loren Dean Williams§

Abstract

RNA backbone conformation analysis has been demonstrated to be particularly difficult due to the large number of torsion angles per residue and the large variability of the raw data. Due in part to the importance of local structures in the understanding of RNA catalysis and binding functions, studies in this area have recently received increased attention. In this work we use classical tools from statistics and signal processing to search for clusters in the RNA backbone torsion angles. Results are reported both for scalar studies, where each torsion angle is separately studied, and for vectorial studies, where several angles are simultaneously clustered. Using techniques from optimal quantization, we automatically find the torsion angle clusters. With these clustering techniques, we find RNA backbone motifs, both at the single residue level (phosphate-to-phosphate) and at the suites level (base-to-base) parsing. These two parsing techniques are also compared using mutual information measurements. We conclude the work with statistical analysis of some of these motifs, and optimal fitting of torsion angle distributions in the most significant clusters. The whole process is fully automatic and based on well-defined optimality criteria.

1 Introduction

RNA plays an important role in storage and communication of information, as well as in other important biological processes. As with proteins, the 3D structure of RNA is essential for performing these functions. The 3D structure of RNA is different than that of proteins, with six torsion angles in each residue; see Figure 1.

The work described here follows recent efforts in studying the local 3D structure of RNA, e.g., [5, 9, 10, 11]. In this paper we use classical techniques from statistical signal processing to study the RNA torsion angles, which are illustrated in Figure 1; see also [15]. We present fully automatic techniques to search for motifs (conformers/rotamers) in the RNA backbone, both at the level of individual residues or suites and at the level of a group of consecutive ones. Note that in [5], we considered the problem of finding repeating conformational states (*conformational motifs*) and representing them as repeating strings of ASCII characters. The use of quantization makes the recent approaches of [5, 9] fully automatic and based on well defined distortion and quality metrics.[1] Additional statistical analysis techniques demonstrated in this paper are mutual information to compare between residue and suite parsing, optimal fitting of the main torsion angle clusters, and principal component analysis of key found motifs.

2 Scalar and Vector Quantization

In this section, we briefly describe the basic concepts of vector quantization that we will use for clustering. Details on this technique can be found, e.g., in [2], from which we have prepared the summary we now present. Note that in this work we restrict ourselves to the use of this clustering technique, while in the future we plan to use more advanced ones such as those reported in [12].[2]

Vector quantization (VQ) is a clustering technique originally developed for lossy data compression. In 1980, Linde *et al.*, [8], proposed a practical VQ design algorithm based on a training sequence. The use of a training sequence by-passes the need for multi-dimensional integration, thereby making VQ a practical technique, implemented in most scientific computation packages, such as Matlab (www.mathworks.com).

A VQ is nothing more than an approximator. The idea

*Work supported by ONR, DARPA, NSF, ARO, AFOSR, and NIH.

†Electrical and Computer Engineering and Digital Technology Center, University of Minnesota, Minneapolis, MN 55455, guille@ece.umn.edu

‡Schools of Electrical & Computer Engineering and Biomedical Engineering, Georgia Institute of Technology, Atlanta, GA 30332-0250, eli@theor.chemistry.gatech.edu, tannenba@ece.gatech.edu.

§School of Chemistry and Biochemistry, Georgia Institute of Technology, Atlanta, Georgia 30332, loren.williams@chemistry.gatech.edu.

[1] Vector quantization was used in the context of protein structure; e.g., [6].

[2] We should also note that vector quantization is often also known in the literature as k-means clustering.

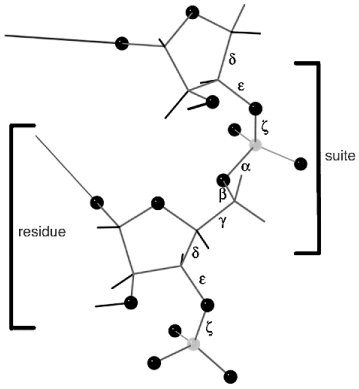


Figure 1: *RNA backbone with six torsion angles labeled on the central bond of the four atoms defining each dihedral. The two alternative ways of parsing out a repeat are indicated: A traditional nucleotide residue goes from phosphate to phosphate (changing residue number between O5' and P), whereas an RNA suite, which is more appropriate for local geometry analysis, goes from sugar to sugar (or base to base). Only the angles α, γ, δ, and ζ are investigated in this study. This image was obtained from [9], where the reader is directed for a detailed description of the reasons for using both parsing approaches.*

is similar to that of "rounding-off" (say to the nearest integer). An example of a 1-dimensional VQ is shown in Figure 2. Here, every number less than -2 are approximated by -3. Every number between -2 and 0 are approximated by -1. Every number between 0 and 2 are approximated by +1. Every number greater than 2 are approximated by +3. Figure 2 also presents a two-dimensional example. Here, every pair of numbers falling in a particular region are approximated by the red star associated with that region.

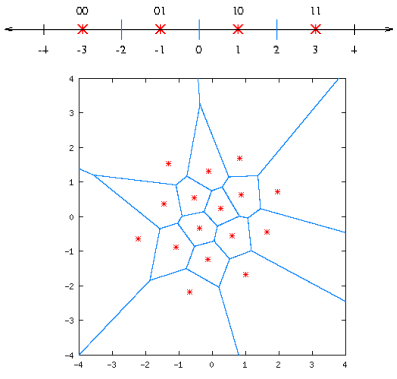


Figure 2: *One (top) and two (bottom) dimensional examples of clustering via (vector) quantization. All the points in a given interval (in one-dimension) or a given cell (two-dimensions) are represented by the red marked "center." (This is a color figure.)*

The VQ design problem can be stated as follows. Given a vector source with its statistical properties known, given a distortion measure, and given the number of desired codevectors, find a codebook (the set of all red stars) and a partition (the set of blue lines) which result in the smallest average distortion.

We assume that there is a training sequence (e.g., the measured torsion angles in RNA backbone) consisting of M source vectors of the form $T = \{x_1, x_2, ..., x_M\}$. We assume that the source vectors are k-dimensional, e.g., $x_m = \{x_{m,1}, x_{m,2}, ..., x_{m,k}\}$, for $1 \leq m \leq M$. Let N be the number of desired codevectors and let $C = \{c_1, c_2, ..., c_N\}$ be the codebook, where each c_n, $1 \leq n \leq N$, is of course k-dimensional as well. Let S_n be the cell associated with the codevector c_n and let $P = \{S_1, S_2, ..., S_N\}$ be the corresponding partition of the k-dimensional space. If the source vector x_m is in the encoding region S_n, then its approximated by c_n, and let denote by $Q(x_m) = c_n$ (if $x_m \in S_n$) such a map. Then, assuming for example a squared error distortion measure, the average distortion is given by $D = \frac{1}{Mk} \sum_{m=1} M \parallel x_m - Q(x_m) \parallel^2$, where $\parallel e \parallel^2 = e_1^2 + e_2^2 + ... + e_k^2$.

The design problem then becomes the following: Given the training data set T and the number of desired codebooks (or clusters) N, find the cluster centers C and the space partition P such that the distortion D is minimized. This problem can be efficiently solved with the LBG algorithm [4, 8], and as mentioned above, its implementation can be found in most of the popular scientific computing programs.

3 Clustering the RNA Backbone Torsion Angles

We first report results from scalar quantization, where each one of the angles are studied separately. Once this is done, we will analyze all torsion angles as a vector. We use two data sets. One follows the work reported in [5], and is for a single RNA with 2914 residues (HM LSU 23S rRNA, rr0033), while the second one follows work reported in [9], and is for a collection of 132 RNAs,[3] giving a total of 10463 residues. Here, as in the rest of this work, residues with unknown torsion angles were ignored in the analysis. The data was obtained from the *Nucleic Acid Database* [13]. Although we have not performed the

[3]With NDB and PDB codes: ar0001, 02, 04, 05, 06, 07, 08, 09, 11, 12, 13, 20, 21, 22, 23, 24, 27, 28, 30, 32, 36, 38, 40, 44; arb002, 3, 4, 5; arf0108; arh064, 74; arl037, 48, 62; arn035; dr0005, 08, 10; drb002, 03, 05, 07, 08, 18; drd004; pd0345; pr0005, 06, 07, 08, 09, 10, 11, 15, 17, 18, 19, 20, 21, 22, 26, 30, 32, 33, 34, 36, 37, 40, 46, 47, 51, 53, 55, 57, 60, 62, 63, 65, 67, 69, 71, 73, 75, 78, 79, 80, 81, 83, 85, 90, 91; prv001, 04, 10, 20, 21; pte003; ptr004, 16; rr0005, 10, 16, 19, 33; tr0001; trna12; uh0001; uhx026; ur0001, 04, 05, 07, 09, 12, 14, 15, 19, 20, 22, 26; urb003, 08, 16; urc002; urf042; url029, 50; urt068; and urx053, 59, 63, 75.

filtering techniques in [9], these might be used to improve our results. As in [5], we here limit the analysis to the torsion angles α, γ, δ, ζ (see Figure 1), since the other ones are either dependent with respect to these ones or have unimodal distributions [14, 16]. There is no intrinsic limitation in our technique in working only with this reduced set of angles (moreover, being the process fully automatic, the work can certainly be carried out for larger sets), but this will clarify the presentation.

In Figure 3 we show the distributions for these four angles for the two datasets. A few remarkable things to notice are the following. First, the distributions are very similar for both datasets, pointing out to the fact that the local structures are not only "rotameric" for a given RNA (first data set) but also across RNAs (second dataset). Secondly, although the distributions for α and ζ are very similar (since these can be considered analogous angles), the secondary picks for ζ are much broader and less well defined, Figure 4. This has been the subject of controversy, and for example, the authors of [9] solve this by filtering, and then reporting more clusters than in the non-filtered approach in [5]. Still, although this filtering is important in the analysis, it doesn't explain the unique long tail in the ζ distribution; see also [15]. In particular, note that the rotation of ζ is sterically more restricted than that of α by proximity to the furanose ring. Here, we will limit our analysis (see below) to what the VQ statistical analysis tells us, working with the raw data and without any additional constraints. Understanding this difference between the α and ζ torsion angles is something that intrigues us and we hope to address in the near future.

Using the automatic and optimal quantization technique, and requesting the number C of codevectors following [5] (or just from visual inspection) we found the codevectors or centers of the clusters given in Table 1.

	Dataset 1
α	68.3 (1), 169.7 (2), 294.3 (3)
γ	50.4, 60.0 (1), 175.8 (2), 292.3 (3)
δ	81.7 (1), 147.8 (2)
ζ	118.0 (2), 286.7 (1)
	Dataset 2
α	68.6 (1), 167.8 (2), 294.0 (3)
γ	50.1, 65.0 (1), 174.4 (2), 290.2 (3)
δ	82.7 (1), 144.4 (2)
ζ	116.4 (2), 286.0 (1)

Table 1: *Cluster centers automatically computed by our technique. Numbers in parenthesis are used for cluster identification.*

We note once again the very similar results for both data sets. We should also note that for γ, two of the centers are very close to each other, and will be considered just one

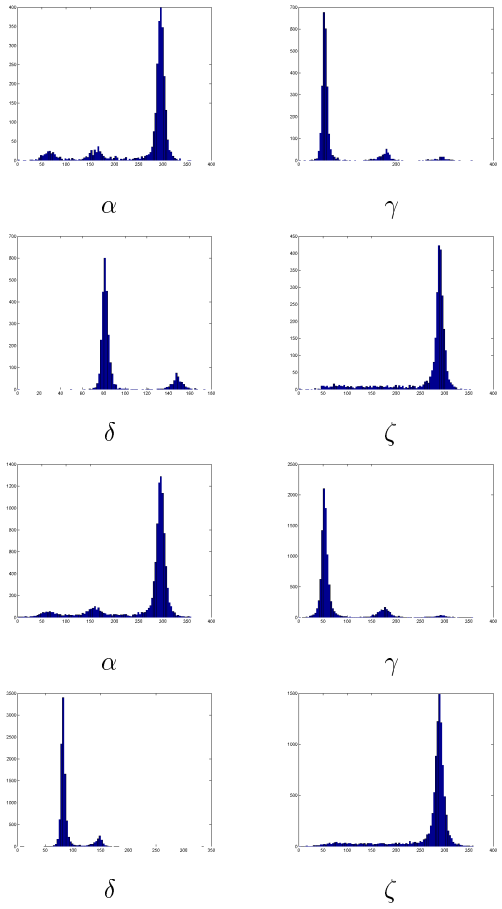


Figure 3: *Cumulative distributions of the torsion angles α, γ, δ, and ζ for the single RNA (first two rows) and the collection of RNAs (last two rows). We observe the similitude among the distributions, marking the presence of "rotamers" not only for a given RNA but also across RNAs. We also observe clear modes, which are automatically detected by the proposed clustering technique. In addition, note that the ζ torsion angle has a large tail not present in the other distributions.*

when we proceed to cluster the data. Note also that although we have pre-defined the number of clusters, this could also be left as part of the automatic process, for example via the expectation minimization (EM) algorithm. We have observed that increasing the number of clusters doesn't produce a significant change in the distortion D, indication that the selected number of clusters is enough. Regarding ζ, if additional clusters are requested, e.g., 3 clusters, for the first dataset these are automatically found at 85.86, 188.25, and 289.27, thereby splitting the large tail (following the directions reported in [9]).

We should also comment on the particular distributions in each cluster. There are a number of reasons for the variability inside each cluster, and therefore it is important to understand the possible statistical explanation for it, since

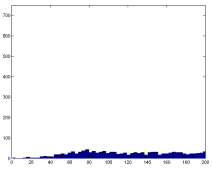


Figure 4: *The tail of ζ for the second dataset. Although two picks can be "guessed," the distribution is much more flat than for example for the α torsion angle.*

this is connected to problems in the data acquisition but also to the RNA dynamics. We have experimented with a number of fitting functions, and we have observed that the best fitting (with a significant improvement) for the major clusters is obtained using exponential distributions, and not Gaussian ones as argued for example in [5]. For example, for the first dataset, the kurtosis for the main cluster is 5.3 for α and 4.6 for ζ, clearly indicating a significant deviation from Gaussian distributions. The log-likelihood while fitting an exponential function improves by 24% with respect to fitting a Gaussian for the α torsion angle and by 23% for the ζ torsion angle. Similar behavior is observed for the other dataset, although sometimes the improvement is a bit more moderate (e.g., for the first mode of α in the first dataset, the improvement is of 16%). Understanding the distributions in each cluster is crucial for future steps of this research, namely probabilistic design.

3.1 Vector Quantization and Binning

The results described above address the scalar quantization of the torsion angles, and will already lead to the fully automatic motif finding technique reported in the next section. We can of course also perform vector quantization, and provide this way an additional automatic way to study the vector clusters, without the need to perform visualization based decisions such as those in [5, 9]. For example, if we request 6 centers for the pair (α, ζ), we obtain $(167.6, 284.6), (291.4, 189.2), (69.1, 284.7), (294.4, 289.4), (105.1, 110.5), (287.4, 86.7)$.[4]

We note that the α component of the automatically detected centers is as in the case of scalar quantization, while the ζ component includes terms that appear both when we request 2 and 3 bins for ζ in the scalar case. Performing this vectorial analysis, for 2 or more torsion angles together, gives us information on the importance of the distribution centers when the angles are considered as a whole. We could then use this as well, instead of the scalar work which we continue below as the basis for vectorial clustering.

[4]These results are for residue-based parsing, while for suite-based parsing we obtain $(167.6, 284.6), (287.5, 86.7), (294.4, 289.4), (105.3, 109.8), (291.4, 189.2), (69.2, 284.2)$. More details in these two types of parsing are provided below.

4 Automatically Finding Motifs

With the above automatic procedure, we can proceed and find motifs. Basically, we cluster the torsion angles according to their proximity to the centers in Table 1. In the results reported below, we have not considered a "dead zone" (equivalent to the manually defined bins "other" in [5], and to some of the results from the filtering approach in [9]), and each torsion angle is classified to one of the clusters. Following the filtering approach in [9] and the "other" bins in [5], we could be more conservative and only consider torsion angles that are at a certain distance of the cluster centers, while considering the rest as "noise." This of course is done also in an automatic fashion, for example requesting the angles to be at p times the variance inside the class. Therefore, the technique here proposed provides not only an automatic clustering approach, but also a way to filter out data if so desired.

Using the notation in Table 1, we present in Table 3 the most frequent cells for the residues in both datasets (left and right for each pair), and for residue and suite parsing (left and right pairs). Similar results were reported in [5] for the first dataset and for residue parsing (that is, corresponding only to the top-left table), where the cluster centers and boundaries were defined manually.

The next step if of course to look for motifs for more than one consecutive residue. In Table 2, we report the larger A-helices we automatically found (these are given by the composition 3111, see [5]) in each residue of the first dataset.

We also found 27 tetraloops (defined by the series 3111, 3111, 2111, 3111), starting at positions 149, 252, 313, 468, 505, 624, 690, 804, 1054, 1197, 1326, 1388, 1468, 1499, 1595, 1628, 1706, 1748, 1793, 1808, 1862, 1991, 2061, 2248, 2411, 2629, 2695; and four e-strands (3111, 3112, 2122, 3222, 3111) starting at locations 172, 210, 1367, 2689.

5 Residue vs. Suite Parsing

RNA can be parsed by residues or by suites as in [9]; see Figure 1. The motivation for the latter is the high correlation between the adjacent phosphate torsional angles ξ and α. This correlation was established for dinucleotides and short oligonucleotides [15]. Here we will extend the relation to any RNA molecule using information theory.

To try to further understand the differences between the two forms of parsing the RNA backbone, we computed

Starting residue	Length
12	12
98	10
294	10
343	13
399	10
418	10
519	13
589	14
606	13
747	12
796	10
1014	14
1139	10
1217	12
1261	16
1291	20
1317	11
1329	11
1453	17
1507	17
1535	24
1606	10
1760	11
1843	12
1896	23
1920	21
2259	12
2429	13
2542	10
2621	10
2708	10

Table 2: *Location and length of larger A-helices automatically found in the first dataset.*

the mutual information between α and ζ, both for residue parsing ($\alpha(i)$ against $\zeta(i)$) and for suite parsing ($\alpha(i)$ against $\zeta(i-1)$). Mutual information is defined as follows [1]: Let x and y be two random variables. First, the *entropy* of x is defined as $H(x) := -E_x[\log(P(x)]$, where $E_x[\cdot]$ stands for the expectation. Entropy measures (in bits) the randomness of a signal, the larger the entropy the more random the variable is. The *joint entropy* is defined as $H(x,y) := -E_x[E_y[\log(P(x,y))]]$, and summarizes the degree of dependence of x on y, while the *conditional entropy* if given by $H(y|x) := -E_x[E_y[\log(P(y|x))]]$, which summarizes the randomness of y given knowledge of x. We can now define the *mutual information*,

$$MI(x,y) := H(y) - H(y|x) = H(x) + H(y) - H(x,y),$$

which is a measure of the reduction of the entropy (randomness) of y given x.

In the case of residual parsing, we obtained $MI(\alpha,\zeta) = 0.83$, while for suites parsing we obtain $MI(\alpha,\zeta) = 1.16$.[5] This increase in mutual information indicates that the suites parsing is more appropriate (as claimed in [9]), at least that these torsion angles are functionally more dependent with this parsing.[6] We should add, for completeness, that $MI(\alpha,\gamma) = 0.82$ ($H(\gamma) = 3.56$), $MI(\alpha,\delta) = 0.46$ ($H(\delta) = 2.74$), and $MI(\gamma,\delta) = 0.38$.

6 Principal Component Analysis of Tetraloops

As done for secondary structures in protein research, e.g., [3], it is important to study the variability of the motifs found in RNA, due once again to its possible implications in the dynamics. Following the work on proteins [3], we perform principal component analysis (PCA) on the 27 tetraloops reported above and in an additional larger data set.

The basic procedure is as follows. Let L denote the number of residues in the motif ($L = 4$ for tetraloops) and N the number of samples (27 for our first example). The first step in the PCA is to compute the covariance matrix C, which is a square matrix of dimension $4L$ (four angles per each residue), whose elements are given by $C_{i,j} = \frac{1}{N-1}\sum_{m=1}^{N}(x_{mi} - < x_i >)(x_{mj} - < x_j >)$, where $< x_i >$,is the i-th coordinate of the mean structure. We then compute the eigenvalues and eigenvectors of this matrix, λ_q and $\vec{v}_q$. The eigenvalues distribution will tell us the number of modes in this class. In Figure 5, top, we clearly see 2 to 3 dominant eigenvalues for this data set, considering the 4 angles $(\alpha,\gamma,\delta,\zeta)$. In the middle, we repeat the computation for a total of 261 tetraloops,[7] considering now all the six torsion angles $(\alpha,\beta,\gamma,\delta,\epsilon,\zeta)$, and defining a tetraloop as the combination $(3?11?1, 3?11?1, 2?11?1, 3?11?1)$, where the symbol ? stands for "don't care" for those angles. We observe again the 2 (maximum 3) dominant eigenvalues (analysis of the eigenvectors will be reported elsewhere). When using the same data set, again with all the six torsion angles, but defining a tetraloop as $(3?11?1, 2?11?1, 3?11?1, 3?11?1)$ we obtain 168 examples. The eigenvalues distribution is shown in the last figure on the bottom, with two dominant eigenvalues once

[5] Both α and ζ have $H = 4.59$.

[6] For computing the MI, we quantized the α and ζ torsion angles in 100 bins. We also tested for different numbers of bins and always the mutual information increased for suite parsing.

[7] rr0011, rr0033, rr0055, rr0043, rr0044, rr0060, rr0061, rr0077, rr0078 and rr0079; HLSU 50 from NDB.

again, even stronger than before.[8] Note that the first and second histograms of Table 5 refer to "tetraloops" in the sense just defined, while the third histogram refers the "tetraloops" in the standard sense [7, 18].

We have used simple (and linear) analysis in this case, while there is no reason to believe that the space of RNA motifs is flat. We plan to investigate the use of tools that consider the geometry of the space of motifs, e.g., [17], where orders of magnitude more data will be needed.

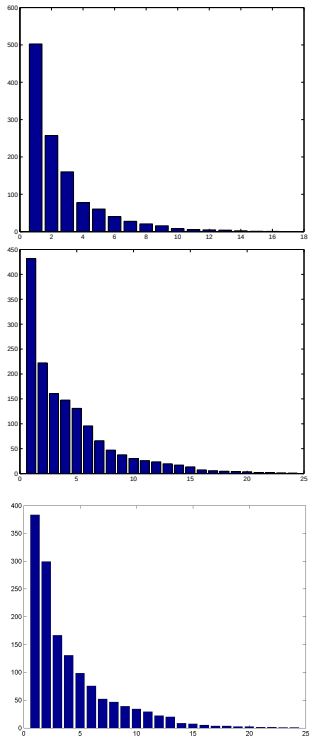


Figure 5: *Frequency plots of eigenvalues corresponding to the tetraloops PCA analysis. The first two plots use tetraloops in the sense defined in this paper while the third in the standard sense.*

7 Concluding Remarks

In this paper we have seen how classical techniques from statistical signal processing are useful for the analysis of RNA structure. These techniques can be augmented with novel clustering approaches being developed by the learning and signal processing community, and investigating those, together with the search for new motifs, is the subject of our current efforts.

[8] The stability of these motifs, and comparison between residue and suite parsing, is the subject of current studies.

References

[1] T. Cover and J. Thomas, *Elements of Information Theory*, Wiley-Interscience, 1991.

[2] *Data Compression*, www.data-compression.com/vq.html

[3] E. Emberly, R. Mukhopadhyay, N. Wingreen, and C. Tang, "Flexibility of alpha-helices: Results of a statistical analysis of database protein structures," *J. Mol. Biol.* **327**, pp. 229, 2003.

[4] A. Gersho and R. M. Gray, *Vector Quantization and Signal Compression*, Kluwer Academic Publishers, January 1992.

[5] E. Hershkovitz, E. Tannenbaum, S. B. Howerton, A. Sheth, A. Tannenbaum, and L. D. Williams, "Automated identification of RNA conformational motifs: Theory and application to the HM LSU 23S rRNA," *Nucleic Acids Res* **1**, pp. 6249-6257, 2003.

[6] A. Hinneburg, M. Fischer, and F. Bahner, "Finding frequent substructures in 3D-protein databases," *Data Base Support for 3D Protein Data Set Analysis – 15th International Conference on Scientific and Statistical Database Management*, pp. 161-170, 2003, Cambridge, MA.

[7] N. B. Leontis and E. Westhof, "Analysis of RNA motifs," *Curr. Opin Struct Biol* **13**, pp. 300-308, 2003.

[8] Y. Linde, A. Buzo, and R. M. Gray, "An algorithm for vector quantizer design," *IEEE Trans. on Comm.*, pp. 702-710, 1980.

[9] L.J. W. Murray, W. B. Arendall, III, D. C. Richardson, and J. S. Richardson, "RNA backbone is rotameric," *PNAS* **100:24**, pp. 13904-13909, 2003.

[10] V. L. Murthy, R. Srinivasan, D. E. Draper, and G. D. Rose, "A complete conformational map for RNA," *J. Mol. Biol.* **291**, pp. 313-327, 1999.

[11] V. L. Murthy, and G. D. Rose, "RNABase: An annotated database of RNA structures," *Nucleic Acids Res.* **31**, pp. 502-504, 2003.

[12] A. Y. Ng, M. Jordan, and Y. Weiss, "On spectral clustering: Analysis and an algorithm," *NIPS* **14**, 2002.

[13] Nuclei Acid Database, http://ndbserver.rutgers.edu.

[14] W. K. Olson, "Configuration statistics of polynucleotide chains. A single virtual bond treatment," *Macromolecules* **8**, pp. 272-275, 1975.

[15] W. Saenger, *Principles of Nucleic Acid Structure*, Springer-Verlag, New York, NY, 1984.

[16] M. Sundaralingam, "Stereochemistry of nucleic acids and their constituents. Allowed and preferred conformations of nucleosides, nucleoside mono-, di-, tri, -tetraphosphates. Nucleic acids and polynucleotides," *Biopolymers* **7**, pp. 821-860, 1969.

[17] J. B. Tenenbaum, V. De Silva, and J. C. Langford, "A global geometric framework for nonlinear dimensionality reduction," *Science* **290**, December 2000.

[18] C. R. Woese, S. Winker, and R. Gutell, "Architecture of ribosomal RNA: constraints on the sequence of 'tetraloops'," *Proc. National Academy of Sciences* **87**, pp. 8467-8471, 1990.

$\alpha\gamma\delta\zeta$	Freq.
3 1 1 1	1812
2 2 1 1	125
3 1 2 2	114
3 1 1 2	111
2 1 1 1	86
3 1 2 1	58
1 1 1 1	47
1 2 1 1	42
2 1 2 2	39
1 1 2 1	38
3 2 1 1	30
1 3 2 2	23
2 1 2 1	21
1 3 1 1	20
1 1 2 2	20
1 1 1 2	19
3 2 2 2	13
3 3 1 1	13
2 2 2 2	12
1 3 2 1	11
3 3 2 1	10
3 2 1 2	10
1 2 2 1	9
2 1 1 2	7
3 2 2 1	6
3 3 2 2	6

$\alpha\gamma\delta\zeta$	Freq.
3 1 1 1	6702
2 2 1 1	593
3 1 1 2	337
3 1 2 2	294
2 1 1 1	294
3 1 2 1	187
1 2 1 1	182
1 1 1 1	161
3 2 1 1	111
1 3 1 1	91
1 1 2 1	77
2 2 1 2	74
2 1 2 2	70
1 1 2 2	70
2 1 2 1	58
2 1 1 2	54
1 1 1 2	53
3 3 1 1	41
3 2 2 2	40
3 2 1 2	40
1 3 2 2	39
2 2 2 2	38
1 2 1 2	37
1 2 2 1	27
1 3 2 1	24
3 3 2 1	23

$\alpha\gamma\delta\zeta$	Freq.
3 1 1 1	1835
3 1 2 1	136
2 2 1 1	125
3 1 1 2	92
2 1 1 1	52
2 1 1 2	42
1 2 1 2	40
3 1 2 2	37
2 1 2 2	36
1 1 2 2	36
1 1 1 1	35
3 2 1 1	31
1 1 1 2	31
2 1 2 1	24
1 1 2 1	22
1 3 2 1	19
1 3 2 2	15
1 3 1 1	14
3 3 1 2	13
2 2 2 1	12
3 3 2 1	12
3 2 2 2	11
1 2 2 2	10
3 2 1 2	9
3 2 2 1	8
2 2 1 2	8
1 2 1 1	7
1 3 1 2	7

$\alpha\gamma\delta\zeta$	Freq.
3 1 1 1	6946
2 2 1 1	630
3 1 2 1	375
3 1 1 2	298
2 1 1 1	206
2 1 1 2	148
1 2 1 2	144
3 2 1 1	123
1 1 1 2	120
3 1 2 2	119
1 1 1 1	104
1 1 2 2	91
1 3 1 1	84
1 2 1 1	76
2 1 2 1	71
2 2 1 2	68
2 1 2 2	64
1 1 2 1	58
2 2 2 1	43
1 3 2 1	38
3 2 2 1	34
3 3 1 2	34
3 2 1 2	32
3 2 2 2	28
1 3 2 2	27
1 2 2 2	26
3 3 2 1	26
1 3 1 2	23

Table 3: *Frequency of most popular torsion angles motifs, both for residue parsing (first two columns) and suite parsing (last two columns). The table on the left of each pair corresponds to the first dataset while the one on the right corresponds to the second dataset. Note that angles of the first two columns correspond to the same residue, while the last two columns to suites; see Figure 1.*