MASARYK UNIVERSITY
FACULTY OF INFORMATICS

Computational analysis of sequence families in DNA quadruplexes

DIPLOMA THESIS

Marek Vavruša

Brno, Spring 2014

Declaration

Hereby I declare, that this paper is my original authorial work, which I have worked out by my own. All sources, references and literature used or excerpted during elaboration of this work are properly cited and listed in complete reference to the due source.

Advisor: Ing. Matej Lexa, Ph.D.

Acknowledgement

I'd like to thank my advisor, Ing. Matej Lexa, Ph.D., not only for the guidance when the results were hard to come by, but also for accepting me as his student and introducing me to the wonderful field of Bioinformatics, along the challenges I could really enjoy. I also want to thank my lab mates, Karel Slaný and Jan Včelák, for many fruitful discussions, and support in form of the coffee. Finally, to all people who positively influenced my life.

Abstract

The thesis presents an automated approach on the G-quadruplex structure classification with respect to the structure topology, and a novel method for heuristic prediction of a G-quadruplex topology from the nucleotide sequence. As a result, tools for both classification and prediction are developed and evaluated.

Keywords

DNA sequence analysis, DNA conformation, G-DNA, quadruplex, tetraplex, machine learning, classification, pattern recognition

Contents

Chapter 1

Introduction

Aside from the Watson-Crick double-stranded helix conformation, the DNA is known to assume various secondary structural motifs including quadruplexes. Guanine-quadruplex (G-quadruplex, GQ) is a secondary DNA/RNA structural motif formed by certain guanine-rich sequences, where guanines self-associate by Hoogsten hydrogen bonding into planar arrays of four guanine molecules (tetrads). These tetrads are stabilized by monovalent alkali cation, usually $K+$ or $Na+$, located in the structure center of gravity, or in between the tetrad planes.

G-quadruplex forming sequences have been identified as repetitions in the eukaryotic telomeres, and later in non-telomeric regions as well. Research [Eddy and Maizels, 2008] suggests the genome-wise distribution is not completely random, but located preferentially in the repetitive genomic sequences, such as telomeres, gene promoter regions, recombination sites and tandem repeats.

The evidence supports a hypothesis that the G-quadruplex structures play an important role in the transcription regulation as suggested by [Huppert and Balasubramanian, 2007] and others. Of particular interest, G-quadruplex-forming sequences have been found with high frequency in the proto-oncogene promoter regions, near the transcription start site. This discovery sparked an interest as a a potential target for an anti-cancer drug development. However, the exact method of function is still a matter of debate and is still actively researched.

Given the functional importance, a lot of effort has been invested into the structure resolution using the Nuclear magnetic resonance (NMR) and crystallography. This has however proven to be difficult since the guanine-rich sequences in high concentration tend to form a variety of structural conformations, and many of the conformations were evaluated using molecular dynamics modeling (MD).

The molecular dynamics simulation is however a potentially very resource-intensive operation, which limits its use for a genome-wide evaluation, and calls for additional filtering of the data. So far, there have been several approaches in Bioinformatics on this topic - the sequence motif representation, heuristic scoring and the prediction based on the energy modeling to name a few. The goal of the thesis is to present a novel method of both G-quadruplex structure classification, and a probabilistic topology prediction from the nucleotide sequence.

Chapter 2

G-Quadruplex

The goal of this chapter is not a complete coverage of the quadruplex conformation, but rather an introduction to the concepts and properties used in chapters 3 and 5. Refer to the [Burge et al., 2006] for supplementary information.

2.1 Structure

G-quadruplex is a type of DNA secondary structure, where the sequences of guanine[1] self-associate in tetrads, planar formations of four nucleotides held together by Hoogsteen hydrogen bonds, much like Watson-Cricks base pairs in duplex DNA ([Burge et al., 2006]). However, at least two adjacent tetrads stacked one on another are required to form a quadruplex structure stabilized with an alkali ion, often K^+ or Na^+, abundant in physiological conditions. These ions are often at, or near the structure center of gravity, not only bonding the tetrads together, but also influencing the quadruplex conformation by the strength of the bond.

2.1.1 Guanine tetrads

From the spatial point of view, tetrad closely resembles a pair of tetragons (Figure 2.1). Outer tetragon is formed by the $N9$ atoms where the guanine connects to the sugar-phosphate backbone, and encloses the inner tetragon formed by $O6$ atoms around the alkali

[1] Or substituted compound like Inosine (e.g. PDB structures `2KOW` or `2KKA`) (DG, G) or modified guanine (8-bromoguanosines in `2E4I`). Usage of modified residues in molecular dynamics simulation (MD) is discussed in [Cang, 2010, p.63, G-quadruplexes with modified residues].

ion near the plane center of gravity. Geometric properties of these two tetragons play an important part when describing GQ conformation, namely planarity and twist.

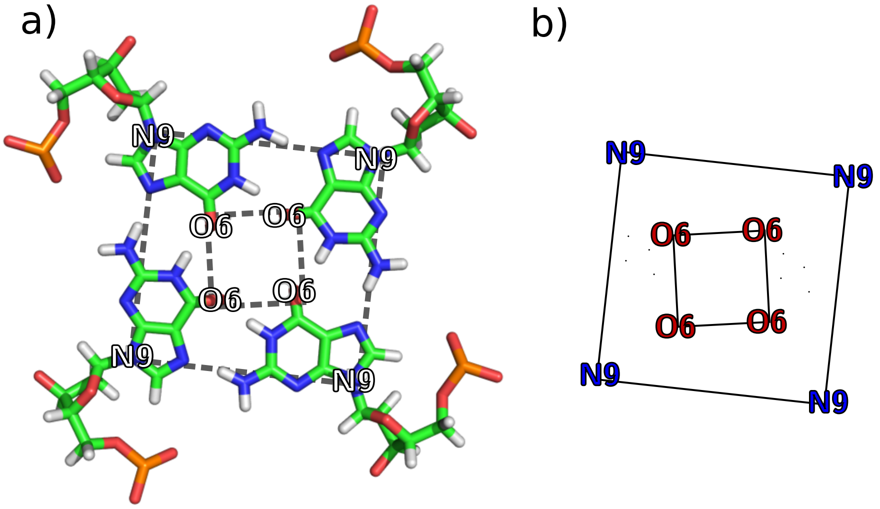

Figure 2.1: Tetrad geometry - inner and outer tetragons.

In order to describe GQ geometry, two known quantitative parameters can be used. First is a twist angle between the two neighbouring tetrads, second a tetrad planarity. The distribution pattern for these two parameters in specific GQ conformation groups was studied by [Reshetnikov et al., 2010], and a correlation between the parameters and the tension in the GQ structure was identified. In another words, for each GQ structural conformation, a specific distribution of the twist angles and planarity can be found.

Twist angle is not a parameter original to the G-Quadruplexes, as it was hitherto used to represent a degree of tension of the duplex DNA [El Hassan and Calladine, 1995] and other oligonucleotide structures in the past. On the presumption that a GQ is stabilized by similar factors as the DNA[2], the applicability for GQ tension description was successfully evaluated[3] in [Reshetnikov et al., 2010]. From the measurements of the available GQ structures, the twist angle range is quite wide, from as low as 15° to 36° , and even larger span is probable. Both the nominal value and the distribution is strongly affected by the number of strands and the types of loops connecting the tetrad planes. For example, simple four-stranded structures without loops have an ideal twist angle and wide distribution because of the lack of structural restrictions, while the added loops tighten the distribution range.

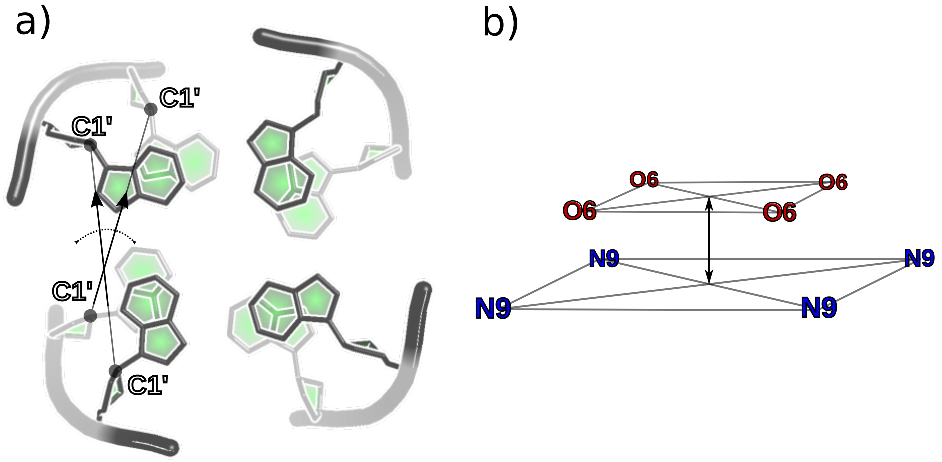

Figure 2.2: Twist angle between the adjacent tetrads and planarity.

[2] Mainly base stacking interactions, Hoogsteen-hydrogen bonds, electrostatic interactions and the hydration of the sugar-phosphate backbone. GQs however have a significant and unique stabilizing factor; the coordination of the tetrad core $O6$ atoms with the alkali ion in the GQ core. The ion properties play a significant role in the stability and the conformation of the GQ, as shown later in the chapter 5.

[3] Details on calculating the twist angle are discussed in chapter 3.

Planarity describes a RMSD[4] (in Å) between the centers of gravity of the inner and outer tetragons (Figure 2.1) of an each individual tetrad. As [Reshetnikov et al., 2010] describes: *"If the quartet is symmetrical and all guanines form hydrogen bonds with each other, this parameter fixes the planarity of the quartet. If the hydrogen bonds break, the symmetry of the quartet is disturbed; the parameter fixes the degree of quartet distortion, which includes deviations in both symmetry and planarity of a quartet."*

As a summary, while the twist angle mainly represents the strands and overall tension of the structure, and the planarity reflects the amount of stacking interactions in the loops between the tetrads. When the tetrad is symmetrical (therefore all guanines form hydrogen bonds, see Figure 2.3), the tetrad plane is fixed. However, should some hydrogen bond break, this tetrad symmetry is violated and introduces a certain degree of planarity distortion.

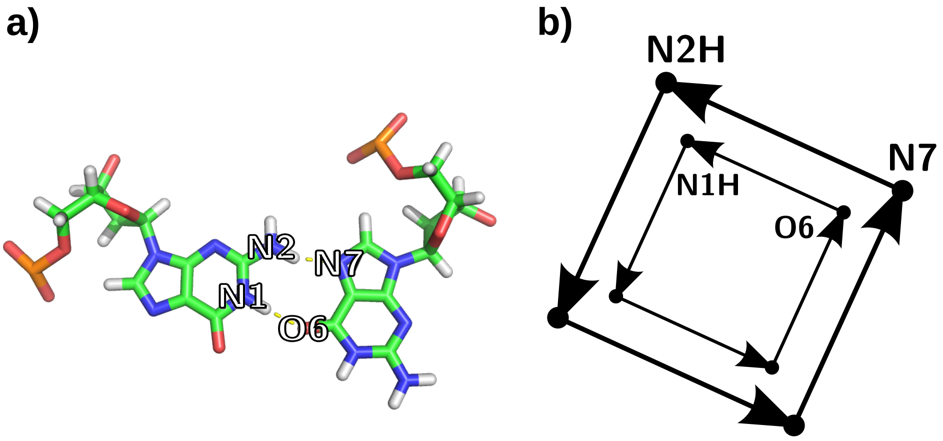

Figure 2.3: Bonds in associated guanines.

[4] Root mean square distance.

Composition There are several important interactions that are specific for the tetrad arrangement. The two neighbouring guanines (DG, DNA guanine) are associated with Hoogsteen-hydrogen bonds including two important interactions [Deng et al., 2001] that define an associated pair of DGs:

- The $O6$ atom is bonded to the $N1H$ of neighbouring guanine with an average distance of 2.85Å(Figure 2.3, part *a)*)
- The neighbouring guanine atom $N2H$ is bonded to the $N7$ of the original guanine with an average distance of 2.80Å

Guanine pairing Using the definition above, a tetrad is defined as a chain of four associated DGs, where the last DG is associated with the first, thus forming a closed circle. Notice the bonds are directional, therefore DGs in tetrads can be treated as ordinal data, if the first DG is arbitrarily chosen[5].

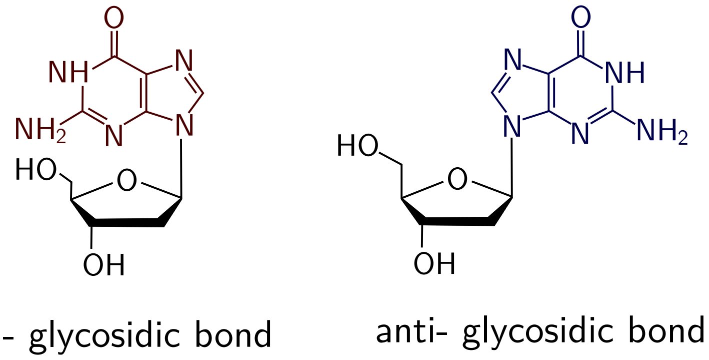

Figure 2.4: syn- and anti- glycosidic conformations.

The mutual orientation of the DGs in the tetrad depends on the the orientation of the individual strands, and can be in either *anti-* or *syn-* conformation (Figure 2.4) with respect to the glycosidic bond. That means for example, that if we have a four stranded GQ with all strands orientated in parallel, all glycosidic angles will be in *anti-*

[5] For example a DG that is attached to the first strand in the structure.

conformation (Figure 2.7). On the other hand, if at least two strands are in an anti-parallel orientation, there will be guanines with both *anti-* and *syn-* glycosidic angles present.

Geometry of paired guanines Unlike the duplex DNA, GQ structures have always four grooves[6], bounded by the connecting loops. Groove width is highly variable and depends on the topology and the types of connecting loops, and may be either symmetric or asymmetric between the opposite DG pairs. One of the most important parameter dictating the groove width is the glycosidic angle conformation, that either packs the sugar-phosphate backbones together or pulls them apart.

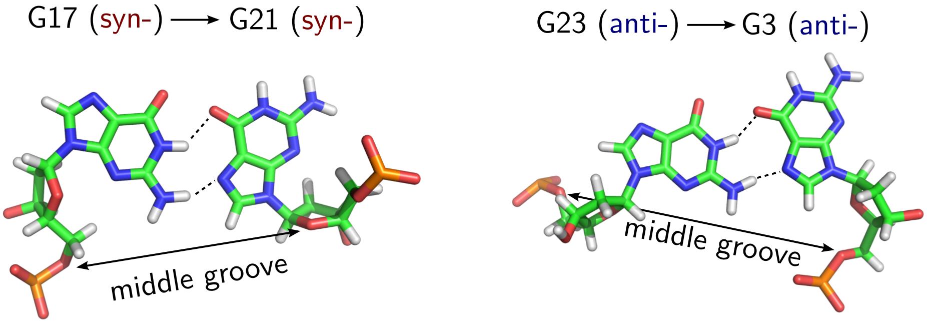

Figure 2.5: Example of *anti-/anti-* and *syn-/syn-* DG conformation (PDB 2F8U).

[6] Cavities in the sugar-phosphate backbone.

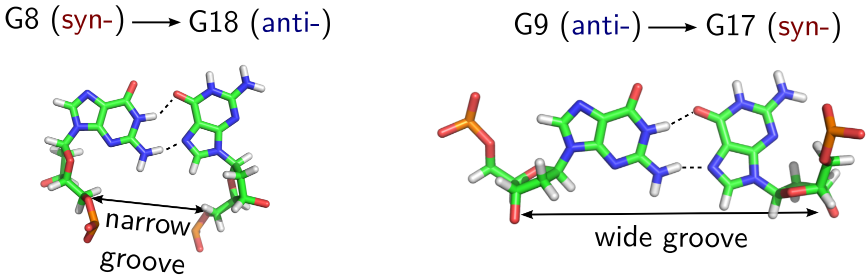

Figure 2.6: Example of *anti-/syn-* and *syn-/anti-* DG conformation (PDB 2F8U).

The grooves are highly symmetric when the two adjacent DGs glycosidic angles are of an equal conformation (Figure 2.5). However, unequal glycosidic angle conformation have a significant effect on the groove geometry, and stresses the overall structure (Figure 2.6) as is shown in chapter 3.

Irregularities While the DG is the natural building block of a tetrad, it may be substituted with a compound of similar properties. For example inosine can substitute guanine, as it has similar properties with respect to the pairing rules, and has likely no significant effect on the resultant conformation[7]. Moreover, GQs are often artificially prepared with modified compounds, like in PDB 2E4I ([Matsugami et al., 2007]), where the specific DGs are substituted with a modified compound[8] in order to stabilize the structure in given environment and reduce the possible conformational heterogeneity.

[7] Found in various conformations, e.g. PDB 2KOW, 2A5P, 2KZD.

[8] 8-bromoguanosine in PDB 2E4I.

2.2 Sequence

G-quadruplexes can fold from either one, two or four strands of DNA, and are called unimolecular[9], bimolecular and tetramolecular respectively. The tetrad folding rules mandate at least four DGs to create a single tetrad. This implies the need for either four consecutive DG sequences (G-tracts) on a single DNA strand, or split between the strands in case of di/tetrameric conformation with a minimal DG run length of 2 (as at least 2 tetrads are required to form a GQ). Moreover, all but the tetrameric conformations require extrahelical loops to connect the strands together. The composition of these loops varies in both length and nucleotides sequences significantly. While there is a clear preference of some loop sequences to fold into specific conformations, the sequence alone is only a single variable and that alone is not sufficient to dictate a specific folding pattern. This hints at a high conformational ambiguity of the GQ structures, as is confirmed in the scanned structures to date. Interestingly, the sequence composition is often species-specific. For example the repeated $TTAGGG$ sequence discovered in the mammal telomeric DNA.

Monomeric GQ

The intramolecular GQ sequence is the most specific, as all four DG occurrences have to be on the same DNA strand. The nucleotide sequence can be described as:

$$G_m \underbrace{X_n}_{\text{Loop}} G_m X_n G_m X_n G_m \tag{2.1}$$

where the X represents any combination of nucleotides, including G, $m \geq 2$. Several approaches on unimolecular GQ sequence description ([Ryvkin et al., 2010]) suggest an $m = 3$ and n in $\langle 1, 7 \rangle$ for improved specificity. While this may hold for the prediction of the most stable GQs, more recent studies show successfully folded GQs with several types of sequence deviations, that are not described by the formula (2.1):

[9] Alternatively called "intramolecular" or "monomeric".

1. Loop sequences longer than 7 nucleotides - depending on the physiological conditions, even loops with 15 nucleotides are stable [Guédin et al., 2010].

2. GQ with zero-length loops can exist, as shown in [10] [Marušič et al., 2013] with intra-tetrad strand reversal zero-length loops. While this may be accepted by the above formula in some cases, parts of DG-tracts would be wrongly assumed as loops.

3. Finally GQs with bulges in the consecutive DG tracts, where some G-tracts are interspersed with various nucleotides, have been confirmed to occur. These structures defy the sequence mandating consecutive runs of DGs altogether, as shown in [Mukundan and Phan, 2013].

The formula (2.1) still allows DG runs of an unequal length. The m would be chosen as the minimum of the G-tract lengths, and the extra DGs would simply be part of the X_n loop regions. This is especially true in the non-telomeric DNA regions and lower eukaryotics [Burge et al., 2006].

Dimeric GQ

Bimolecular GQ sequences are less specific, with only two G-tracts and allowing the side X_n sequences to be zero-length.

$$X_n G_m X_o G_m X_n \tag{2.2}$$

The central X_o sequence represents the single extrahelical loop between the G-tracts, and the same rules as for the loops in the unimolecular GQs apply. The resultant GQ is folded from the two strands conforming the above formula (2.2). The sequences on each strand are likely to be equal, although in principle GQ can fold from non-equal sequences as well.

[10] 2M53

Tetrameric GQ

$$X_n G_m X_o \tag{2.3}$$

Tetrameric GQ sequences are loopless, since each strand attaches to the guanines at the same position in the tetrad, thus not requiring connecting loops. The typical conformation for four stranded GQs is when all four strands are in parallel (Figure 2.7, first structure), which makes it the simplest GQ conformation.

2.3 Loops

Monomeric and dimeric GQs require three or two loops respectively, since the number of strands is lower than four; the number of stems needed to form a stack of tetrads. The loops consist of a mixed sequences of nucleotides interleaving the sequences of G-tracts, thus not usually involved in the composition of the tetrads as shown in Figure 2.1. Loop conformation is highly varied and is determined by the loop length, stacking interaction in the nucleotides, polarity of the connected strands and overall structural restrictions of the GQ. There are three basic loop types existing in the GQ - **lateral**, **diagonal** and a **propeller** type [Burge et al., 2006]. The composition of these loop types together with strand polarity is often used to classify GQ into families, as shown in chapter 3.

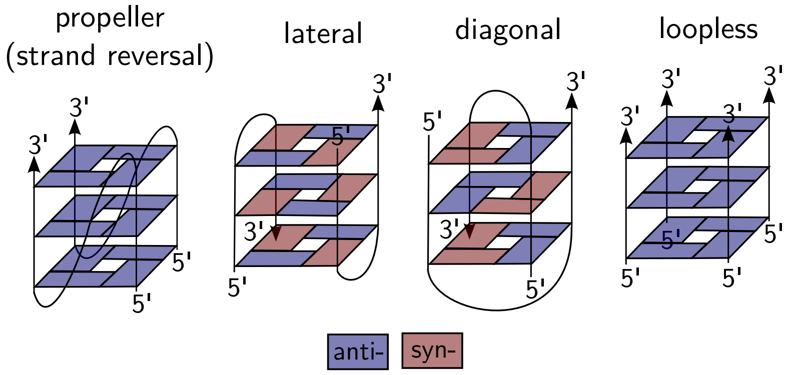

Figure 2.7: Example strand orientations and loop types.

Lateral loop Lateral (or edge-wise) loops connect the two adjacent DGs on the same tetrad plane (Figure 2.7). Depending on the connected strands polarities, these loops can be either head-to-head or head-to-tail configuration. The *head* or *tail* refers to the strand head or tail the loop connects in dimeric GQs. Two lateral loops can also be present either on the same or opposite faces of the GQ, if we define GQ face analogous to a face on the cuboid.

Diagonal loop Like the lateral loop type, diagonal loops connect two DGs on the same tetrad plane, but the connected DGs are opposite to each other. In this instance, the orientation of all adjacent strands must alternate between parallel and anti-parallel, because the loop type requires the connected strands to have the opposite orientation (similarly to lateral loops).

Propeller (strand-reversal) loop Propeller type loops[11] are the only type of loops that cross between the tetrad planes. Moreover, propeller type loop is the only one formed by connecting two adjacent strands of the same orientation (in parallel) by an external loop from the top to the bottom tetrad and vice versa.

Loop combinations in GQs While in theory monomeric GQs (with 3 loops) can have as much as 27 possible combinations, research [Webba da Silva, 2007] indicates that some of potential loops are sterically unfavourable. It also suggests 13 possible combinations that are likely to be stable, although not all topologies have not been observed so far, so this remains to be verified. Based on that hypothesis, rather than a novel loop combination, structures with known loop combinations stacked on top of each other using the interconnecting loop are more likely to be discovered[12]. Possible loop topologies are discussed in the next chapter 3.

[11] Also sometimes referred to as strand-reversal (or chain-reversal), although "propeller" is preferred to not introduce any confusion with the strand polarity.

[12] These structures are referred to as G-wires

Chapter 3

Structure classification

3.1 Spatial properties

There are two fundamental structural units in the GQ, which properties can be used for classification. The tetrad structure, and the geometry of the tetrad stacking described in section 2.1.1. The four guanines in the tetrad structure can be arranged in a finite number of configurations. Given that there are four guanine pairs in the tetrad, and each guanine pair have four possible conformations, there are there are 4 combination options for the first pair, 2 for the second and third pair, and only 1 for the last pair. This makes 16 possible combinations per tetrad plane, as shown in the [Webba da Silva, 2007, Figure 2].

Depending on the number of stacked tetrads, this would make 16^n GQ configurations (or topologies). Fortunately, not all topologies are possible given the glycosidic angle conformation must be the same for adjacent parallel strands and opposite for anti-parallel strands, so the number of topologies is better described by the possible loop configurations.

GQ type	Theoretical configurations	Probable configurations
Monomeric	3^3	13
Dimeric	3^2	9
Tetrameric	2	2

Table 3.1: Number of possible topologies in GQs per strand count [Webba da Silva, 2007] and [Karsisiotis et al., 2013].

Relationship between the loops and the tetrad configuration

There are three possible loop configurations - the lateral, diagonal and propeller type. The connected strands have to be in the antiparallel orientation for the lateral and diagonal loop type, and in the parallel orientation for the propeller loop type. The loop can also be formed only between the guanines in *anti→syn* (or *syn→anti*) conformations. This is obvious for the lateral and diagonal loops, but the propeller loop interconnects different tetrad planes, which further restricts the possible topologies in which the loop can be formed (i.e. strand polarity is not the minimal requirement).

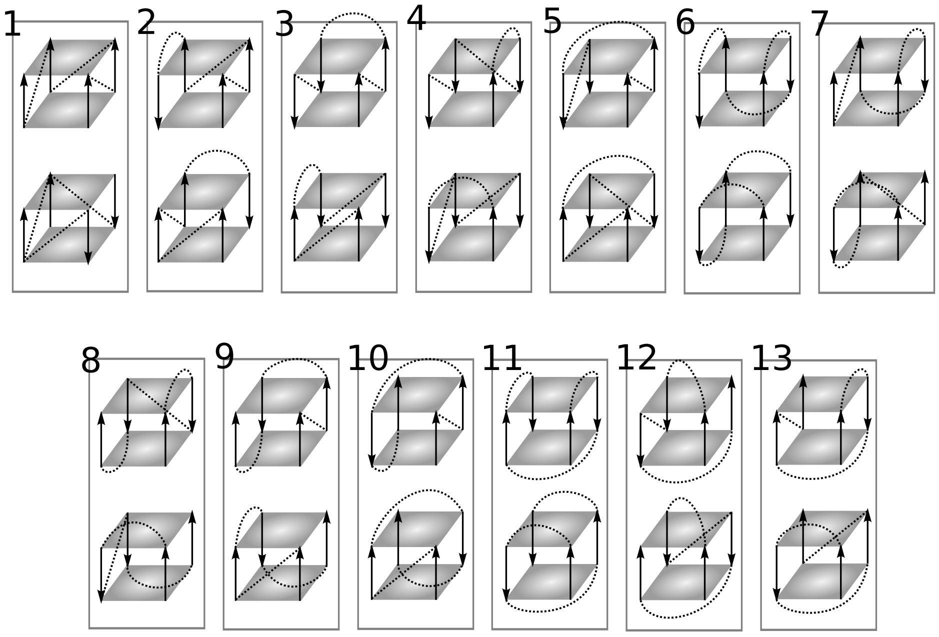

Figure 3.1: Possible 26 topologies for monomeric GQs.

Not all loop configurations are also sterically possible, for example monomeric GQ containing two consecutive diagonal type loops is impossible, and even two diagonal type loops on the same tetrad plane (connected with a lateral loop) are highly improbable. The recent research [Karsisiotis et al., 2013] further divides the 16 possible configurations into groups according to the 8 possible groove-width combinations (paragraph 2.1.1), which is very practical as the groove-width combination can be derived from the loop sequence and establishes not only single-tier GQ families, but a hierarchy. The groove-width combination is defined as a sequence of grooves in the GQ from the first $5'$ strand point of view in an anti-clockwise directions, where the narrow, medium or wide groove is represented by the *n, m, w* letters respectively. For example a groove width combination *"mwmn"* describes a GQ where the first groove has medium width, the second is wide, the third has again a medium width and the last groove is narrow (e.g. model *3/a* in Figure 3.1).

Systematization of the naming

The hierarchy subdivision is also useful in systematization of the naming. Several loop topologies have been given a variety of names, e.g. well represented *PPP/1* loop combination is simply referred to as *"propeller"*. This arbitrary naming came out of necessity because of structural versatility, it is not however practical to invent names for every of the 26 theoretical (13 possible) topologies.

Loop configuration	Name
PPP/1	Propeller-type
LDL/11	Basket-type
LLL/6	Chair-type
LLP/7	3+1
LPL/8	2+2

Table 3.2: Table of conveniently named loop configurations, the number in the loop configuration refers to the group in Figure 3.1.

Relationship between the topology and the GQ geometry

Added extrahelical loops put an additional tension on the overall structure, and are thus capable of altering the GQ geometry. Lateral loops are often favoured by short loop sequences, which, aside from tensioning the backbone, often form stacking interactions with the DGs in the tetrads. This results in both abnormal twist angle[1]Either lower or higher. and higher planarity. An addition of either propeller type loop or a diagonal loop introduces significant changes in the structure twist angles. For example, a longer diagonal loop in between the two lateral loops compensates the tension from the short lateral loops with the elasticity and less stacking interactions in the longer loop, which on the other hand unfavourably affects the planarity. Last loop type, the propeller has even more profound effect on the structure stability than lateral loops, with the twist angle geometry close to the loopless structures.

So among other contributing factors, the chosen loop configuration is known to affect the overall GQ geometry and vice versa, so the classification can be twofold - to classify a GQ structure by describing a topology or by it's geometric properties.

3.2 Classification using the GQ geometry

Until [Reshetnikov et al., 2010] proposed an algorithm for the general analysis of the GQ polymorphism in the PDB[2], no systematization attempt on larger scale structure analysis was published. The article investigates a correlation between the GQ group, and the structure twist angle and planarity. While the group is defined by the GQ topology, it can also contain sequences of related origin. About[3] 74 structures were analyzed and divided into groups. The correlation between the distribution of the evaluated parameters and the GQ topology features was then studied. The results suggest an influence of the three possible loop types on the overall geometry.

3.2.1 Planarity deviation

In order to measure planarity (or out-of-plane) deviation, guanine tetrads must be identified in the structure. The proposed definition of the tetrad is as follows - let a guanine have a neighbour in contact with the $O6$ atom of the initial guanine via the $N7$ atom. A combination of four such guanines form a tetrad if the fourth (last) guanine

[2] Protein Data Bank (RCSB PDB).

[3] Not all structures were classified, since the topology of some structures was too unique to form representative samples.

interacts with the first one and the maximal permissible offset of the atoms from the atom from the surface is lesser or equal to 2Å. The planarity for such tetrads is calculated as in the paragraph 2.1.1. It is however unclear whether maximum, mean value or variance between all the tetrads should be used, as both the mean value and the inter-tetrad variance parameters were used in [Reshetnikov et al., 2010].

3.2.2 Twist angle variation

Measuring inter-tetrad twist angle introduces a new definition of the *next tetrad*. The next tetrad is defined as follows:[4]

- Tetrad, which is nearest to the initial one.
- Each DG in this tetrad has a unique nearest DG in the initial tetrad, and the maximum RMSD between the $C1'$ atoms of these DGs is lesser or equal to 10Å.

Adjacent tetrad pair is then defined as an arbitrary tetrad and it's next tetrad, should the next tetrad exist.

Twist angle is then calculated as an angle between the two vectors. The first vector joined the $C1'$ atoms of the adjacent DGs in a tetrad and the second vector joined the $C1'$ atoms of the corresponding DGs in the next tetrad (Figure 2.2, *a)*). This is also referred to as a dihedral (or torsion) angle between the four $C1'$ atoms, which is widely used to describe the angle between two planes that can be seen by looking at the planes along their line of intersection. In this particular case, the planes are defined by the $C1'$ atoms of the two paired DGs (Figure 2.2).

Similarly to the planarity deviation, both average twist angle and a variance between the adjacent tetrad pairs is used for classification.

3.2.3 Method results

Since there is no publicly available implementation, the planarity and the twist angle measurements were implemented as part of the

[4] This is disputed in subsection 3.3.1, as some of the newer structures defy this definition because of the irregularities in sequence or in the structure.

classifier implementation to confirm [5] the data used in the [Reshetnikov et al., 2010]. The dataset was extended with several newly published structures, only monomeric GQs were used for the prediction part. Three topologies were removed from comparison, namely *2+2, PDL, PPLP* for an insufficient amount of samples. The experimental results confirm the distribution profiles of the measured parameters may be possibly used as parameters for classification, with each group expressing a clear pattern in the profile:

Name	#	Twist angle	Planarity	Motif
Propeller	10	$31° \pm 3°$	High	Pronounced maximum.
Basket	6	$18° \pm 4°, 36° \pm 4°$	Low	Two pronounced shoulders, decreased planarity.
Chair	9	$15° \pm 5°$	High	Less pronounced maximum, high planarity variance.
3+1	9	$12° - 24°, 28° \pm 4°$	High	Additional maximum, upshifted angle distribution from chair-type.

3.2.4 Method difficulties

Although [Reshetnikov et al., 2010] is unclear on this topic, the described method could be used for an automated structure classification. There are however several difficulties, which render it impractical for fully-automated use.

[5] The differences probably exist in the method of calculation which is not elaborated in the article. For example the implementation (section 6.1.4) interprets the twist angle as an average of angles in all four sides of the tetrad pair versus measuring a single angle on an arbitrarily chosen side or treating each measurement as unique. This approach has a smoothing effect on the twist angle results.

- High intra-class variability of the samples. While the group twist angle distribution motifs, like pronounced shoulders, can be found, they are insufficient in describing majority of the structures in the group. For example the PDB `143D` can have a mean twist angle as low as 10.30°, while the `230D` from the same basket-type group as high as the 25.42°. The measurements of the models within a single PDB structure vary significantly, it is therefore difficult to identify a GQ within a group based on the twist angle and parameters only.

- Low inter-class variability. The broad range of values in each group introduces significant overlap between the groups. This insufficient separation introduces errors in automated classification, as shown in the evaluation (section 3.4).

- The original study worked with 34 different monomeric structures with a similar number of representatives per each group. This makes a very small set of data to infer knowledge about the whole groups. Since then, the number of studied structures multiplied, especially the structures with widely represented topologies like the propeller-type, that now introduce a problem of overfitting for these popular topologies and underfitting for unique topologies with only a few samples.

3.2.5 Useful implications

Despite the method drawbacks, some useful knowledge about the relationship between loops and GQ geometry can be inferred. For example lateral loops have the most profound effect on the overall geometry, represented by lowering the twist angle. Lateral loop influence could be mitigated by the introduction of either a diagonal or a propeller-type loop, where the diagonal loop relaxes the tension through it's typically longer loop length, which results in shifting up the twist angle distribution. The propeller-type loops are only a minor source of tension in the structure and are often favoured by the short, single nucleotide sequences. This information could be used to further refine the topology prediction.

3.3 Classification using path tracing

While most of the researched structures are published in the RCSB PDB, there is no extensive compiled list of all the structures that have their structure scanned. The common approach is to search PDB for keywords, as for example *tetrad, tetraplex, quadruplex* or the tetrad structural motif, either approach however yields not only relevant tetrads, but also ligands or related structures. As to this date, there is no fully automated method for such database-wide pulldown, GQ structures recognition, and classification. Previously published studies either analyzed and separated GQs into groups by hand, or by a semi-automatic search for keywords in the publications related to the structure. This approach was sufficient when there was only a handful of GQ structures scanned, but it ultimately falls short with the database growth.

The path tracing method is fundamentally different from the classification of geometric properties. While the previous method worked with measurements of the overall structure, the path tracing method is a deterministic algorithm which identifies atoms and residues that form G-tracts and loops. By observing where the transitions between G-tract and loop happens, it is able to determine overall loop topology.

3.3.1 Tetrad discovery

Tetrad discovery is based on the definition in previously described method (subsection 3.2.1). There are however arbitrarily imposed limits on the bond distances, that are required to filter out partially-complete tetrads in the loops.

Chain filtering In order to assemble tetrads, first a list of all DG residues is compiled by walking through the atoms in all chains. Some substitutes like inosine (*DI*), 8-bromoguanosine (*BGM*) or modified residues like *LCG* are accepted into the list, as they are present in the various structures for reasons described in paragraph 2.1.1. These irregular substitutes will be treated as guanines in the algorithm description for the sake of clarity. First a potential tetrad-forming DG is picked from the list head, and the list remainder is iterated three times in order to find the three subsequent *next tetrads*, where the *next tetrad* is represented by the DG, whose $N2$ atom is closest to the $N7$ atom of the current tetrad.

If the algorithm recognizes the fourth closest DG, a tetrad is formed under condition that the last DG bonds to the first (tetrad-forming candidate) DG. However, if the condition is violated or the four DGs cannot be identified, the potential tetrad is reinserted at the end of the list. The index of the first failed tetrad-forming DG candidate is marked in order to prevent revisiting failed tetrad-forming residues. Such residues theoretically may still be part of an overlapping tetrad (although unlikely), so the first DG is reinserted **after** the marked index, and the rest **before** the index. In another words, only the guanines before the index mark represent potential tetrad-forming candidates.

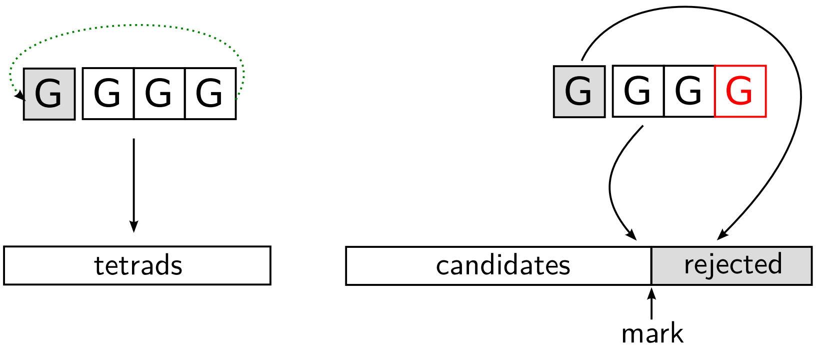

Figure 3.2: Guanine tetrad is either complete and accepted or disbanded and partially reinserted into the list.

The output is a list of assembled tetrads. If at least 2 tetrads are recognized, the structure is treated as a potential GQ, otherwise rejected. The time complexity of the assembly is $O(n^2)$ and can be further improved by presorting the residues relatively to the first (or building a spatial tree) residue, which would allow a cut off for residues too far away. In practice however, this is rarely needed as the number of guanines in the structure is in the order of 10^1.

The mutual orientation of the tetrads and the stacking order is lost in the process, but it can be reconstructed as needed.

Tetrad orientation In order to identify loop connecting points, both the stacking order and the orientation of the DGs in the tetrad must be reconstructed. There are two intuitive ways to achieve this - either following an arbitrary chain and stacking the tetrads with respect to the order of the DGs on the chain, or by looking up a DG, that is closest to the first DG in the tetrad. Given the directionality of the DG bonds in the tetrad, a single closest pair suffices to recognize both stacking order and mutual orientation.

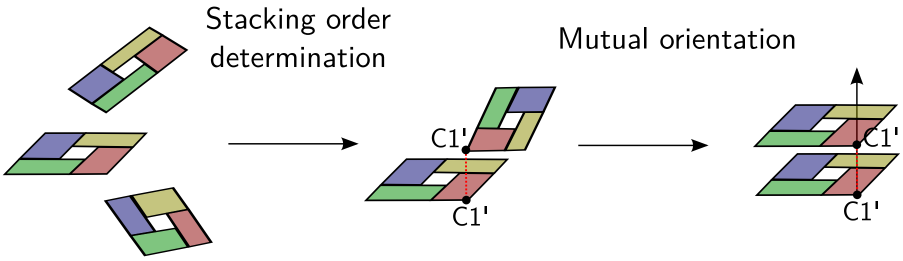

Figure 3.3: Determination of tetrad stacking order and mutual orientation.

The algorithm3.3 first separates the list of tetrads into sorted (S) and unsorted tetrads (U). In the beginning, an arbitrary tetrad is put into S and serves as a starting point. Then a closest DG for the last tetrad in the S is looked up. If it is discovered, the tetrad containing such guanine is moved from the U to the S. This tetrad is also rotated until the identified guanine is first to keep the mutual orientation. In this case, because the guanines between the planes are not hydrogen-bonded, the closeness function is defined as the RMSD between the $C1'$ atoms of the guanines with a maximum distance size of 15Å.

It is possible for some GQs to contain multiple connected tetrad stacks (multimerized quadruplexes or G-wires). This method may or may not recognize the different stacks based on the connecting loop length. Should the need for multimerized tetrad stack analysis arise, this algorithm could be easily extended, so that it is called iteratively to identify separate tetrads stacks. The connecting loops would be identified by chain threading, as described in the following section.

3.3.2 Chain threading

The output of the chain filtering phase is an ordered list of mutually oriented tetrads, therefore the non-tetrad forming DGs are filtered out. This is useful in the precise loop sequence identification, as the *loop sequence set* can be defined as a complement of the nucleotide residues on the chains to the DGs in assembled tetrads. Moreover, the residues in the chains are ordered with respect to the $5' \rightarrow 3'$ direction, which allows the algorithm to determine the order of the loops as well.

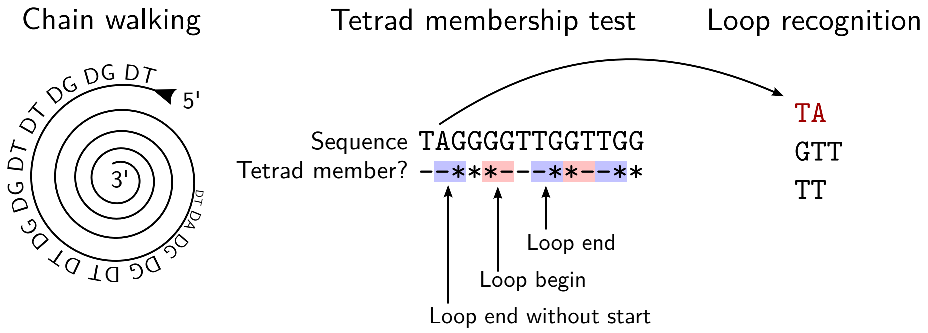

Figure 3.4: Tetrad (or G-tract) membership test and loop recognition.

The algorithm requires the list of assembled tetrads and and the chains from the PDB structure. The residues in the chains are scanned in the $5' \rightarrow 3'$ direction and tested for membership in the assembled tetrad list (Figure 3.4). Should the current residue be a member of any tetrad and the previous residue isn't, this means an end of the current loop and the current residue represents the loop endpoint. On the contrary, if the current residue is **not** a part of any tetrad and the previous is, the previous residue represents a loop entry point.

This produces a triple $(sequence, entrypoint, endpoint)$ for each loop occurrence with a linear time complexity.

Loop type determination While the order of the loops and their sequences are identified, a relationship between the loop types and the entry/endpoints on the tetrads must be established in order to determine loop topology.

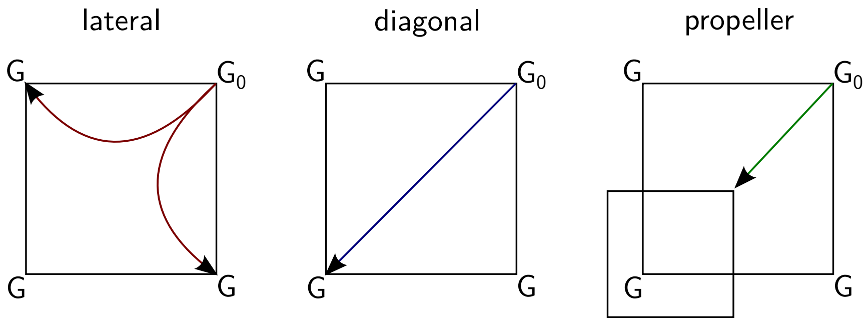

Figure 3.5: Relationship between endpoints and the sequence type.

Since the tetrads are mutually oriented, the task of loop type recognition is analogous to the relationship of the vertices in the rectangle (Figure 3.5). If both endpoints lie on the same tetrad, the loop is either lateral or diagonal with respect to their position. Should the endpoints be neighbouring guanines (or bonded, according to the paragraph 2.1.1), the resulting loop is of a lateral type. If not, then they are connected by a diagonal.

The endpoints connecting different tetrads are always classified as a propeller type. In theory, this could yield false positives if the loop connected anti-parallel strands, although in practice this loop would not be possible and neither would be a cyclic loop in the same chain.

3.4 Methods evaluation

The classification methods are evaluated on the dataset of 105 PDB structures retrieved from the RCSB PDB. The data set contains all the GQs groups according to the strand count[6]. The classification using geometric properties employed a machine learning for clustering and recognition. A 80-20% split rule was used to divide the dataset into training and test set respectively (84 samples in the training set, 21 samples for validation). Since each PDB structure contains multiple models, there are 537 models in the training set and 144 models in the test set. In order to evaluate the ability to correctly identify GQ structural motif, additional 10 randomly selected duplex/triplex structures[7] were introduced into the test set.

3.4.1 Methodology and test set

	Classified positively	Classified negatively
Positive sample	True positive (TP)	False negative (FN) (Type II error)
Negative sample	False positive (TN) (Type I error)	True negative (TN)

Table 3.3: Types of errors in evaluation.

The types of errors in evaluation are represented in the confusion matrix (Table 3.3) for binary classification. This can be easily extended for multiple classes, if we define *Positive sample* as *Sample belongs to class **A***, and *Classified positively* as *Classified as **A***, where ***A*** represents an arbitrary class, in this case a GQ structural family. The *Negative sample* can be defined as a structure that is not a GQ. With that, a GQ of a *"propeller"* type classified as a *"chair"* type would result in a Type II error, while a triplex structure classified as a *"chair"* type GQ would result in a Type I error.

[6] Monomeric, dimeric and tetrameric.

[7] False positive samples: PDB 1D3X, 1AT4, 1R3X, 1D3R, 1AO9, 1XJ9, 136D, 1W86, 1GN7, 1BWG.

The performance of both algorithms was evaluated using the two measures of performance - precision and recall. While the overall accuracy would suffice to measure the proportion of the correct classifications, the above metrics are more useful to show how the methods cope with different types of errors.

Precision The precision represents the proportion of the correctly classified samples over all positively classified samples, or, in another words, how many Type I errors are in the positively classified samples. With respect to the confusion matrix (Table 3.3), the formula for the precision can be written as:

$$Precision = \frac{TP}{TP + FP} \tag{3.1}$$

Both described methods use the same GQ structure identification method with the simulated tetrad assembly. The method was able to reject all non-GQ samples and accept all positive samples, therefore the $Precision = 1$ for both methods.

Recall (Hit rate, sensitivity) The recall metric is the ability of the method to identify all the positive samples. This differs from the previous metrics, as it doesn't take Type I errors into account, but rather shows how many of the positive samples are classified correctly. With the confusion matrix 3.3, the recall could be expressed as:

$$Recall = \frac{TP}{TP + FN} \tag{3.2}$$

GQ geometry-based classifier evaluation The use of geometric properties for semi-supervised classification was evaluated for Support Vector Machines, Nearest Neighbor and Decision Tree classifiers. Due to the high overlap of the parameter distributions between the classes, additional knowledge had to be incorporated into the training values - number of strands to separate clusters of monomeric GQs from di- and tetrameric GQs.

Different parameter coefficients are used for ML:

$$\begin{aligned} K_{planarity} &= planarity + 2.0 \times count(strands) \\ K_{twist} &= twist_s tddev^2 + 10.0 \times count(strands) \end{aligned} \tag{3.3}$$

The planarity is separated by 2Å(maximum permitted) per strand count (3.3), the twist angle variance by 10°. Either too low or too high separation hinders the classifier performance. The Decision Trees classifier shows the best separation of the data, however this test is prone to overfitting given the size of the dataset (84 structures, 537 models) and imbalanced weight of the classes.

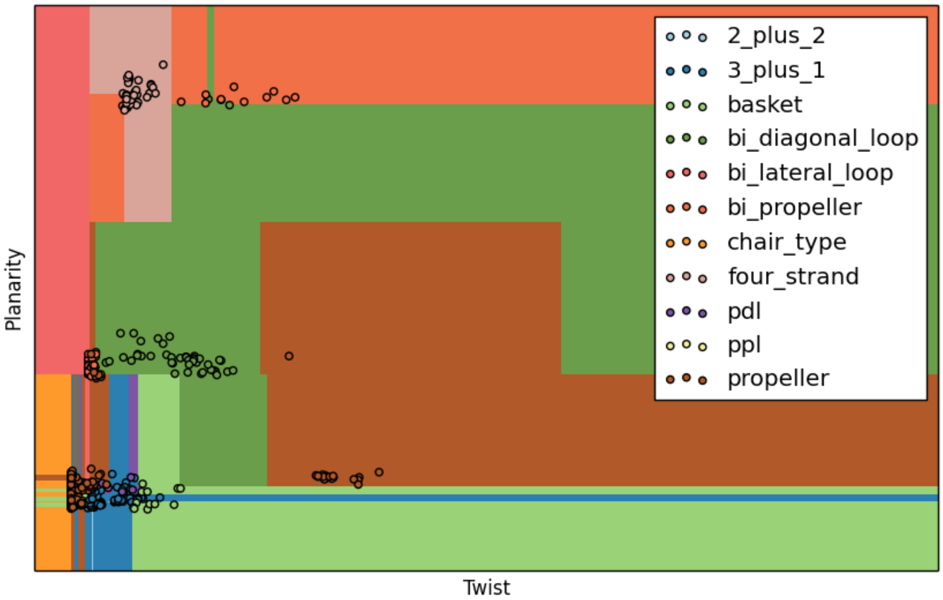

Figure 3.6: Decision surface of the Decision Trees classifier for the training data set.

Recall The classifier predicted correctly only 63 model topologies out of 144. This represents a hit rate:

$$Recall_{geo} = \frac{63}{63 + 81} = 0.4375 \tag{3.4}$$

While the separation thresholds and classifiers may be further refined, the overlapping spatial parameters make the prediction method inherently inaccurate for fully automated analysis.

Path tracing classifier evaluation Unlike the geometry metrics, the path tracing method is a deterministic algorithm, and thus doesn't require the data split and learning phase. The classifier identified correctly all regular samples, with the exception of the `PDB 2M53` (10 models) because of the presence of the zero-length nucleotide loop, where the propeller-type loop connects to an inner tetrad layer and possibly changes direction to the lower/uppermost tetrad. Such a pair of loops is misclassified as a single loop, therefore the `PPLP` topology was predicted to be `PLP`. This could be improved in the future by checking whether the intermediate loop entry point is in the middle tetrad. Such loop would always be strand-reversal, as it would otherwise need to cross the GQ center of gravity.

The classification also discovers extra loops, that were omitted in the earlier studies. For example the `2A5P`, classified as the propeller-type in the [Reshetnikov et al., 2010] contains an extra diagonal loop. Such extra features may or may not be the part of the GQ structure depending on the point of view. Fortunately, such errors are easily resolved in processing. Similar to the previous problem, some models within the same PDB structure may contain a different topology. This is a result of the natural GQ polymorphism, and a source of errors in previous studies. A good example is the PDB `2O3M` structure of a `PPLP` topology, but containing a single model without the lateral loop. While some deviating models were left in the structure, some were replaced as is the case of the `2KYP/2KYO` [Kuryavyi et al., 2010], which is the same structure evaluated in the two separate studies, folding into both monomeric and a dimeric variants.

Recall The path tracing classifier computed 688 models out of 698 correctly:

$$Recall_{pt} = \frac{688}{688 + 10} = 0.9857 \tag{3.5}$$

Results and limitations The sufficiently high hit rate (3.5) proves the applicability of the method to the fully-automated topology identification, which makes it a novel tool. The automated method also proved valuable, revealing the human errors in earlier classification efforts as described above. The further improvements to the algorithm would be a support for the structures with bulges in the stem, which structures were not available at the time of development. Moreover, [Webba da Silva, 2007] hinted at the possibility of the GQ multimerization into the G-wires. Such structures could still be described as a sequence of loops, although another tool would be necessary to distinguish inter-GQ from GQ connecting loops.

Chapter 4

GQ sequence recognition

The description of the GQ-forming sequence has made a prediction of the potential G-quadruplexes possible in the genome-wide pull-down, but with a caveat. The stability of such discovered structures could not be evaluated so easily for a large data set because of the complexity of the folding rules and factors involved. The recent research [Guédin et al., 2010] evaluated the effect of the loop length and the composition on the conformation and the structural stability with success, although a clear relationship is yet to be formed. To this date, there is no ideal method for either structure conformation prediction or the evaluation of the stability, although several best-effort methods have been explored with moderate success, depending on the use case. For example the molecular dynamics simulation produces reliable results and has been successfully used in the studies of most of the known structures so far, however it is very expensive (in terms of processing time) for mass-prediction of the topology from sequences. On the contrary, statistical and semi-assisted methods are able to produce both good or poor predictions, depending on the structural uniqueness, but they are also usually inexpensive (compared to the simulation), and therefore suitable for mass-prediction. The goal of this work is to present a comprehensive tool for the recognition and evaluation of a large number of sequences, thus the emphasis is put on the statistical methods, while the simulation is explored only as a way of populating the current database of known structures.

GQ-forming sequence prediction methods

The prediction of the GQ-forming sequences is focused on the unimolecular GQs for the specificity reasons. In theory, similar approach can be used for di/tetramolecular GQs as well with some extra data filtering, as the sequences are very short. Moreover, any unimolecular GQ can be interpreted as two (four) consecutive dimolecular (tetramolecular) GQs sequences. Likelihood of formation of that many inter-thread GQs is improbable, and thus the initial filtering would be needed to sort out the false positives.

The prediction of the intramolecular from the sequence prescript is prone to the same Type I error, as not all occurrences of the matching sequence actually form GQs for various reasons, such as the steric restrictions or structure instability in physiologic conditions. Unlike the previous case, the number of such errors is lower and can be mitigated by the stability prediction methods, like the free energy calculation or the the guanine/cytosine ratio. So the methods described in this chapter are not mutually exclusive, but work in parallel.

4.0.2 Regular expressions

The simplest GQ-forming sequence recognition methods leverages the known sequence format, as shown in formula (2.1), which can be interpreted as a regular language expression *(RegEx)*. For example the intramolecular GQ sequence can be expressed as:

$$G\{3,\}[ACGT]\{1,7\}G\{3,\}[ACGT]\{1,7\}G\{3,\}[ACGT]\{1,7\}G\{3,\} \tag{4.1}$$

The (4.1) is an example of a very specific expression which filters sequences starting with at least 3 guanines, followed by three repetitions of 1-7 arbitrary nucleotides tailed with at least 3 guanines.

For instance, the `GGGTTTGGGTTTGGGTTTGGG` would match, while the `GGGTTTGGGTTTGGGTTT` would not, because of the missing G-tract tail. This is not the only used RegEx, as for example the [Guédin et al., 2010] suggests loops longer than 7 nucleotides can be stable enough. Also, GQs with two tetrad levels have been discovered, suggesting the `G{2,}` would be needed to predict such GQs at the cost of additional Type I errors. As a result, the expression used varies depending on the specific goal or motive.

The RegEx filtering is implemented as a part of tool set (section 6.2).

G-Quadruplexes with bulges The G-Quadruplex structure is remarkably variable, as shown in the recent studies [Varizhuk et al., 2014] which present GQ structures defying the original sequence description by incorporating bulges in the G-tract sequences (the `G{3,}` motif in the (4.1)). The algorithm searches for similar sequence as in the RegEx method, but also allows a single imperfect G-tract with an emphasis on the defect position, something that would be harder to achieve with RegEx only. Such defects can produce two types of structures - either an imperfect tetrad (for example a 4-tetrad quadruplex, where the last G-tract contains a bulge would still produce a stable 3-tetrad) or a perfect tetrad that folds despite the bulge. Fortunately, from a correlation between the stability of the final structure conformations and the position/type of defects, specific regular expressions can be derived[1].

4.0.3 Secondary structure folding

The sequences predicted with the RegEx method or a more sensitive scanner evaluate sequence pattern only and inherently are prone to Type I errors. This stems from using only a partial information, such as a sequence motif, without taking steric considerations and environment into consideration. One such method that is able to refine (or score) the results is the heuristic prediction of the energy model for such sequences, using the minimal free energy (Mfe) as the scoring function to determine sequence stability. [Lorenz et al., 2012] presented both the energy model and a tool set, albeit for the RNA G-quadruplexes only. The reason is that, unlike the DNA quadruplexes, the RNA GQs appear to be structurally very monomorphic, forming tetrameric parallel-stranded conformations regardless of the physiologic conditions. While this makes the energy model for RNA GQs

[1] [Varizhuk et al., 2014, p.8, Table 1] evaluates such formulas.

possible with currently available data, it cannot be however applied to the DNA sequences.

ViennaRNA Package The ViennaRNA Package implements the secondary RNA structure prediction using the free energy minimization approach, and as of the version 2.0 [Lorenz et al., 2012], supports the RNA G-quadruplexes as well. Because of the focus of this work on a classification and prediction of a broad range of GQs, the `RNAfold` runnable for the user interface was implemented for the sake of completeness. However, the applicability of the method on the DNA sequences was not further explored due to the strong DNA GQ polymorphism and the lack of experimental data. Moreover, the accuracy of the method is not yet benchmarked for a large-scale data set, although an upper bound of Type I errors is estimated to 1.4% from the screening of the Rfam database [Lorenz et al., 2013].

4.0.4 cG/cC score

Another recently studied method, that is able to further refine the predicted results is a heuristic scoring based on the inferred sequence parameters. Akin to the prediction based on the free energy minimization [Lorenz et al., 2012], the method is evaluated mainly for the RNA GQs for the appreciably simpler folding model. The applicability on the DNA GQs is neither accepted or rejected. The cG/cC scoring system, presented in the [Beaudoin et al., 2014] is based on the observation of the nucleotide composition of the loops (section 2.3). Based on the data from already evaluated structures, it appears that the cytosine contents is what governs the RNA GQ folding mechanism.

Scoring formula The formula for scoring a consecutive run of a nucleotide *N* could be defined as follows:

$$cN(s) = \sum_{i=1}^{n}(|Ns(i)| * 10 * i) \tag{4.2}$$

For instance a score of a consecutive run of `GGG` would be counted as: 3 singlets, 2 doublets and a single triplet, which would result in a final score of:

$$score(cG) = 3(G) \times 10 + 2(GG) \times 20 + 1(GGG) \times 30 \tag{4.3}$$

It can be seen(4.3), that all substrings of length up to *n* are counted and weighed according to the substring length.

With the formula (4.2), the final score for the ratio of the `G, C` runs can be expressed as:

$$cG/cCscore = \frac{cG(s)}{cC(s)} \tag{4.4}$$

Evaluation The scoring was evaluated [Beaudoin et al., 2014] in comparison to the ViennaRNA energy minimization method, and yielded 100% sensitivity and 83.3% specificity[2] with a threshold of $cG/cC = 2.05$. The general rule is that, in order to increase the specificity, a higher threshold should be chosen. Since this parameter is specific to the evaluated sequences, it was left as a variable in the cG/cC runnable implementation (section 6.2).

For the same performance evaluation, ViennaRNA energy-based model prediction yielded only 40% specificity with the same 100% sensitivity and roughly linear ROC curve ([Beaudoin et al., 2014, Figure 7].). The dataset included only 14 samples, so an assessment of the relationship between scoring function and GQ folding of a large set of RNA GQs is yet to be seen, which presents a similar challenge as with the energy-based models; to date, very little knowledge about RNA/DNA GQ folding exists to infer reliable prediction rules. For this reason, both sequence prediction methods are included in the tool set for either cross-validation or as a convenient complementary tools to help identify incorrect folding predictions.

[2] With respect to the confusion matrix (Table 3.3), the specificity can be defined as a hit rate for negative samples, e.g. $Specificity = \frac{TN}{FP+TN}$.

Chapter 5

Structure prediction

The main goal of the methods described in the previous chapter is to predict potential GQ-forming sequences. A natural course of action would be to classify recognized sequences into groups with respect to the conformation topology with the PDB models (chapter 3). While this idea is not new, but has proved even more challenging than the GQ-forming sequence recognition and evaluation for the similar reasons. [Neidle, 2011] states that *"The prediction of quadruplex topology and stability from knowledge of sequence alone is ultimately key to understanding the large amount of bioinformatics data on quadruplex sequence occurence. Experimental structural data provide an essential starting point, although there is as yet very little data on the modeling of unknown structures based on existing ones, analogous to protein homology modeling."*. In another words, this also means that, while there are several analyzed GQ structures from which some folding rules can be inferred[1], the applicability of the inferred rules for the not yet known structures is yet to be answered.

Similarly to the sequence recognition, there have been two general approaches - heuristic and regression analysis, and energy-model based simulation. These approaches are not mutually exclusive. The simulation is able to precisely determine minimal energy for any given conformation, but at the cost of lengthy processing time, which makes it impractical for genome-wide screening, where the heuristic methods are able to provide a best effort hints and/or suggest the most probable conformations for the simulation.

[1] For instance the cG/cC scoring method leverages the correlation between the nucleotide content in RNA quadruplexes and stability.

5.1 Structure prediction heuristics

Several studies used a statistical approach based on the knowledge inferred from the known structures [Wong et al., 2010]. More interestingly, [Fogolari et al., 2009] published an approach that used the available structures from the PDB to break down the GQ structures into fragments (stems and loops), each expressing a different geometry and a nucleotide sequence. Such fragments were then reassembled into various models, which were refined by the energy minimization using MD simulation, and finally clustered. Using this method a data set of 14 available structures was used to produce a library of 4418 structures covering more than half of the possible topologies. The library was cross-validated against the original sample with success, although it is clear that the predictive power for (at the time unknown) sequences could not be measured. Despite the drawbacks, the approach has proven not only as a potentially very powerful method once more data is available, but also provided several interesting observations:

- The predicted models were dominated by parallel topologies.
- Same sequence can adopt a variety of conformations, which proves either MD simulation or NMR resolution challenging without base modification.
- Distribution of loop lengths is uneven (although not completely random). Moreover, while there appears not to be a direct relationship between loop lengths and the topology, there is a preference for a loop type for certain loop lengths. Typically, the longer the loop, the more topologies are possible.

Based on these findings, a simpler (and on the other hand faster, therefore suitable for large-scale analysis) method for structure prediction is proposed and implemented in the tool set (chapter 6), which uses the structures known today to find a possible correlation between the sequence fragments and loop topology.

5.1.1 Loop length configuration

While the direct relationship between the loop length and type is yet to be confirmed, [Fogolari et al., 2009] reveals a possible connection between the lengths of the loops, and the overall topology. The loop length configuration can be defined an ordered sequence:

$$len(L_1)len(L_2)len(L_3) \tag{5.1}$$

where the L_x denotes one of the three loops in the intramolecular GQs. For example, a configuration `123` represents a GQ, where the L_1 loop length is 1, L_2 loop length is 2 and the L_3 lengths is 3.

Interestingly, the study shows several most frequent configurations - `434` (122 models), `131` (53 models) and `113` (53 models), for which the [Fogolari et al., 2009, Table 2] assigns loop topologies (in respective order) `DPD`, `PPP` and `PPP`. Compared to the classified PDB structures from the `quadclass` tool, this still holds true today. Another interesting observation is the relative frequency of the most frequent `PPP` loop topology[2], which (although affected by the subjective availability and the interest in specific structures) is still consistent with the results on the available dataset.

Based on these observations, the mapping of a single loop configuration to the possible topologies could be written as:

$$loop\,configuration \rightarrow [list\ of\ topologies] \tag{5.2}$$

To overcome the ambiguity of the loop configurations with more than 1 known topology, a weighting system is proposed in form of a p-value:

$$P(configuration, topology) = 1 - \frac{count(configuration)\ in\ topology}{count(configuration)} \tag{5.3}$$

Since the prediction is not binary, p-value must be defined with respect to an arbitrary topology (5.3) as the ratio of occurrences of the loop length configuration in given topology to all occurrences of the topology.

[2] [Fogolari et al., 2009] topology distribution states 39.93% of the models were `PPP`, currently available experimental data consists of 41.13% `PPP` models.

For instance, let:

$$\begin{aligned} 131 &\rightarrow [(1)PPP, (3)LLL] \\ P(131, PPP) &= 1 - \frac{1}{1+3} = 0.75 \\ P(131, LLL) &= 1 - \frac{3}{1+3} = 0.25 \end{aligned} \tag{5.4}$$

The p-value is inherently normalized to the $\langle 0, 1 \rangle$ range, where the lower bound 0 represents the most significant (and therefore best in terms of predictive power) and the upper bound the least significant (most frequent and/or ambiguous) mapping. The normalization also allows the use of the significance level α as the threshold for the likelihood of the predictions.

Limitations Since the predictor relies heavily on the knowledge about the currently known structures, the ability to present useful predictions for complexities, like inclusion of the loop DG into the tetrad core or loop extensions, will require much better understanding of the GQ folding rules, than simple length measurement. On the other hand, the loop length is observed [Guédin et al., 2010] as one of the main driving forces in the folding of DNA quadruplexes[3]. The loop length thus could provide a reliable predictor, as shown in [Bugaut and Balasubramanian, 2008], that investigates the effect of the loop length with randomized sequences to GQ stability.

5.2 Novel predictors

5.2.1 Loop length derivation

The relative abundance of unique topologies presents a useful predictor, its predicting power is limited only to the the loop configurations of the same length as observed. In another words, the configurations `131, 113` from the study [Fogolari et al., 2009] predict a `PPP` topology, but is incapable of predicting for instance `121` configuration outcome. This is impractical for the large-scale prediction, as the

[3] As opposed to the RNA quadruplexes, as noted in [Lorenz et al., 2012].

GQ-forming sequences often vary in lengths of the loops, therefore the previous method could be applicable to only a small percent of the structures. To overcome this problem, either better understanding of the folding is necessary or a more general approach has to be devised.

From the gathered structure metrics on the available GQs in PDB using the by the path tracing (section 3.3), the affinity of some loop lengths to form certain loop types was confirmed. For example, the short loops of a 1 nucleotide length often form the propeller-type loops (subsection 3.2.5). While this knowledge is not enough to confidently predict all three loops, the same principle can be applied on the pair of loop lengths. For the sake of clarity, assume a linear function is defined by the two points (Figure 5.1), where each points represents a loop. The x-value represents the ordering between the loops, while the y-value represents the loop length. The derivation of the function then defines whether the loop length increases or decreases.

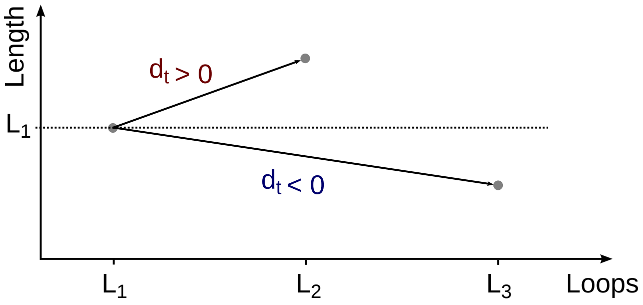

Figure 5.1: Simplified loop length derivations between $L_1 \rightarrow L_2$ and $L_1 \rightarrow L_3$.

In order to generalize the result, the nominal value of the derivations may be omitted and replaced with a sign, according to the function:

$$dt(y) = \begin{cases} dt(y) > 0 & : \quad \text{“+''} \\ dt(y) = 0 & : \quad \text{“ =''} \\ dt(y) < 0 & : \quad \text{“−''} \end{cases} \tag{5.5}$$

For instance, the relationship between the pair of lengths $(5, 3)$ is represented by a ″-″ sign, while a $(5, 5)$ would be written as a ″=″ sign.

K_1 **predictor** Since there are three loops in the intramolecular GQ, there would be:

$$C(2, 3) = \frac{n!}{k!(n-k)!} = \frac{3!}{2!(3-2)!} = 3 \tag{5.6}$$

possible pairs of two. Fortunately, the significance of the pairs can be measured in the same way as the loop length configurations using p-values, and compared to the previous method to identify the most discriminating pairing with the least information.

The $L_1 \leftrightarrow L_2$ and $L_1 \leftrightarrow L_3$ pairs have shown[4] to be most significant as with many structures, the L_1 and L_3 seem to be dependent. This could be caused by high variability of the longer loop in the middle, and the shorter loops on the sides, as it can be observed in the table5.1 for the predictor K_1, defined as a sequence of loop derivations for all viable combinations:

$$K_1 = [dt(L_3, L_1), dt(L_1, L_2), dt(L_2, L_3)] \tag{5.7}$$

where the difference between the fringe loop lengths (dt_{31}, first symbol) is often ″=″. Peculiarity of this predictor is, that the p-value for some sequences may be lower than using previous method, for instance the ″===″.

This is partially caused by overfitting, as the some datasets are much larger than other. But also because of the absence of scale in this predictor. For instance a p-value for the aforementioned ″===″ is 0.2 lower for K_1, than in the previous method. This is based on

[4] The `seqlearn` (section 6.1.6) tool reports p-values for both predictors side to side for comparison in columns 2 and 3.

the occurrence of the "`111/===`" configuration. In another words, it means that the ubiquitous $(TTA)_3$ loop repeat is 20% more likely to fold into a propeller-type GQ, or that a $(AAAAAA)_3$ loop sequence folds into the propeller-type with the same chance. Both assumptions are wrong as it has been already shown, that the folding rules differ with respect to the loop size. Therefore it is not possible to infer a knowledge about the `666` configuration from the `111` configuration.

K_2 **predictor** The p-values for the K_1 predictor were evaluated on the propeller-type GQ family, as it is the most widely represented in the dataset with the problems mentioned above. As the table5.1 for this GQ family shows, the derivation of the loop lengths alone is not significant enough to distinguish between GQ structure types, with uncertainty for almost every configuration. This is also confirmed in the chart Figure 5.2 comparing the p-values for all samples.

To mitigate both the absence of scale and the high p-values, the length of the opening loop is introduced to the predictor, replacing the dt_{31} symbol. The opening length was chosen as it's the most conserved, with only a few families starting with an opening diagonal loop (models *12a/12b* in Figure 3.1). Moreover, the sense of scale is now in line with the experimental data. For instance, propeller-type loop length configurations of `1N1`, where $N = \{1, 2, 3, 4, 5, 9\}$ suggest that a `161` is likely to assume propeller-type topology as well, but no knowledge about the `333` can be derived, unlike with the K_1 predictor.

Based on the experimental results, the K_2 (Table 5.1) predictor was evaluated as the most discriminating, with the same amount of information. The most problematic loop length combinations are the most frequent ones[5], as with the previous method. In addition, sequences containing a short opening loop may still be misclassified as propeller-type because of the overfitting. The K_2 however is not the definite predictor, and should be evaluated with the more data available. The inclusion of the specific loop lengths improves the specificity, but at the cost of sensitivity.

[5] For example `333` and `232`.

Propeller-type GQs

	$dt_{12}dt_{23}dt_{13}$		$L_1dt_{12}dt_{13}$	
Loop lengths	Predictor K_1	P-value	Predictor K_2	P-value
123	−++	0.000	1++	0.500
111	===	0.600	1==	0.000
121	=+−	0.200	1+=	0.000
121	=+−	0.200	1+=	0.000
121	=+−	0.200	1+=	0.000
333	===	0.600	3==	0.800
151	=+−	0.200	1+=	0.000
313	=−+	0.000	3-=	0.000
191	=+−	0.200	1+=	0.000
111	===	0.600	1==	0.000
141	=+−	0.200	1+=	0.000
151	=+−	0.200	1+=	0.000
232	=+−	0.200	2+=	0.667
321	+−−	0.667	3--	0.500
343	=+−	0.200	3+=	0.500
151	=+−	0.200	1+=	0.000
111	===	0.600	1==	0.000
121	=+−	0.200	1+=	0.000
131	=+−	0.200	1+=	0.000

Table 5.1: Comparison of the loop length-only predictor (K_1) and the combined loop length derivations with the length of the opening loop (K_2) for the propeller-type GQs. (Lower p-value is more discriminating, therefore better.)

Limitations In general, the longer the loop length is, the more challenging it is to understand, which is a shared problem with both the stability and structure prediction. The prediction from sequence only also does not take the nature of the central cation into consideration (and environment in general) into consideration, which significantly affects the prediction performance. For example, the 333[6] is partic-

[6] Coincidentally, this exact $(GGGTTA)_3GGG$ sequence repeat is overrepresented in the telomeric regions of the mammalian genome, and therefore intensively studied.

ularly challenging, as it has been observed to form either propeller-type, 2+2, basket, or 3+1 topologies for the intramolecular GQs only. Despite the limited predicting power for such ambiguous samples, most of the evaluated sequences are unique enough with p-values nearly the same (Figure 5.2) as for the loop length configuration, but for much broader range of sequences.

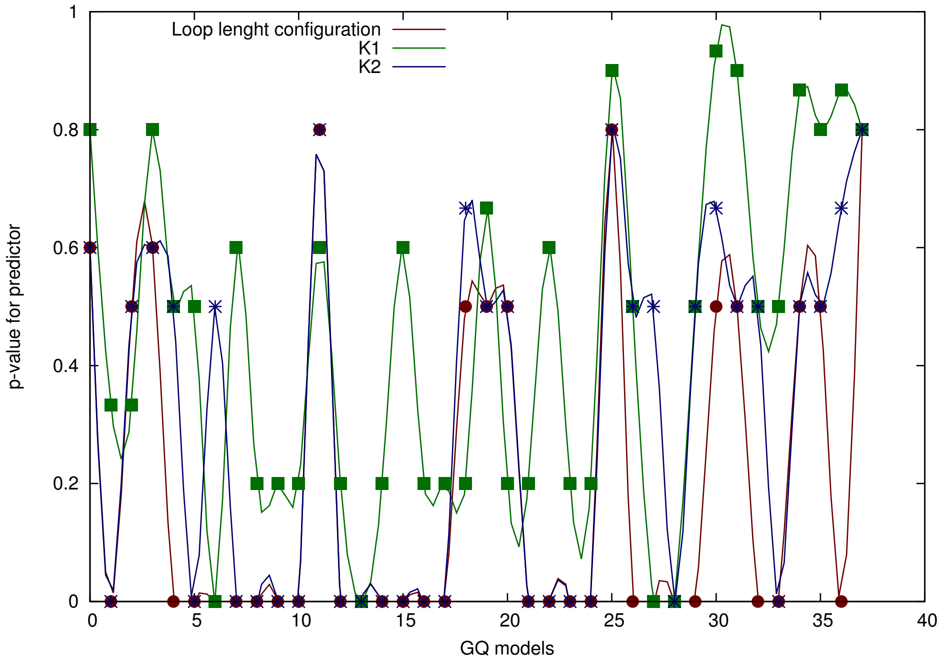

Figure 5.2: Plot of the p-values of the K_1 and K_2 predictors in comparison with the previous method.

5.2.2 Loop sequence composition

Contrary to the DNA GQs, RNA folding model appears to be simpler, as it relies mostly on the nucleotide composition. This observation has been already used heavily in structure stability prediction, for instance QGRS Mapper includes the ratio of G nucleotides in the "G-score ([Kikin et al., 2006])", or the cG/cC score (section 4.0.4) described in the previous chapter. The usefulness of this predictor on the DNA quadruplexes is neither disputed or confirmed, although some studies evaluated the preference of loops consisting of a particular nucleotides to form a certain sequence, for example [Cang et al., 2011] evaluates effect of the T_n sequences on the loop types.

The sequence composition predictor may be written a quartet of:

$$K_3 = [p(A), p(C), p(G), p(T), p(U)] \tag{5.8}$$

where the $p(N)$ represents the relative frequency of the nucleotide N in the loops:

$$p(sample, N) = \frac{count(sample, N)}{count(sample, \{A, C, G, T, U\})} \tag{5.9}$$

Unlike the previous methods, the sequence composition is highly varied and therefore an equality match and p-value significance is not applicable, because most of the models are unique. The match of a sample against a model is represented by an accumulated error, defined as the sum of differences for all nucleotides:

$$K_3error(sample, model) = \frac{1}{5} \sum_{x=A}^{\{A,C,G,T,U\}} |p(sample, x) - p(model, x)| \tag{5.10}$$

Since the sum of the composition is equal to 1, the error value can attain a number from $\langle 0, 1\rangle$, where the 0 represents an ideal match and the 1 a complete mismatch. The composition similarity can be then defined as a $(1 - K_3error)$.

The fitting function for this predictor attempts to find a composition with a minimal error value and returns a single predicted model along with the error. Due to the inapplicability of the p-value on this predictor, the error value must be interpreted by the user on an arbitrary significance level, unlike the previous methods.

Molecular dynamics

The molecular dynamics simulation (MD) is often used in the studies of the structure stability. The thesis [Cang, 2010] explored the applicability of the MD to the structure prediction problem with a goal to investigate the driving forces behind the folding rules of the GQ. The main advantage of the MD in comparison to the heuristic functions is that the environment, and the binding cation, may be taken into consideration. For example, when evaluating the effect of both Na^+ and K^+ the results of the MD are consistent with the NMR for structures bound by a single cation, but differ with multiple cations. That is caused by either cation repulsions or broken hydrogen bonding, requiring a refitting of the parameters to compensate. Moreover, it has been shown that a certain GQ sequences readily form almost any known conformation, but other rarely do so in vitro, which requires modified nucleotides to stabilize the structures for an assessment.

This level of detail is obviously superior to the heuristic approach, but also very time and computational power intensive. The goal of this thesis is both large-scale structure and sequence analysis, therefore the method is impractical to use for hundreds and thousands of GQ sequences. On the other hand, the statistic approach today suffers from the lack of data on many GQ families, and the MD could be used to extend the learning set with predicted structures, not yet confirmed to fold in vitro. In reverse, the performance of the MD could benefit from the results of the heuristic methods to reduce the number of potential conformations.

5.2.3 Heuristic predictors summary

The slow expansion of the experimental data allowed to define a set of rules that specifies the influence of sequence parameters like e.g. length or composition on the overall topology of the GQs. It has also been found that the inferred rules governing the GQ folding differ with the DNA/RNA quadruplexes and also the strand count. The proposed predictors make use of these observations on the GQ structure data set available to date. The principle of the loop length measurement was generalized to a variety of sequences with the novel loop length derivation on the assumptions from studies ([Guédin et al., 2010]) that suggest a preference of the short loops to certain loop types and the loose upper limit on a maximal loop length, which can be stabilized by the two short loops. This relaxed loop limit is represented by the derivation of the loop length function instead of the nominal value.

The predictors are able to provide a list of candidate conformations (except the sequence composition predictor that is able to predict only one topology) along with a degree of confidence expressed by the p-value. The p-value threshold is however left to the user, as it varies depending on the application. If the results are used as a filter for candidates in MD, the threshold for p-value should be chosen with respect to the number of predictions and required accuracy.

5.3 Prediction methods evaluation

The interface provides the p-values and all possible predictions for open interpretation, an objective measure is however required for evaluation, so each of the predictors is evaluated separately and compared to others. The methodology is similar to the structure classification, but because of the independent evaluation, it can be further reduced to the binary classification problem. Moreover, the prediction algorithm can presume that given sequences are GQ-forming, as this decision problem is solved beforehand using the algorithms described in the sequence recognition chapter (4). The prediction algorithm is then able to predict a topology, which is either true or false and provide a confidence level (p-value). This relationship between the accuracy and given confidence threshold is well expressed by the receiver operating curve (ROC curve), revealing the dependency of the rate of positive predictions on the rate of false predictions.

Training and test set

The same 80-20% dataset split from the topology classification (section 3.4) evaluation was used. In addition, a 70-30% split was introduced to illustrate the predictor behaviour with a lack of learning data.

5.3.1 80-20% ROC curve comparison

The ROC curve (Figure 5.3) shows all three predictors with the center line. The center line represents a baseline of random predictions, where the false positive is equally probable as the true positive. Line points below the baseline present worse than random predictions, while points above the baseline present better predictions. An ideal prediction would be in the top left corner $[0, 1]$, which represents a case where all the predictions would be true positives.

The novel loop length derivation K_2 predictor shows a strong predicting power, as the TPR is above 0.8 with still zero FPR. The length match predictor suffers from the lack of equal-length loops in the training set, and produces almost a half of uncertain predictions, which is expected due to the high variability of the samples. The sequence composition shows an above-average characteristic, but also worse than random.

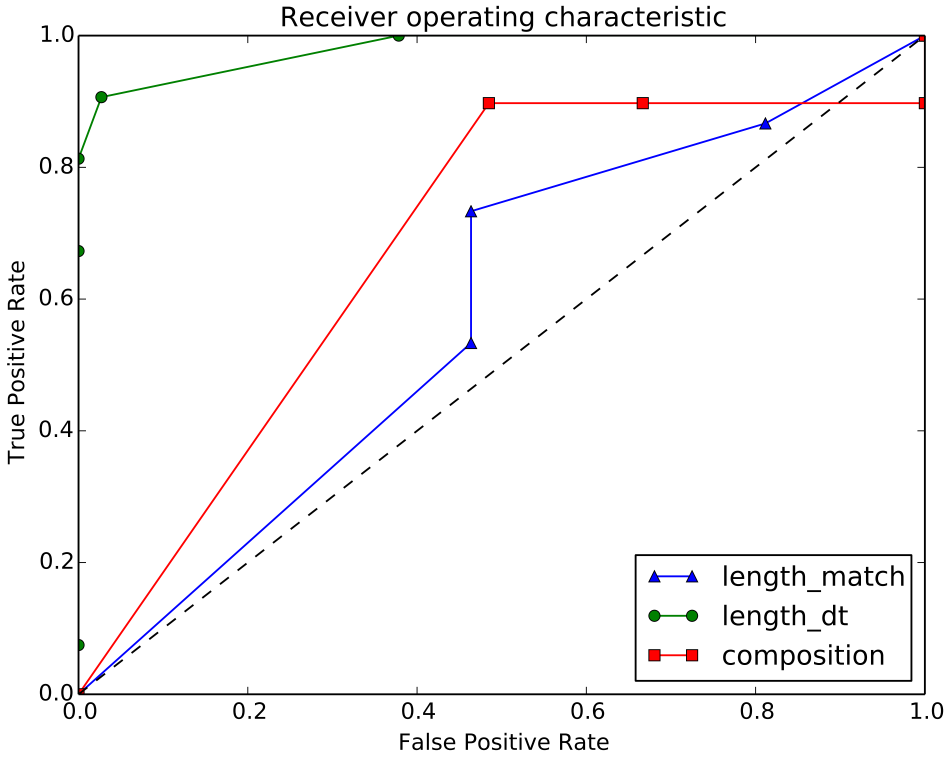

Figure 5.3: ROC curve for the 80-20% data split.

5.3.2 70-30% ROC curve comparison

Unlike the 70-30% split, the half split shows how the lack of the training data affects the predictor performance. Interestingly, the performance of the length match and the composition predictors improves, which could be explained by the shift of a portion of similar sequences to the training set, which is in line with the lower ratio of uncertain predictions of the loop length predictor.

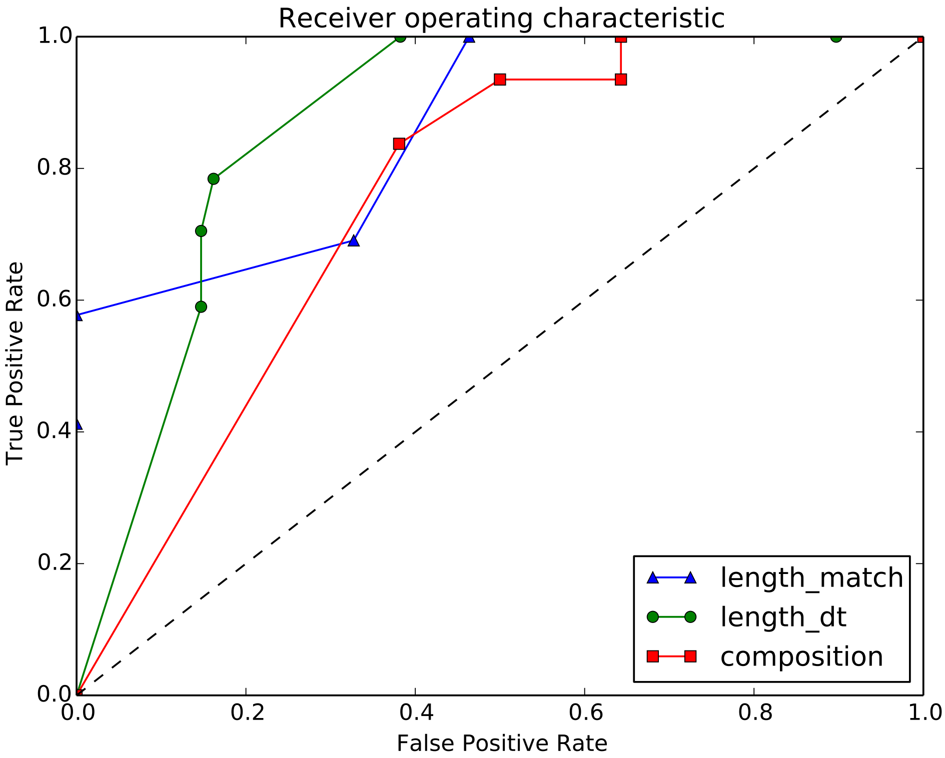

Figure 5.4: ROC curve for the 70-30% data split.

5.3.3 Evaluation

The prediction results suggest an overall better performance of the loop length-based predictor. That correlates with the hypothesis that a loop length plays a significant role in the DNA GQ folding. The nucleotide composition predictor shows a highly unstable prediction performance based on the training set size. Given the small size of the training set, the effects of both over and underfitting can be seen on the loop length-based predictors as well, but are much less pronounced.

While the database size today is not large enough to prove a predicting power on a statistically significant sample, the stable above-average predictor performance shows a promising heuristic method for the computationally intensive precise analysis.

Chapter 6

Implementation

6.1 Quadclass library

The developed tools are separated into a library for GQ analysis, CLI tool set, and a web interface for convenience and integration in the sequence analysis workflow.

6.1.1 Dependencies

The library and tool are written in Python, using the BioPython library to access read and write interchangeable data like FASTA[1] and PDB. The quadlearn utility additionally requires an sklearn[2] library for machine learning.

6.1.2 Library design

There are tools for both automated classification of the PDB structures and the GQ-forming sequence predictions. The `tetrad.py` implements most of the functions for PDB structure analysis and the `seqlearn.py` contains functions for sequence analysis, prediction and fitting. An example of the library use is the quadclass (section 6.2) runnable implemented in the web interface, that uses the library for prediction and fetches information about the families.

The library organizes known GQ structures in the directories with symbolic family names (e.g. `3_plus_1` (Figure 3.2) or `pdl`), where each directory contains a `list` file with names of the PDB structure

[1] http://www.ncbi.nlm.nih.gov/BLAST/blastcgihelp.shtml

[2] http://scikit-learn.org

names that belong in that quadruplex family. Optionally, the directory may contain a `DESCRIPTION` file, where the brief class information can be written in the Markdown format[3]. The symbolic GQ family name should always be on the first line, with the description following. This simple storage format was chosen its the practicality over binary database for a small number of entities.

The CLI tools use a text format as well, with only the quadclass (section 6.1.4) tool produces an additional output in the TSV[4] file for further processing. More specific information about the tools can be invoked by the `-h` or `--help` parameters.

6.1.3 pdbfetch

This pdbfetch is a helper tool to fetch PDB files from a remote RCSB PDB servers. The only parameter is directory, that contains a `list` file with PDB structure names. The tool checks the existence of the PDB files in that folder and downloads only missing files. The source code does not include PDB files, but only lists, therefore populating the directories with PDB files is required before the structure analysis.

```
$ ./pdbfetch.py 3_plus_1
```

Figure 6.1: Example of the pdbfetch usage to fetch PDB structures belonging to the 3+1 family.

6.1.4 quadclass

This tool is an interface for the `tetrad.py` PDB structure analysis library. Similarly to the pdbfetch tool, the tool accepts an optional directory name (or list of directories) as a parameter. However, if no parameter is supplied, all known intramolecular GQ families are evaluated, and the metrics are written to the `gqclass.tsv` TSV file for further processing.

[3] `http://daringfireball.net/projects/markdown`

[4] Tab-separated values.

The tool performs both structure geometry analysis (section 3.1) and path tracing (section 3.3), and writes the measured values in a tab-separated file. The tool also identifies invalid or non-GQ PDB structures by the tetrad assembly, such structures are rejected and do not appear in the output file, but are still printed to the standard output, which looks like follows:

```
$ ./quadclass.py test
> processing "test/1EAM.pdb"
 [!!] less than 8 DGs found, ignoring
> processing "test/1EVM.pdb"
 * scanning model <Model id=0>, 12 DGs
  - assembly: 3 tetrads
  - mean planarity: 0.65 A, stddev 0.17 A
  - twist angle: 0.44 rad (25.32 deg), stddev 0.07
  - chains: 4, consensus: AGGGT
  - topology: O, fragments: GGGT
```

The PDB files usually have several models built-in, so the analysis is performed for each of them. The resulting TSV file contains following data columns:

Planarity	stddev	Twist	stddev	Chains	Topology	Loops

It should be noted that only the classification of the intramolecular structures is discussed in the thesis, as it is useful for the sequence prediction. The quadclass tool is however capable of classification of dimeric and four-stranded structures as well. The O in the example usage represents and open loop, therefore a four-stranded structure. Moreover, some PDB files surprisingly contain different topologies within the models, such as the the 2MGN.

Input filtering The quadclass scans the PDB structure for guanines for tetrad assembly. At least 8 guanines are required to form a tetrad, so the PDB structures with less than 8 guanines may be filtered out. Some structures contain either inosine or modified guanines, which is accounted for in the input filtering, however currently unknown modifications of guanines have to be incorporated into the tetrad library, otherwise guanine discovery would reject such residues. The tetrad library requires at least 2 complete tetrads, that are recognized using the algorithm described in chapter 3.2.1, section 3.3.1. PDB models with less than 2 tetrads are rejected. Passed models are subjected to the path tracing algorithm and geometry analysis.

6.1.5 quadlearn

The tool for learning and fitting, using the the PDB structure spatial metrics. The input is a training and test sets, both in form of a TSV file produced by the quadclass tool. The ML learning algorithm is trained on the twist angle and planarity metrics (section 3.2.1) parameters from the training set and plots a decision surface along with the estimated GQ family predictions for each entry in the test set.

Example of the quadlearn usage to fit a control set:

```
$ ./quadlearn.py -t train90.tsv control.tsv
```

6.1.6 seqlearn

The tool for prediction of the GQ topologies from the sequences. The input of the program is a sequence (or list of sequences) as parameters, and optionally a training TSV data set. If no sequence is passed to the tool, the whole training set is processed and the overview of all predictors including the p-values is printed to the standard output. The seqlearn tool includes all the predictors described in chapter 5, named as: `length_match`, `length_dt`, `composition` respectively:

```
$ ./seqlearn.py GGGTTGGGTTAGGGTTGGG
> GGGTTGGGTTAGGGTTGGG ...
232 2+= GGGTT|GGGTTA|GGGTT
propeller (PPPD)
 * length_match..'232' p-value=0.50
 * length_dt..'2+=' p-value=0.67
```

Loop length predictor Unlike in the structures, the loop starting and ending points are not known in the string sequence. The loop fragments are identified by looking for a non-G → G transition, representing a G-tract start. When in a G-tract, first non-G symbol represents a loop start. The loop is closed by a sequence of 2 consecutive Gs (which are part of the G-tract of the following fragment). It may happen, that a sequence of 2+ Gs happen in the middle of the loop, such input would be interpreted as a false loop closure. To overcome this, identified fragments with shortest G-tract are merged until there are at most 3 loops for intramolecular GQs.

The loop fragments identified by this method contain both the stem and the loop parts for completeness. For example, the sequence composition predictor takes the stem into account as well. For loop length-based predictors, the loop start is extracted from the fragment using the tetrad stacking rule - the number of Gs forming the tetrads must be the same in all 4 G-tracts.

Loop length derivation The K_2 predictor defined in section 5.2.1 was used.

Loop sequence composition The composition is calculated from the fragments, taking G-tract length into account as well.

6.2 Web interface

There is a web interface implemented on top of the library and utilities for convenience and integration into the existing workflow. The interface is built using the Python programming language, and uses the BioPython library for reading and writing interchangeable format. In addition, the web interface requires Flask[5], and Markdown[6] bindings for rich text processing. The RNAfold runnable (section 6.2) requires the ViennaRNA Package RNAfold tool installed.

[5] `http://flask.pocoo.org/`

[6] `http://pythonhosted.org//Markdown`

Datasets

The interface uses the standard FASTA file format as a basis for the sequence set processing. The files reside in the `data` directory, but may also be uploaded from the web interface using the *"upload new dataset"* in the header.

Runnables

The aim of the web interface is an integration with the existing sequence analysis workflow. Admittedly, the web interface may not be usable for the broad range of sequence analysis operations, but fortunately many tools produce results in interchangeable formats. Such results may be uploaded to the interface as datasets, then the filtered data may be pulled from the web interface in form of a FASTA or GFF[7] file formats, and reintegrated in the workflow.

The interface supports a generic interface for various data filtering operations called runnables. At the moment, several runnables are implemented for the quadruplex prediction pipeline - from sequence recognition, stability evaluation to structure prediction. The concept however is not limited for GQ prediction only, and it could be interesting to enrich the available filters with tools for genomic surveys.

RegEx The RegEx runnable implements string search using the regular expressions over sequences in the FASTA file format. The predefined query is an example GQ-forming sequence search. The output of the runnable is a FASTA file with the filtered out sequences.

cG/cC score The cG/cC score runnable implements the algorithm described in the sequence recognition chapter 4.0.4. The input of the algorithm is a FASTA file and the parameter is a threshold for the cG/cC score cutoff. The output of the runnable is a FASTA with sequences, for which the score was below the set threshold. The web interface interprets the result in an interactive table sorted by the

[7] `http://www.sanger.ac.uk/resources/software/gff/spec.html`

score, and displaying the accession number (clickable), and the original sequence. Moreover, the chart of the data per-quartiles and the histogram of the data distribution is shown.

RNAfold score As of the ViennaRNA Package 2.0, the prediction of the RNA GQ stability is supported. The runnable is a wrapper for the ViennaRNA Package "RNAfold" tool, that accepts the input in the FASTA file format, with a parameter that sets the threshold for the minimum free energy cutoff. The default value is set to -20.0 kcal/mol. The web interface interprets the result in the interactive table sorted by the minimum free energy, displaying the accession number, predicted fold and the original sequence.

Quadclass The quadclass runnable should be used on the predicted GQ sequences (previously filtered by any of the three runnables), and produces the GQ structure prediction from sequence using the seqlearn (section 6.1.6) tool. The sequences in the table are annotated by the potential predicted topologies. The table rows are interactive and reveal more detailed breakdown on the predicted topologies, p-values and reasons after table row expansion.

Filtering

The runnables are designed to be run iteratively, where each run refines the previous results. Most of the runnables support a threshold to narrow down the filtered data, and may be combined. The plotted distribution provides a visual aid in the data filtering process.

Persistent searches The web interface also supports saving and resuming of the iterative searches. Each search footer has the option to either start a new search or save the search for later. Such searches may be resumed using the *"Saved searches"* link in the interface header.

6.3 Limitations

The aim of the web interface is not to replace a complete sequence analysis pipeline. While the runnables allow for more filters, only a

GQ-forming sequence related runnables are implemented with only FASTA input and output. For some applications like the structure prediction, GFF output might be more useful for integration with existing genome annotation tools, for example to reveal a correlation between a gene promoter positions and GQ family variation within the promoters. The runnables are also designed to be decoupled from the interface processing

Chapter 7

Conclusion

The thesis presents a novel method of a quadruplex structure classification, where the guanine residues in the PDB model are first assembled into correctly orientated tetrads, and then the chains are threaded through the tetrads which identifies the loop endpoints and their topology. The method is shown to be viable for a fully-automated classification. The algorithm also reveals previously misclassified models as well as the inconsistencies within the models, that were omitted with the manual analysis. The future improvements on this algorithm could make it applicable for the structures containing bulges in the stem, or detect multimerized G-wires.

The state of the art of the GQ-forming sequence recognition and the stability estimation, from the fundamental RegEx rules to cG/cC heuristic scoring based on the sequence composition and the energy-based models of the ViennaRNA tools is discussed. While no novel approach is presented, the chapter introduces the problem of the complexity of the folding rules. It appears that the driving forces behind the RNA GQs are different from the DNA GQs, which are much more complex to understand and predict.

While the effect of the environment and the nature of the binding cation is acknowledged, a possible relationship between the GQ-forming nucleotide sequence and the overall topology is explored. A predictor using the configuration of the loop length in the GQ is defined based on the study proposing a correlation between the loop length and the topology using the randomized sequences. The predictor usefulness is mitigated by high variability of the data set. The novel predictor attempts to overcome this problem, leveraging the observation that certain loop types have stronger affinity for certain loop lengths, and the fact that longer loops have more complex interactions. The predictor takes the length of the opening loop into account, and replaces the subsequent lengths with simple comparisons, where the next loop length is represented by either increase, decrease or a constant level. This broadens the application of the predictor to various loop lengths with enough accuracy. The predictors are compared to the sequence composition-based predictor and show a promising performance. However, despite the recent growth of the GQs in the database, the number of structures still do not present a statistically relevant sample, and there is a significant disparity of abundance in the known GQ families, which leads to a possible problem of overfitting. This problem is acknowledged, and the performance of all predictors is evaluated with a variable training set size.

The software tools including the library for both classification and prediction methods discussed in the thesis were implemented and used for evaluation of the proposed methods. In addition a web interface built on top of the library is introduced for an easier integration with the existing sequence analysis workflow, with a support for either input or output of the data in an interchangeable format. The future improvements of the user interface may incorporate a map/reduce parallelization model, and a visual annotation of the results on the genome.

Bibliography

Jean-Denis Beaudoin, Rachel Jodoin, and Jean-Pierre Perreault. New scoring system to identify rna g-quadruplex folding. *Nucleic acids research*, 42(2):1209–1223, 2014.

Anthony Bugaut and Shankar Balasubramanian. A sequence-independent study of the influence of short loop lengths on the stability and topology of intramolecular dna g-quadruplexes. *Biochemistry*, 47(2):689–697, 2008.

Sarah Burge, Gary N Parkinson, Pascale Hazel, Alan K Todd, and Stephen Neidle. Quadruplex dna: sequence, topology and structure. *Nucleic acids research*, 34(19):5402–5415, 2006.

Xiaohui Cang. *Molecular dynamics simulations on G-quadruplexes*. PhD thesis, The University of Utah, 2010.

Xiaohui Cang, Jiri Sponer, and Thomas E Cheatham, III. Insight into g-dna structural polymorphism and folding from sequence and loop connectivity through free energy analysis. *Journal of the American Chemical Society*, 133(36):14270–14279, 2011.

Junpeng Deng, Yong Xiong, and Muttaiya Sundaralingam. X-ray analysis of an rna tetraplex (uggggu) 4 with divalent sr2+ ions at subatomic resolution (0.61 å). *Proceedings of the National Academy of Sciences*, 98(24):13665–13670, 2001.

Johanna Eddy and Nancy Maizels. Conserved elements with potential to form polymorphic g-quadruplex structures in the first intron of human genes. *Nucleic acids research*, 36(4):1321–1333, 2008.

MA El Hassan and CR Calladine. The assessment of the geometry of dinucleotide steps in double-helical dna; a new local calculation scheme. *Journal of molecular biology*, 251(5):648–664, 1995.

Federico Fogolari, Haritha Haridas, Alessandra Corazza, Paolo Viglino, Davide Corà, Michele Caselle, Gennaro Esposito, and Luigi E Xodo. Molecular models for intrastrand dna g-quadruplexes. *BMC structural biology*, 9(1):64, 2009.

Aurore Guédin, Julien Gros, Patrizia Alberti, and Jean-Louis Mergny. How long is too long? effects of loop size on g-quadruplex stability. *Nucleic acids research*, 38(21):7858–7868, 2010.

Julian L Huppert and Shankar Balasubramanian. G-quadruplexes in promoters throughout the human genome. *Nucleic acids research*, 35(2):406–413, 2007.

Andreas Ioannis Karsisiotis, Christopher O'Kane, and Mateus Webba da Silva. Dna quadruplex folding formalism–a tutorial on quadruplex topologies. *Methods*, 64(1):28–35, 2013.

Oleg Kikin, Lawrence D'Antonio, and Paramjeet S Bagga. Qgrs mapper: a web-based server for predicting g-quadruplexes in nucleotide sequences. *Nucleic acids research*, 34(suppl 2):W676–W682, 2006.

Vitaly Kuryavyi, Anh Tuân Phan, and Dinshaw J Patel. Solution structures of all parallel-stranded monomeric and dimeric g-quadruplex scaffolds of the human c-kit2 promoter. *Nucleic acids research*, 38(19):6757–6773, 2010.

Ronny Lorenz, Stephan H Bernhart, Fabian Externbrink, Jing Qin, Christian Höner zu Siederdissen, Fabian Amman, Ivo L Hofacker, and Peter F Stadler. Rna folding algorithms with g-quadruplexes. In *Advances in Bioinformatics and Computational Biology*, pages 49–60. Springer, 2012.

Ronny Lorenz, Stephan H. Bernhart, Jing Qin, Christian Honer Zu Siederdissen, Andrea Tanzer, Fabian Amman, Ivo L. Hofacker, and Peter F. Stadler. 2d meets 4g: G-quadruplexes in rna secondary structure prediction. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, 10(4):832–844, 2013. ISSN 1545-5963. doi: http://doi.ieeecomputersociety.org/10.1109/TCBB.2013.7.

Maja Marušič, Rakesh N Veedu, Jesper Wengel, and Janez Plavec. G-rich vegf aptamer with locked and unlocked nucleic acid modifications exhibits a unique g-quadruplex fold. *Nucleic acids research*, 41(20):9524–9536, 2013.

Akimasa Matsugami, Yan Xu, Yuuki Noguchi, Hiroshi Sugiyama, and Masato Katahira. Structure of a human telomeric dna sequence stabilized by 8-bromoguanosine substitutions, as determined by nmr in a k+ solution. *FEBS Journal*, 274(14):3545–3556, 2007.

Vineeth Thachappilly Mukundan and Anh Tuân Phan. Bulges in g-quadruplexes: broadening the definition of g-quadruplex-forming sequences. *Journal of the American Chemical Society*, 135(13): 5017–5028, 2013.

Stephen Neidle. *Therapeutic applications of quadruplex nucleic acids*. Academic Press, 2011.

RV Reshetnikov, AM Kopylov, and AV Golovin. Classification of g-quadruplex dna on the basis of the quadruplex twist angle and planarity of g-quartets. *Acta naturae*, 2(4):72, 2010.

Paul Ryvkin, Steve G Hershman, Li-San Wang, and F Brad Johnson. Computational approaches to the detection and analysis of sequences with intramolecular g-quadruplex forming potential. In *G-Quadruplex DNA*, pages 39–50. Springer, 2010.

Anna Varizhuk, Dmitry Ischenko, Igor Smirnov, Olga Tatarinova, Vyacheslav Severov, Roman Novikov, Vladimir Tsvetkov, Vladimir Naumov, Dmitry Kaluzhny, and Galina Pozmogova. An improved search algorithm to find g-quadruplexes in genome sequences. *bioRxiv*, 2014.

Mateus Webba da Silva. Geometric formalism for dna quadruplex folding. *Chemistry-A European Journal*, 13(35):9738–9745, 2007.

Han Min Wong, Oliver Stegle, Simon Rodgers, and Julian Leon Huppert. A toolbox for predicting g-quadruplex formation and stability. *Journal of nucleic acids*, 2010, 2010.

Appendix A

Appendix

A.1 How to run command line tools

The command line tools and the library are in the `quadclass` directory.

A.1.1 Prerequisites

The library requires Python and addditionally a BioPython library for the FASTA/PDB reading and writing:

```
$ pip install biopython
$ pip install sklearn
```

Each of the utilities supports the `-h` or `--help` parameter with an example usage and a short overview of what the tool does.

A.2 Evaluating the performance of the predictors

The training and testing sets can be found in the `test` subdirectory. Both quadlearn and seqlearn tools support a performance evaluation and custom data set splits. There are test data sets included, which are derived from the quadclass analysis of all available structures and randomized.

Structure classification The quadlearn tool supports a `-t` parameter for a training set.

```
# 80-20 split
$ ./quadlearn.py -g -t train80.tsv test20.tsv
# 20-80 split
$ ./quadlearn.py -g -t test20.tsv train80.tsv
```

Structure prediction The `-g` parameter shows the ROC curves, which is also saved in the `seqlearn-roc.pdf` file.

```
# 80-20 split
$ ./seqlearn.py -g -t train80.tsv -v test20.tsv
# 20-80 split
$ ./seqlearn.py -g -t test20.tsv -v train80.tsv
```

A.3 How to run web interface

A.3.1 Prerequisites

The web interface requires Flask, Jinja2, Markdown and BioPython. The libraries may be installed either using the package management or using PIP:

```
$ pip install Flask
$ pip install markdown
$ pip install biopython
```

Alternatively, the libraries may be installed in the virtual environment as recommended by the Flask installation manual at `flask.pocoo.org/docs/installation`

A.3.2 Running the web interface

The interface in the `web` directory may either be run as a standalone application or as a container. As a standalone, it bind to the `0.0.0.0` host address and a port `5000`, but this can be changed along with the secret for sessions in the `seqalpha.py` file configuration.

The application can be then run in the standalone mode:

```
$ python ./seqalpha.py
```

Executing runnables from CLI The runnables may also be run from the command line as follows:

```
$ python runner.py <runnable> <parameters>
$ python runner.py cgscore 5UTRaspic_small.fasta 10
>5HSAA000316
agaagggagtgaagataaga
```