École polytechnique fédérale de Lausanne

Semester project

Master in Applied Mathematics

Computations with the cgDNA+ coarse-grain model of DNA

Author:
Federica Padovano

Supervisor:
John H. Maddocks
Raushan Singh

Contents

1 Introduction

DNA is essential to life, in our growth, reproduction, and health. It contains the instructions necessary for the cells to produce proteins that affect many different processes and functions in our bodies. Because DNA is so important, its damage or mutation can sometimes contribute to diseases development. For these reasons there are many research projects studying DNA in all its shapes. In particular, sequence dependent mechanics of DNA research is becoming more and more important in biology. Many mechanistic models have been created to represent DNA structure. For example, researchers have done copious modelling efforts in order to interpret the experimental data of some circular shaped DNA oligomers:

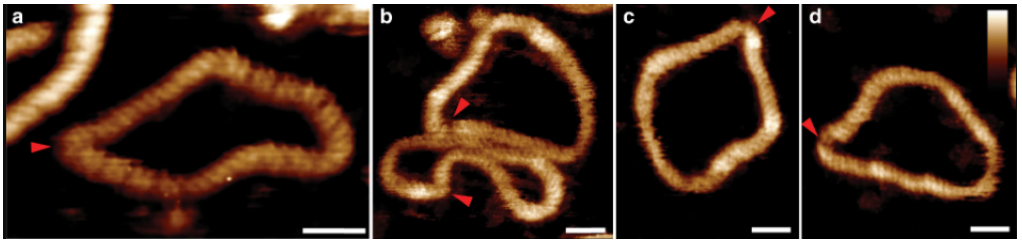


Figure 1: High resolution images (AFM) of DNA minicircles taken from [13]. Picture *a*) is the Noy 251bp sequence. Pictures $b - d$) are Noy 339bp sequence.

These rings, called *"minicircles"*, are an important case of study in the DNA modelling and are an experimental technique adopted to prove sequence dependent mechanics. In *Figure 1* we can observe high resolution pictures of the circles [13]. We are going to describe later in the report the two sequences in the image above. Many models have been created in the last years, but we will focus on the cgDNA+min one, which M. Beaud discuss in his master thesis [1]. In particular, the cgDNA+min algorithm is composed of different steps, of which we will talk later in the report, and the aim of this study is to remove one of those steps in order to speed the algorithm up and make it more accessible for everyone to model DNA minicircles.

2 DNA models

2.1 dsDNA

Deoxyribonucleic acid (abbreviated DNA) is the molecule that carries genetic information for the development and functioning of an organism. Double stranded DNA (dsDNA) is composed of two strands (Watson and Crick) connected by hydrogen bonds, each of which is composed of a long chain of monomer nucleotides. The nucleotides of DNA consist of a deoxyribose sugar molecule to which is attached a phosphate group and one of four nitrogenous bases: two purines (adenine and guanine) and two pyrimidines (cytosine and thymine). The DNA double helix is anti-parallel, which means that the 5' end of one strand is paired with the 3' end of its complementary strand (and vice versa). The conventional reading direction for a single strand is referred to as the $5' \rightarrow 3'$ direction.

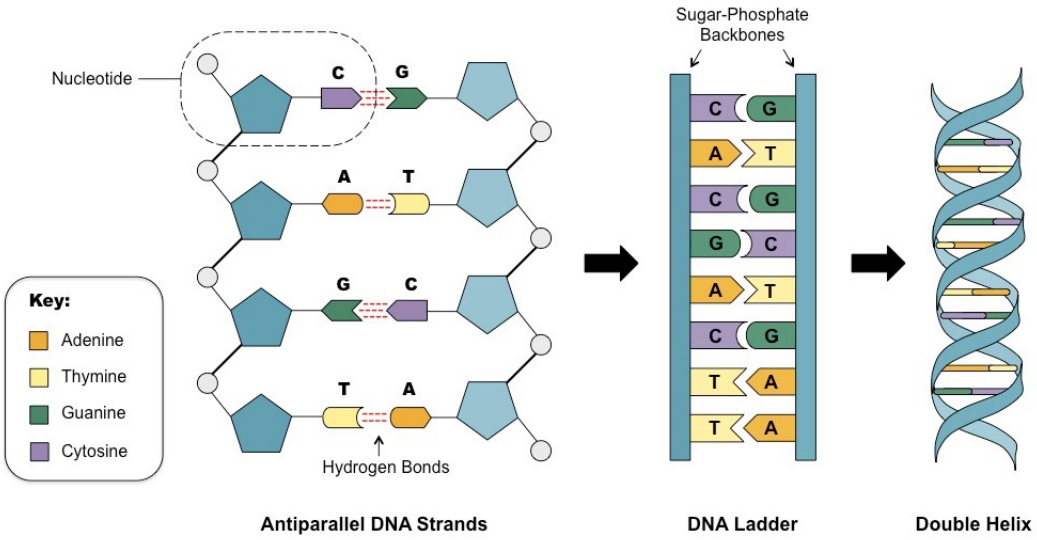


Figure 2: dsDNA structure from [2]

2.2 bBDNA

The birod model is a model developed for DNA oligomers, in particular it is a continuum model. It is composed of two functions $(g, \mathcal{P})$:

- $g(s) = (R(s), r(s)) \in SE(3)$, $s \in (0, L]$, called the *macrostructure*, is a continuum rod configuration that represents the average of the two strands. $R(s)$ is the rotation matrix representing the orientation of the cross section, while $r(s)$ is the position of the rod center-line;
- $\mathcal{P}(s)$, called the *microstructure*, represents the relative rotation and translation and allows the reconstruction of the two strands g^+ and g^-.

The energy associated with the birod DNA is assumed to have dependencies both on the local interactions (between neighboring nucleotides) and on local sequence (the nitrogenous bases). An equilibrium configuration for a rod is a configuration where the total couple and force densities acting on each cross section balance. In particular, the equilibrium condition for a double rod model is obtained by considering each strand as a single continuum rod in an external field and requiring that both equilibrium conditions hold. In his thesis [9], Głowacki proposed the bBDNA software which computes the equilibrium configuration for a specific sequence. The application of this software will be discussed later in the report for the minicircle model.

2.3 cgDNA+

The cgDNA+ model is a rigid-base coarse-grain DNA model, and it differs from the bBDNA one because it is not continuum. Given a sequence S and a parameter set $\mathcal{P}$, this model predicts the relative position and orientation of the bases in the form of a probability density function (pdf), $\rho(w; S, \mathcal{P})$, where w is the DNA coordinates. Three main assumptions are made:

1. each base and phosphate is represented by a rigid body. The interactions inside the molecule are then simplified to only consider the contributions of the rigid bases;

2. a natural choice for this pdf is a Gaussian distribution. The probability density depends on the ground-state μ and the stiffness matrix K of the molecule. The ground-state is then the most likely shape of the molecule under no external constraints or loadings, and the stiffness represents the resistance to change. In particular:
$$\rho(w; S, \mathcal{P}) = \frac{1}{Z} \exp\Big(-\frac{\beta}{2}(w - \mu(S,\mathcal{P}))K(w - \mu(S,\mathcal{P}))\Big) \tag{1}$$
where Z is the normalization constant, β is the inverse temperature energy scalar, and
$$U = -\frac{1}{2}(w - \mu(S,\mathcal{P}))K(w - \mu(S,\mathcal{P}))$$
is the shifted quadratic cgDNA energy of configuration w.

3. the molecule energy is assumed to have *double local dependence*. We only consider local contributions of the energy which means that a nucleotide is assumed to only interact with its nearest neighbours. We also assume that the energy depends locally on the dimer sequence, so the blocks in the parameter set depend only on dimer sequences. It follows that the stiffness matrix $K(S,\mathcal{P})$ is a banded matrix with overlapping blocks along the diagonal. The blocks represent all the interactions between two consecutive nucleotide pairs and the overlaps represent the fact that each base pair is interacting with both the previous and the following base pairs. The interactions between non-neighbouring base pairs are assumed to vanish. Hence the banded diagonal structure.

2.3.1 Rigid bodies

A N base pair sequence has N bases in each strand. Each base i is represented as a frame $X_i \in SE(3)$, $\forall i = 1, ..., N$, and $\bar{X}_i \in SE(3)$ is its Crick-Watson complement, i.e. $\bar{A} = T$, $\bar{G} = C$, $\bar{T} = A$, $\bar{C} = G$.

$$X_i = \begin{pmatrix} R_i & r_i \\ 0 & 1 \end{pmatrix} \in SE(3) \tag{2}$$

where $r_i \in \mathbb{R}^3$ and $R_i \in SO(3)$.

We first want to define the relationship between $X_i = [R_i^+, r_i^+]$ and its complement $\bar{X}_i = [R_i^-, r_i^-]$, which is described as:
$$\begin{pmatrix} R_i^+ & r_i^+ \\ 0 & 1 \end{pmatrix} = \begin{pmatrix} R_i^- & r_i^- \\ 0 & 1 \end{pmatrix} \begin{pmatrix} Q_i & Q_i^{1/2} q_i \\ 0 & 1 \end{pmatrix} \tag{3}$$

where $[Q_i, Q_i^{1/2} q_i]$ is the relative rigid body displacement from frame $\bar{X}_i$ to frame X_i. In particular $X_i = \bar{X}_i Z$

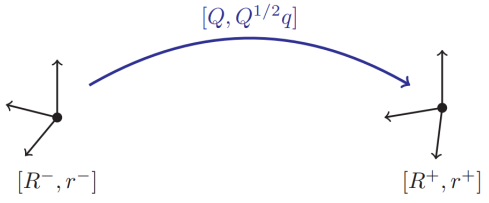


Figure 3: Relative rigid body displacement.

means

$$Z = \bar{X}_i^{-1} X_i = \begin{pmatrix} (R_i^-)^T R_i^+ & (R_i^-)^T (r_i^+ - r_i^-) \\ 0 & 1 \end{pmatrix}$$

therefore we can express $Q_i = (R_i^-)^T R_i^+$. Furthermore, $(R_i^-)^T (r_i^+ - r_i^-) = Q_i^{1/2} q_i$ where $q_i = (R_i^- Q_i^{1/2})^T (r_i^+ - r_i^-)$ is the coordinates $(r_i^+ - r_i^-)$ in a midway rotation frame between $\bar{X}_i$ and X_i, which is $R_i = R_i^- Q_i^{1/2}$.
This relationship is represented in w by the ***intra coordinates*** x_i. The intra coordinates for base pair i are then the Cayley vector of the relative rotation Q_i and the relative translation q_i expressed in the mid frame,

$$x_i = [Cay(Q_i), q_i] \in \mathbb{R}^6 \tag{4}$$

We now have to consider the interaction between two consecutive base pairs $(X_i, \bar{X}_i)$ and $(X_{i+1}, \bar{X}_{i+1})$. Between two base frames there is a midway frame called base pair frame:

$$Y_i = \begin{pmatrix} R_i & r_i \\ 0 & 1 \end{pmatrix} \in SE(3), \qquad R_i = R_i^- Q_i^{1/2} \in SO(3) \quad \text{and} \quad r_i = \frac{1}{2}(r_i^+ + r_i^-) \in \mathbb{R}^3 \tag{5}$$

We want to describe the relationship between two consecutive base pair frames Y_i and Y_{i+1}. This is done as for the intras coordinates defining the relative displacement between the two frames:

$$\begin{pmatrix} G_i & G_i^{1/2} g \\ 0 & 1 \end{pmatrix} = \begin{pmatrix} R_i^T R_{i+1} & R_i^T (r_{i+1} - r_i) \\ 0 & 1 \end{pmatrix} \tag{6}$$

and the mid-way frame, also called *"Junction frame"*:

$$K = \begin{pmatrix} J_i & j_i \\ 0 & 1 \end{pmatrix} = \begin{pmatrix} R_i G_i^{1/2} & \frac{1}{2}(r_{i+1} + r_i) \\ 0 & 1 \end{pmatrix}$$

The matrix G_i represents the relative rotation between R_i and R_{i+1}, and g represents the coordinates of $(r_i - r_{i+1})$ expressed in the midway rotation frame K. Additionally, we define the ***inter coordinates*** the ones describing the relationship between consecutive base pairs. The inter coordinates for junction i between base pair i and $i+1$ are then the Cayley vector of the relative rotation G_i and the relative translation g_i expressed in the mid frame,

$$y_i = [Cay(G_i), g_i] \in \mathbb{R}^6 \tag{7}$$

Each phosphate group is also represented by a frame $[p_i^\pm, P_i^\pm] \in SE(3)$, Crick $(-)$ or Watson $(+)$. The coordinates of each phosphate are a $SE(3)$ displacement from the base to the associated phosphate, expressed in the base frame:

$$\begin{pmatrix} R_i^\pm & r_i^\pm \\ 0 & 1 \end{pmatrix} \begin{pmatrix} M_i^\pm & m_i^\pm \\ 0 & 1 \end{pmatrix} = \begin{pmatrix} P_i^\pm & p_i^\pm \\ 0 & 1 \end{pmatrix} \tag{8}$$

Therefore, the relationship between the phosphate group and the base pair are:

$$x_i^c = [Cay(M_i^-), m_i^-] \in \mathbb{R}^6 \qquad x_i^w = [Cay(M_i^+), m_i^+] \in \mathbb{R}^6 \tag{9}$$

Each phosphate group is associated to a base pair, however phosphates are not exactly aligned with the bases because they are located in between two base pairs. This induces a problem at the ends, in fact we cannot model accurately the position of a phosphate group that is not lying in between two bases using a Gaussian probability density function. Therefore we remove the extra phosphate groups at the end bases only to keep the phosphates that are inside junctions of the sequence.

As a consequence of these definitions, given a sequence $S = X_1 X_2 ... X_n$, we can define the cgDNA+ coordinates w as:

$$w = (x_1, x_1^c, y_1, x_2^w, x_2, x_2^c, y_2, ..., x_i^w, x_i, x_i^c, y_i, ..., y_{n-1}, x_n^w, x_n) \in \mathbb{R}^{24n-18} \tag{10}$$

which describes the structure of a dsDNA molecule.

2.3.2 Energy

There are three assumptions made on the energy:

1. has a shifted quadratic form;
2. the total energy is a sum over level junctions energies;
3. the coefficient in the local junction energy depends on the local dimer sequence step.

The second assumption implies $U = \sum_{i=1}^n U_i$, where U_i is the local energy at base pair i. The first and the third ones imply

$$U_i = \frac{1}{2}(w_i - \hat{w}_i^{XY}) \cdot \hat{K}_i^{XY} (w_i - \hat{w}_i^{XY})$$

In particular $w_i = (x_i^w, x_i, x_i^c, y_i, x_{i+1}^w, x_{i+1}, x_{i+1}^c) \in \mathbb{R}^{42}$; $\hat{w}_i^{XY} \in \mathbb{R}^{42}$ and $\hat{K}_i^{XY} \in \mathbb{R}^{42\times 42}$ are respectively the local ground-state and stiffness matrix, where i represents the locality and XY the dependence on the sequence, $\forall i = 2, ..., n-1$. When $i = 1$ or $i = n$, w_i will be in $\mathbb{R}^{36}$ because of the absence of the first and last phosphate

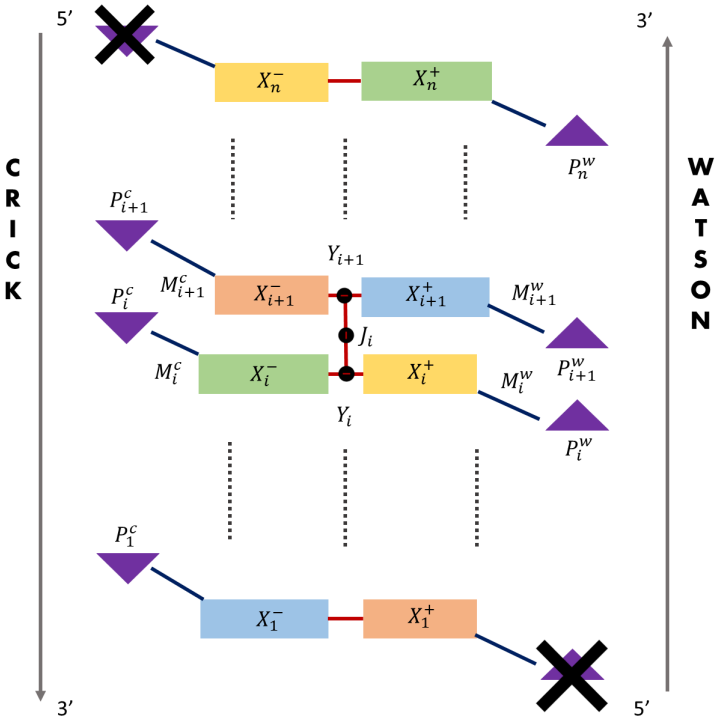
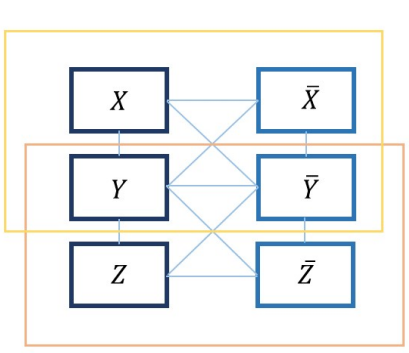


Figure 4: Rigid body structure for cgDNA+ model.

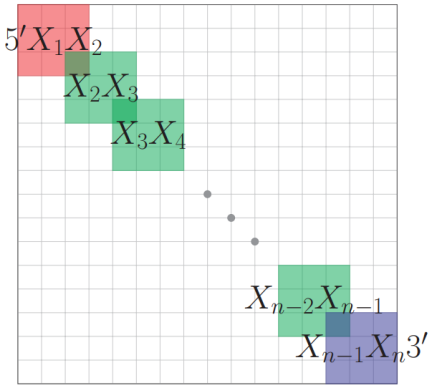
group, and as a consequence $\hat{w}_i^{XY} \in \mathbb{R}^{36}$ and $\hat{K}_i^{XY}\mathbb{R}^{36\times36}$. The free energy U is a sum of shifted quadratic energies so it has as well a shifted quadratic form:

$$U(w;S,\mathcal{P}) = \frac{1}{2}(w-\mu)\cdot K(w-\mu) + c \tag{11}$$

The constant c plays no role and is therefore not taken into account; while $w \in \mathbb{R}^{24n-18}$, $\mu \in \mathbb{R}^{24n-18}$ and $K \in \mathbb{R}^{24n-18\times24n-18}$. The stiffness matrix K is a banded matrix with $(n-1)$ 42×42 blocks (except for the first and last one that are 36×36) with 18×18 overlaps. The aim is to find the coordinates that are more likely to be found in real life, it then means the vector w with the highest probability and as a consequence with the lowest free energy. If there are no constraints, $w=\mu$ is the point with the lowest energy and the highest probability.



(a) Interactions among the neighbours.



(b) Stiffness matrix construction.

Figure 5: Energy dependencies.

2.3.3 Parameter set

cgDNA+ uses a pre-computed parameter set which is obtained by:

1. choosing a training library of sequences S_j, with $j = 1, ..., M$;
2. running a full atomistic Molecular Dynamics [10] simulations for each training library oligomer and obtain $\{w_j^k\}_{k=1}^N$, for each $j = 1, ..., M$;
3. computing some statistics, as the sample mean and the sample covariance, to make the model biofitting the observed data.

Since the *pdf* depends on the parameter set, it is used to reconstruct the ground-state and stiffness matrix corresponding to the given sequence S. The parameter set has been changed different times in order to improve it. In particular, there are sixteen (4×4) possible couples XY, given the bases $\{A, G, T, C\}$, and it contains:

- K^{XY}, 42×42 symmetric positive definite blocks and σ^{XY}, 42×1 vector. In this case X and Y are in the interior of the sequence;
- $K^{5'XY}$, 36×36 symmetric positive definite blocks and $\sigma^{5'XY}$, 36×1 vector. In this case X is the 5'end of the sequence;
- $K^{XY3'}$, 36×36 symmetric positive definite blocks and $\sigma^{XY3'}$, 36×1 vector. In this case Y is the 3' end of the sequence;

Parameter block symmetry comes from the fact that we model a quadratic energy. The energy, the stiffness and the ground-state have to be independent of the choice of reading strand.

For the reconstruction of the stiffness matrix and the ground-state, we first read the sequence S and build the banded stiffness matrix by positioning the 42×42 blocks (the first and the last one have dimension 36×36 because of the missing phosphates), there will be a 18×18 overlap between two consecutive blocks. The shape vector $\sigma = K\mu$ has a local dependence on the sequence, while μ doesn't. For this reason it is necessary to construct σ first, with the parameter set, and then compute μ. Each block of σ is composed of the vector $\sigma^{X_iX_{i+1}}$ and the contributions of the previous and next dimers for the 18×1 overlap. The ground-state is then obtained using the relation $\mu = K^{-1}\sigma$.

2.3.4 Periodicity

In nature we observe that some DNA sequences are composed by a smaller one, called *tandem repeat*, that repeats itself an N number of times, for example $Poly(\alpha\beta)_N$ in which the dimer $\alpha\beta$ is repeated N times. In order to reduce the computational cost of the algorithm, Glowacki and Grandchamp ([9],[8]) decided to take advantage of this periodicity and proposed a new method in which they construct the ground state and stiffness matrix only for the tandem sequence. This allows to create infinite periodic sequences using only one period of the base sequence. In order to achieve it the ground state and the stiffness matrix have to be constructed in a different way, in particular the main difference with respect to the old ones lies on the extremities.

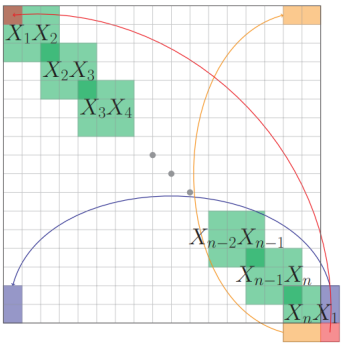


Figure 6: Periodic stiffness matrix.

To construct the periodic stiffness matrix we need to consider the relationship between the last and the first base pairs which were not considered before. For a sequence of length n, K_p is of size $24n \times 24n$ with overlapping 42×42 blocks and 18×18 overlaps. The last block is truncated to be 24×24. The extra entries are added to the first 18×18 block, the 18×24 upper right and 24×18 bottom left corners. The extra blocks in the anti-diagonal corners represent the interactions between the last and first base pairs and are taken from the $K^{X_nX_1}$ parameter block. In *Figure 6.* it is shown how it is constructed.

2.4 cgDNA+min

In order to construct the minicircle configuration we will use the cgDNA+ and its periodicity. In fact, in the special case of minicircles, the *tandem sequence* is the entire sequence, and we consider as it repeats itself infinite times but enforcing a cyclized shape, so a closure of the extremities. The base coordinates are then periodic when repeating the sequence.

2.4.1 Continuous and discrete model

In order to find a first minicircle configuration, we first use the bBDNA software with the given sequence and the chosen periodic parameters. This software finds minicircles equilibrium continuum configurations (birod model) and generate a bifurcation diagram which represents different stable points. The highest points in the bifurcation diagram have higher energy. It finally chooses a continuum energy equilibrium configuration selecting the configuration that is correctly closed to be as close as possible to a valid discrete configuration.

After obtaining the continuum energy equilibrium configuration it constructs its discretization configuration. In particular, it adds the phosphate relative coordinates of the cgDNA+ periodic configuration of the specific sequence because they are not considered in the birod model. After the discretization, it needs a discrete energy minimization step. Indeed, the assumptions for the continuum model are slightly different than the ones of the discrete case. Moreover, during the discretization process, errors are induced. Therefore, it applies a cgDNA+ minicircle energy minimization procedure to ensure that the final configuration is a proper minimizer for the discrete energy.

2.4.2 Coordinate vector

In [12] Manning proposed to change representation and use absolute coordinates for each base pair frame because inter variables lead to highly non-linear and non-local closure constrainst. He used quaternions to represent the absolute rotation of each base pair frame. The closure assumption then becomes a local condition, the last base pair must be close to the first one. After the discretization, for each base pair i we have intra and phosphate coordinates $x_i, x_i^w, x_i^c \in \mathbb{R}^6$, and base pair coordinates $(o_i, q_i) \in \mathbb{R}^7$, where $o \in \mathbb{R}^3$ is the absolute translation between base pair frame i and the origin and $q \in \mathbb{R}^4$ is the absolute rotation of the base pair frame expressed with a quaternion. So the vector of coordinates z of a sequence of length n becomes:

$$z = (x_1^w, x_1, x_1^c, o_1, q_1, x_2^w, x_2, x_2^c, o_2, q_2, \ldots, x_{n-1}, o_{n-1}, q_{n-1}, x_n^w, x_n, x_n^c, o_n, q_n) \in \mathbb{R}^{25n}, \tag{12}$$

He also fixed the first base pair as the reference point, hence $o_1 = (0, 0, 0)$ and $q_1 = (0, 0, 0, 1)$.

2.4.3 Energy

We now have to describe the energy configuration with this new vector of coordinates z, given $F(z) = w$:

$$U(w) = U(F(z)) = \frac{1}{2}(F(z) - \mu)^T K(F(z) - \mu),$$

We need to minimize this function $U^*(z)$ with the constraint $\|q_i\|^2 = 1 \; \forall i = 1 \ldots, n$. So we write the Lagrangian function $\mathcal{L}$ for U and its constraint:

$$\mathcal{L}(z) = \frac{1}{2}(F(z) - \mu)^T K(F(z) - \mu) + \lambda \sum_{i=1}^{n} (\|q_i\|^2 - 1)^2. \tag{13}$$

where λ is a constant scalar penalty weight. In our case, we will consider $\lambda = 100$. We use *Matlab* function *fminunc* in order to minimize the energy. We also give the explicit expressions for the gradient and the Hessian matrix as inputs of the function, to speed the minimization algorithm.

2.4.4 Reconstruction of vector w

Once we obtain the optimum configuration for z, we need to reconstruct the vector w. In particular, quaternions are generally represented with the Euler parameter:

$$q = (a, b, c, d) \in \mathbb{R}^4$$

where a, b, c, and d are real numbers. Each rotation matrix can be described by a quaternion q.

$$R(q) = \frac{1}{a^2+b^2+c^2+d^2}\begin{bmatrix} a^2+b^2-c^2-d^2 & 2bc-2ad & 2ac+2bd \\ 2bc+2ad & a^2-b^2+c^2-d^2 & 2cd-2ab \\ 2bd-2ac & 2ab+2cd & a^2-b^2-c^2+d^2 \end{bmatrix}, \tag{14}$$

which is invariant with respect to the norm of q.
In particular, for a given normalized quaternion $\tilde{q}$, any arbitrary quaternion q can be written as:

$$q = k_1 B_1 \tilde{q} + k_2 B_2 \tilde{q} + k_3 B_3 \tilde{q} + \sqrt{1-k_1^2-k_2^2-k_3^2}\tilde{q}, \tag{15}$$

with

$$B_1 \equiv \begin{bmatrix} 0 & 0 & 0 & 1 \\ 0 & 0 & 1 & 0 \\ 0 & -1 & 0 & 0 \\ -1 & 0 & 0 & 0 \end{bmatrix}, B_2 \equiv \begin{bmatrix} 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \\ 0 & -1 & 0 & 0 \end{bmatrix}, B_3 \equiv \begin{bmatrix} 0 & 1 & 0 & 0 \\ -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & -1 & 0 \end{bmatrix}.$$

Therefore $\{\tilde{q}, B_1\tilde{q}, B_2\tilde{q}, B_3\tilde{q}\}$ is an orthonormal basis for $\mathbb{R}^4$. Morevoer, the inverse of a normalized quaternion $q = [q_1, q_2, q_3, q_4]^T$ is $q^{-1} = [-q_1, -q_2, -q_3, q_4]^T$ and its associated rotation matrix is $R(q^{-1}) = R(q)^T$. We also know that:

$$q_a^{-1} \circ q_b = \begin{bmatrix} q_b^T B_1 q_a \\ q_b^T B_2 q_a \\ q_b^T B_3 q_a \\ q_a^T q_b \end{bmatrix}. \tag{16}$$

and we can compute

$$q_i \circ \sqrt{q_i^{-1} \circ q_{i+1}} = \frac{q_i + q_{i+1}}{q_i + q_{i+1}} = \frac{q_i + q_{i+1}}{\sqrt{2 + 2q_i^T q_{i+1}}}. \tag{17}$$

We need to define the function $F(z) = w$ that reconstructs the standard inter coordinate vector y_i from (o_i, q_i) and (o_{i+1}, q_{i+1}).

$$y_i = F(o_i, q_i, o_{i+1}, q_{i+1}) = (\theta_i^1, \theta_i^2, \theta_i^3, \zeta_1^1, \zeta_i^2, \zeta_i^3), \tag{18}$$

with ζ the relative translation part and θ the relative rotation part of the coordinates. In [1], the relative rotation between two base pair frames is the rotation defined by $q_i^{-1} \circ q_{i+1}$, hence $R(q_i \circ \sqrt{q_i^{-1} \circ q_{i+1}})$. Therefore the relative translation expressed in the junction frame is:

$$\zeta_i^T = (o_{i+1} - o_i) R(q_i \circ \sqrt{q_i^{-1} \circ q_{i+1}}), \tag{19}$$

And θ_i is the Cayley vector of the rotation matrix. Therefore, once we found the vector z we can reconstruct w with the procedure above, and obtain the relative configuration of the DNA minicircle structure.

3 Sensitivity of cgDNA+ to computational parameters

The cgDNA+min model starts from the periodic configuration of the DNA sequence. It then creates a continuous birod model starting from this configuration, and use the bBDNA software to find equilibrium minicircle configurations. This equilibrium is then discretized to provide an initial configuration and finally a minimization procedure is done to find the discrete energy minimizer.

Since bBDNA software is not available for everyone and it takes time to understand its employment, we want to find a way to construct an initial minicircle configuration for the energy minimization step without the use of bBDNA software. Our first goal is to prove our hypothesis: *"Given the periodic ground-state of a specific sequence, if we change its inter variables, while intras and phosphate are left as they are, we can find a good initial guess for the energy optimization"*.

In order to see if this hypothesis holds we decided to consider two different initial guesses for the energy optimization:

1. configuration obtained with the bBDNA software (will be called *bBDNA*);
2. configuration in which the intra and phosphate coordinates are the same as the periodic ground-state, while the inter coordinates are the ones of the configuration obtained with the bBDNA software (will be called *bBDNA with gd-intras*).

We can notice that the second configuration is the one described in the hypothesis, while the first one is the one described by M.Beaud in his thesis. We tested the energy minimization with these two configurations as starting point, because we want to compare the two numerically optimal results and analyze if they converge to the same optimum. In fact, since the assumed correctness of the bBDNA method has already been demonstrated, if the two optimal results converge to the same one we can state that our hypothesis is correct.

The criteria for a good enough initial guess is that cgDNA+min converges to a configuration that passes out numerical criteria for being a real and believable equilibrium. In [1] it is stated the theorem that for at least two values of the linking number m there are at least two equilibria, one a local minimum and the other one a saddle point, but the local minimum is a more stable configuration than the saddle point. In general, there can be more than one local minimum or saddle point, indeed, the energy might be a periodic function and any periodic function has p critical points. If the configuration has really high twist it means that it is close to isotropic, i.e. the energy is nearly constant with respect to register (unless the centre line has an intrinsic curvature), it happens for example for $Poly_{\alpha\beta}$, in those cases it's hard to find minimal values. So we have three cases:

1. one local minimum and one saddle point;
2. p local minima and p saddle points;
3. energy close to constant;

but usually we are in the first case.

3.1 Trust Region and Quasi Newton methods

Trust Region method. Trust-region methods define a region around the current iterate x_k within which they trust the model to be an adequate representation of the objective function (f), and then choose the step to be the approximate minimizer of the model in this region. Given two tolerances ρ' and $\bar{\Delta}$, at each iterate it requires to solve a first sub-problem (model):

$$m_k(t) = f(x_k) + \langle \nabla f(x_k), t\rangle + \frac{1}{2}\langle t, H(x_k)t\rangle$$

find $t \leq \Delta_k$ such that t minimizes $m(t)$ and remains in the specific region centered in x_k of radius Δ_k. Then it sets $x_k^+ = x_k + t$ and defines the ratio between the *actual reduction* and the *predicted reduction*:

$$\rho_k = \frac{f(x_k) - f(x_k^+)}{m_k(0) - m_k(t)}$$

Then, with dependence on how close are the two reductions, it sets:

$$x_{k+1} = \begin{cases} x_k^+ & \rho_k > \rho' \\ x_k & \rho_k \leq \rho' \end{cases}$$

A failed step is an indication that our model is an inadequate representation of the objective function over the current trust region and in this case it reduces the size of the region and finds a new minimizer:

$$\Delta_{k+1} = \begin{cases} \min(2\Delta_k, \bar{\Delta}) & \rho_k > \frac{3}{4} \\ \Delta_k & otherwise \\ \frac{\Delta_k}{4} & \rho_k < \frac{1}{4} \end{cases}$$

The trust-region method is reliable and robust, and it has very strong convergence properties. In fact, it uses Newton's method with safe-guards so that it gains good global convergence properties, while preserving the quadratic local convergence rate of Newton method. However, since each step is constrained to a region centered in the current point, of which radius may or may not vary from iteration to iteration, this method can be very long. For this reason the matlab function *fminunc* allows to set a step-size tolerance which can stop the algorithm if it gets too slow. But if it stops because of reached step-size tolerance it doesn't mean that it found a local minimum or a stationary point.

Quasi-Newton method. Newton's method is an iterative method for finding the roots of a differentiable function F. In the optimization contest, given $\min_{x\in\mathbb{R}^n} f(x)$, this method is used to solve $\nabla f(x) = 0$ and find stationary points which are minimums. In particular using Taylor expansion we have that the direction that at each step minimizes the function the best, satisfies:

$$\begin{aligned} f(x_k + t) &\approx f(x_k) + \langle \nabla f(x_k), t\rangle + \frac{1}{2}\langle t, H(x_k)t\rangle \\ 0 &= \frac{d}{dt}\Big(f(x_k) + \langle \nabla f(x_k), t\rangle + \frac{1}{2}\langle t, H(x_k)t\rangle\Big) \\ 0 &= \nabla f(x_k) + H(x_k)t \\ t &= -[H(x_k)]^{-1}\nabla f(x_k) \end{aligned}$$

And as a consequence the iterative method is constructed as:

$$x_{k+1} = x_k - [H(x_k)]^{-1}\nabla f(x_k)$$

Finding the inverse of the Hessian in high dimensions to compute the Newton direction can be an expensive operation, for this reason some Quasi-Newton methods have been invented in order to simplify the Hessian computation. Calculating H numerically involves a large amount of computation and Quasi-Newton methods avoid this by using the observed behavior of $f(x)$ and $\nabla f(x)$ to build up curvature information to make an approximation to H using an appropriate updating technique, for example BFGS method (Broyden–Fletcher–Goldfarb–Shanno algorithm). This method is the one used in matlab function *fminunc*, and it uses a cubic line search procedure ([7]). Moreover, in a local neighborhood of the optimal solution, the iterates generated by BFGS converge to the optimal solution at a superlinear rate ([6]). However, it often fails if the current iterate does not have a positive definite Hessian matrix, because in this case the quadratic model is not convex. For these reasons, the advantages and disadvantage of Quasi-Newton method are:

- is computationally cheap;
- is computationally fast ;
- there is no need to evaluate the second derivative;
- the lack of precision in the Hessian calculation leads to slower convergence in terms of steps.

In our case we will also evaluate the condition number of the Hessian H, $\kappa(H)$, which measures how much the output value of a linear system is sensible to a small change in the input argument. In particular, it gives a bound on how inaccurate the solution x will be after approximation:

$$\begin{aligned} H(x + \delta x) &= \nabla f + \delta\nabla f \\ \frac{||\delta x||}{||x||} &\leq \kappa(H)\frac{||\delta\nabla f||}{||\nabla f||} \end{aligned}$$

If the condition number is large, the problem will be said to be *ill-conditioned*, and even a small error in ∇f may cause a large error in x.

3.2 Sequences

In particular, for the minimization we used both the *Trust region* and *Quasi-Newton* methods, depending on the sequence. Using these two optimization methods we managed to analyze each initial guess for Kahn-Crothers ([5]), Widom ([4]) and Pyne ([15]) sequences. For each sequence we will evaluate the absolute difference between the two optimal configurations obtained with the minimization, the energy, the eigenvalues and the Hessian condition number. It's necessary to check the Hessian matrix eigenvalues because we know that the equilibrium might be a local minimum or a saddle point. In fact, if:

- all the eigenvalues are positive, it is said to be a positive-definite matrix. In this point the function is "concave up" and we have a local minimum;
- all the eigenvalues are negative, it is said to be a negative-definite matrix. In this point the function is "concave down" and we have a local maximum;
- mix eigenvalues (one negative and the others positive), the matrix is indefinite. In this case, if the gradient is zero we have a saddle point.

The input parameters were the energy function, its gradient and hessian, and the starting point for the algorithm

3.2.1 Kahn-Crothers

We first analyze the result for the Kahn-Crothers sequence, which has 158 base pairs. In particular we used four initial guesses to start with which are the same used by M. Beaud in [1].

GUESS 1: we used *fminunc* Matlab function with Trust Region algorithm and tolerance for the first order optimality of $1.0e-6$. The absolute difference of the two configurations obtained with the two different initial guesses is of the order of 10^{-3}. In particular since the two Hessians don't have non-positive eigenvalues we can state that the optimum is a local minimum in both cases (Link number is $m = 15$).

	bBDNA	**bBDNA with gd-intras**
Energy	23.2548	23.2543
Norm of the gradient	0.0767	0.0778
Min Hessian Eigenvalue	4.8661e-06	4.8346e-06
Max Hessian Eigenvalue	6.6359e+04	6.7084e+04
Median Hessian Eigenvalue	11.5859	11.5858
Number of negative eigenvalues	0	0
Hessian Condition number	3.5723e+10	3.5927e+10

GUESS 2: we used *fminunc* Matlab function with Trust Region algorithm and tolerance for the first order optimality of $1.0e-6$. The absolute difference of the two configurations obtained with the two different initial guesses is of the order of 10^{-1}. If we low the first order optimality tolerance to $1.0e-8$, then the absolute difference becomes of the order of 10^{-2}. More in general if we decrease the tolerance in the algorithm then it gains more accurate optimal results. Since in our case decreasing the tolerance brings to lower absolute difference, it means that the more the results are accurate the more the two optimums get closer, and as a consequence they seem to converge to the same one. In particular since the two Hessians don't have non-positive eigenvalues we can state that the optimum is a local minimum in both cases (Link number is $m = 14$).

	bBDNA	**bBDNA with gd-intras**
Energy	56.1674	56.4639
Norm of the gradient	11.5856	11.5857
Min Hessian Eigenvalue	1.2581e-05	7.7524e-06
Max Hessian Eigenvalue	6.7400e+04	6.7375e+04
Median Hessian Eigenvalue	11.6828	11.5857
Number of negative eigenvalues	0	0
Hessian Condition number	1.5657e+10	2.6192e+10

GUESS 4: we used *fminunc* Matlab function with Trust Region algorithm and tolerance for the first order optimality of $1.0e-6$. The absolute difference of the two configurations obtained with the two different initial

guesses is of the order of 10^{-3}. In particular since the two Hessians don't have non-positive eigenvalues we can state that the optimum is a local minimum in both cases (Link number is $m = 16$).

	bBDNA	**bBDNA with gd-intras**
Energy	55.0552	55.0508
Norm of the gradient	0.0722	0.0663
Min Hessian Eigenvalue	1.1101e-05	1.1223e-05
Max Hessian Eigenvalue	7.8887e+04	6.2543e+04
Median Hessian Eigenvalue	11.4382	11.4355
Number of negative eigenvalues	0	0
Hessian Condition number	7.1063e+09	5.5724e+09

GUESS 10: we used *fminunc* Matlab function with Trust Region algorithm and tolerance for the first order optimality of $1.0e-6$. The absolute difference of the two configurations obtained with the two different initial guesses is of the order of 10^{-3}. In particular since the two Hessians don't have non-positive eigenvalues we can state that the optimum is a local minimum in both cases (Link number is $m = 15$).

	bBDNA	**bBDNA with gd-intras**
Energy	23.3988	23.4270
Norm of the gradient	0.0562	0.1169
Min Hessian Eigenvalue	3.8948e-06	4.7605e-06
Max Hessian Eigenvalue	6.6367e+04	6.6369e+04
Median Hessian Eigenvalue	11.6873	11.6875
Number of negative eigenvalues	0	0
Hessian Condition number	4.7163e+10	3.8725e+10

3.2.2 Widom 601

We analyze the results for the Widom sequence, which has 94 base pairs. In particular we used four initial guesses to start with which are the same used by M. Beaud in [1].

GUESS 1: we used *fminunc* Matlab function with Trust Region algorithm and tolerance for the first order optimality of $1.0e-6$. The absolute difference of the two configurations obtained with the two different initial guesses is of the order of 10^{-2}. In particular, the bDBNA optimum Hessians have one negative eigenvalue, and the bBDNA with gd-intras optimum has all positive eigenvalues. Hence, the first one seems to be a saddle point, while the second one a local minimum (Link number is $m = 9$).

	bBDNA	**bBDNA with gd-intras**
Energy	38.5508	38.6636
Norm of the gradient	0.7167	0.6347
Min Hessian Eigenvalue	-1.3285e-05	3.4011e-05
Max Hessian Eigenvalue	6.4146e+04	6.4122e+04
Median Hessian Eigenvalue	11.6359	11.6346
Number of negative eigenvalues	1	0
Hessian Condition number	1.2426e+10	4.8016e+09

GUESS 2: we first tried using *fminunc* Matlab function using Trust Region algorithm, however, after 52 hours of running it didn't get to and end, so we decided to use a different method. In particular we used *fminunc* Matlab function with Quasi-Newton algorithm and tolerance for the first order optimality of $1.0e-8$. The absolute difference of the two configurations obtained with the two different initial guesses is of the order of 10^{-3}. Since the obtained solutions have still a significant gradient value we decided to run the *fminunc* Matlab function again with Trust Region algorithm and first order optimality of $1.0e-6$, but having as starting point the solutions obtained with the first Quasi-Newton run. Doing that we obtained an absolute difference of the two configurations with the two different initial guesses of the order of 10^{-4} and the configurations are the same M. Beaud found in [1]. Moreover, since the two Hessians have two negative eigenvalues and the evaluated gradients are quite big, the optimal points seem not to be local minima (Link number is $m = 8$).

	bBDNA	bBDNA with gd-intras
Energy	124.6879	124.6760
Norm of gradient	0.2698	0.2954
Min Hessian Eigenvalue	7.1000e-06	2.3600e-05
Max Hessian Eigenvalue	6.2960e+04	6.2959e+04
Median Hessian Eigenvalue	11.5905	11.5906
Number of negative eigenvalues	0	0
Hessian Condition number	2.9683e+10	8.4643e+09

GUESS 3: we used *fminunc* Matlab function with Trust Region algorithm and tolerance for the first order optimality of $1.0e-6$. The absolute difference of the two configurations obtained with the two different initial guesses is of the order of 10^{-2}. In particular since the two Hessians have only one negative eigenvalue we can state that the optimum seems to be a saddle point in both cases (Link number is $m = 9$).

	bBDNA	bBDNA with gd-intras
Energy	47.3628	47.6631
Norm of the gradient	0.2870	0.0535
Min Hessian Eigenvalue	-8.6882e-06	-1.4128e-06
Max Hessian Eigenvalue	6.4387e+04	6.4394e+04
Median Hessian Eigenvalue	11.6271	11.6276
Number of negative eigenvalues	1	1
Hessian Condition number	2.2151e+10	3.8152e+10

GUESS 4: we run both the Trust Region and the Quasi-Newton algorithms, with tolerance for the first order optimality of 1.0e-6. We have adopted:

- Trust Region with first order optimality of $1.0e-6$: we found two different optimal results (absolute difference greater than $1.0e+00$) which are shown in *Figure 7*. In particular the two solutions are different because with the bBDNA initial guess the algorithm stops because it found a local minimum, while with the other initial guess it doesn't stop because it found a local minimum (gradient value is still significant) but because *step size tolerance*. This last *exit* appears when the algorithm is going too slow.
- Quasi-Newton with first order optimality of $1.0e-8$ and Trust Region with first order optimality of $1.0e-6$ and starting point the one obtained with Quasi-Newton: we found the same optimal result (absolute difference of the order of $1.0e-4$). In particular the obtained configurations are the same M. Beaud found in [1].

With the Trust Region method the obtained results are:

	bBDNA	bBDNA with gd-intras
Energy	106.7857	139.6462
Norm of the gradient	0.4332	10.4564
Min Hessian Eigenvalue	6.4245e-05	-4.9079e-05
Links	10	8
Max Hessian Eigenvalue	6.7709e+04	6.7768e+04
Median Hessian Eigenvalue	11.5856	11.5857
Number of negative eigenvalues	0	1
Hessian Condition number	8.4430e+10	8.8909e+11

With the Quasi Newton method+ Trust Region the obtained results are:

	bBDNA	bBDNA with gd-intras
Energy	107.1669	106.9373
Norm of the gradient	0.4399	0.4376
Min Hessian Eigenvalue	8.9821e-05	6.3080e-05
Links	10	10
Max Hessian Eigenvalue	6.5538e+04	6.5598e+04
Median Hessian Eigenvalue	11.5856	11.5857
Number of negative eigenvalues	0	0
Hessian Condition number	1.9035e+09	2.7651e+09

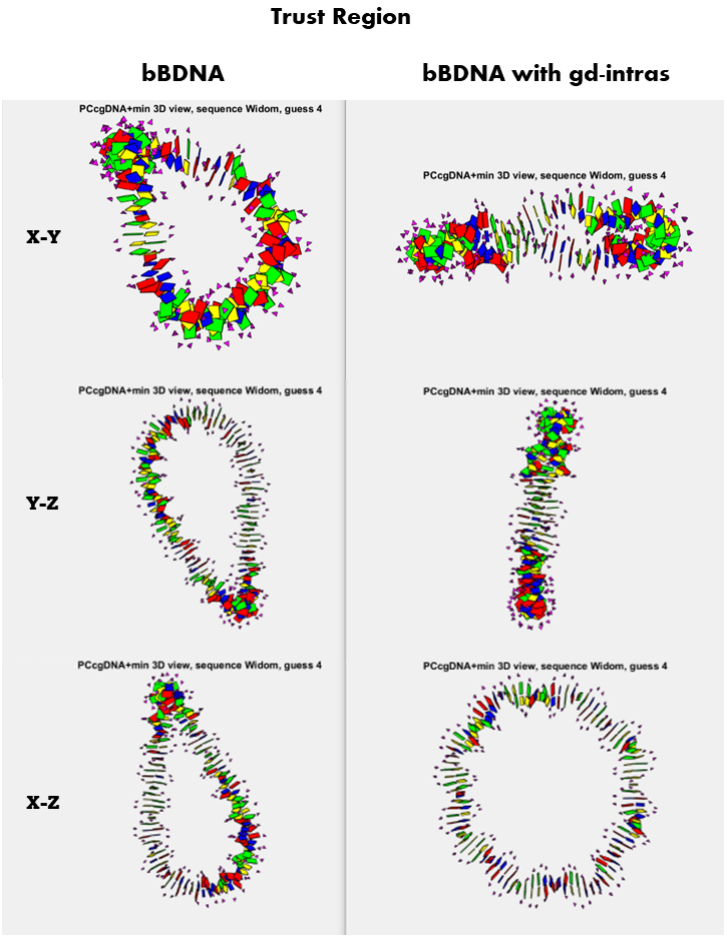


Figure 7: 3*D* structures of the optimums from Trust Region algorithm. The result from bBDNA initial guess is shown in the left side, and the one with ground-state intras is shown in the right side. In both cases they are structure of Widom sequence.

In our case the optimization problem with the Trust Region method and the bBDNA initial guess brings to a configuration with a lower energy and a positive definite Hessian matrix so it is more likely to be a minimum. The other obtained configuration has a greater energy and an indefinite Hessian matrix so it is not a local minimum. We also notice that they have a different number of links, in fact the bBDNA with gd-intras gets to an optimum with the same number of links of the initial guess, i.e. 8; while the other one reaches an optimum with 10 links, which is two more than its starting configuration. Using the Quasi-Newton+Trust Region method the two initial guesses converge to the same solution that have 10 links (same solution as the one obtained with Trust Region and bBDNA). In *Figure 8* is shown the continuum line linking all the Watson phosphates of bBDNA, left side, and bBDNA with gd-intras, right side, when applying only Trust Region method. If we count the number of peaks we obtain the linking number.

3.2.3 Pyne 251bp

We analyze the results for a Pyne sequence, the one which has 251 base pairs. In particular we used four initial guesses to start with which are the same used by M. Beaud in [1].

GUESS 1: we used *fminunc* Matlab function with Trust Region algorithm and tolerance for the first order optimality of $1.0e-6$. The absolute difference of the two configurations obtained with the two different initial guesses is of the order of 10^{-2}. In particular since the two Hessians don't have non-positive eigenvalues we can state that the optimum is a local minimum in both cases (Link number is $m = 25$).

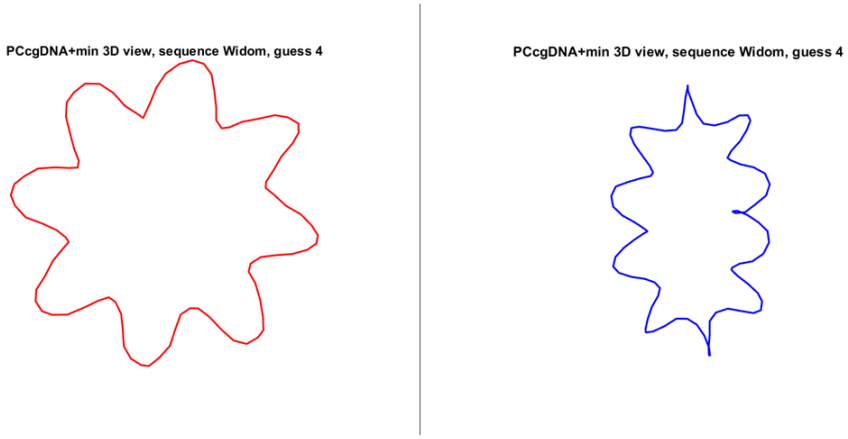


Figure 8: Phosphates line for Widom sequence, Guess 4.

	bBDNA	**bBDNA with gd-intras**
Energy	39.6014	39.5334
Norm of gradient	0.0082	0.0079
Min Hessian Eigenvalue	2.0416e-06	2.1203e-07
Max Hessian Eigenvalue	6.7709e+04	6.7768e+04
Median Hessian Eigenvalue	11.6407	11.6404
Number of negative eigenvalues	0	0
Hessian Condition number	8.4430e+10	8.8909e+11

GUESS 2: we used *fminunc* Matlab function with Trust Region algorithm and tolerance for the first order optimality of $1.0e-6$. The absolute difference of the two configurations obtained with the two different initial guesses is of the order of 10^{-2}. In particular since the two Hessians don't have non-positive eigenvalues we can state that the optimum is a local minimum in both cases (Link number is $m = 24$).

	bBDNA	**bBDNA with gd-intras**
Energy	16.4158	16.4159
Norm of gradient	0.0332	0.0333
Min Hessian Eigenvalue	5.3207e-07	5.8142e-07
Max Hessian Eigenvalue	6.0911e+04	6.7323e+04
Median Hessian Eigenvalue	11.6002	11.6011
Number of negative eigenvalues	0	0
Hessian Condition number	7.9421e+10	8.1229e+11

GUESS 3: we used *fminunc* Matlab function with Trust Region algorithm and tolerance for the first order optimality of $1.0e-6$. The absolute difference of the two configurations obtained with the two different initial guesses is of the order of 10^{-2}. In particular since the two Hessians don't have non-positive eigenvalues we can state that the optimum is a local minimum in both cases (Link number is $m = 24$).

	bBDNA	**bBDNA with gd-intras**
Energy	17.0876	17.1798
Norm of gradient	0.0232	0.0234
Min Hessian Eigenvalue	7.9233e-07	7.9302e-07
Max Hessian Eigenvalue	7.2311e+04	7.2308e+04
Median Hessian Eigenvalue	11.6631	11.6632
Number of negative eigenvalues	0	0
Hessian Condition number	3.6720e+11	8.9902e+11

GUESS 4: we used *fminunc* Matlab function with Trust Region algorithm and tolerance for the first order optimality of $1.0e-6$. The absolute difference of the two configurations obtained with the two different initial guesses is of the order of 10^{-2}. In particular since the two Hessians don't have non-positive eigenvalues we

can state that the optimum is a local minimum in both cases (Link number is $m = 24$).

	bBDNA	bBDNA with gd-intras
Energy	16.6328	16.3974
Norm of gradient	0.0572	0.0424
Min Hessian Eigenvalue	1.5211e-07	2.1331e-07
Max Hessian Eigenvalue	6.7081e+04	6.7083e+04
Median Hessian Eigenvalue	11.6248	11.6250
Number of negative eigenvalues	0	0
Hessian Condition number	9.9887e+10	1.5556e+11

3.2.4 Pyne 339bp

We analyze the results for a Pyre sequence, the one which has 339 base pairs. In particular we used four initial guesses to start with which are the same used by M. Beaud in [1].

GUESS 1: we run *fminunc* Matlab function with Quasi-Newton algorithm and tolerance for the first order optimality of $1.0e-6$. The absolute difference of the two configurations obtained with the two different initial guesses is of the order of 10^{-3}. Since the obtained solutions have still a significant gradient value we decided to run the *fminunc* Matlab function again with Trust Region algorithm and first order optimality of $1.0e-6$, but having as starting point the solutions obtained with the first Quasi-Newton run. Doing that we obtained an absolute difference of the two configurations with the two different initial guesses of the order of 10^{-4} and the configurations are the same M. Beaud found in [1]. Moreover, since the two Hessians don't have non-positive eigenvalues we can state that the optimum is a local minimum in both cases (Link number is $m = 33$).

	bBDNA	bBDNA with gd-intras
Energy	15.5709	15.5434
Norm of gradient	0.0232	0.0113
Min Hessian Eigenvalue	2.1046e-07	1.2818e-07
Max Hessian Eigenvalue	6.7081e+04	6.7083e+04
Median Hessian Eigenvalue	11.6687	11.6688
Number of negative eigenvalues	0	0
Hessian Condition number	8.2116e+11	1.3584e+12

In particular, the solution after Quasi-Newton algorithm has 31 links, instead the one after Trust Region has 33 number of links, as the one Beaud found in [1]. In *Figure 9* is shown the continuum line linking all the Watson phosphates in both the solutions, if we count the peaks we obtain 31 in the left side and 33 in the rigth side.

GUESS 3: we used *fminunc* Matlab function with Trust Region algorithm and tolerance for the first order optimality of $1.0e-6$. The absolute difference of the two configurations obtained with the two different initial guesses is of the order of 10^{-2}. In particular since the two Hessians don't have non-positive eigenvalues we can state that the optimum is a local minimum in both cases (Link number is $m = 32$).

	bBDNA	bBDNA with gd-intras
Energy	20.4514	20.4059
Norm of gradient	0.0876	0.0539
Min Hessian Eigenvalue	2.8500e-07	2.8502e-07
Max Hessian Eigenvalue	6.8838e+04	6.8732e+04
Median Hessian Eigenvalue	11.6586	11.6585
Number of negative eigenvalues	0	0
Hessian Condition number	6.4429e+11	1.0332e+12

GUESS 6: we used *fminunc* Matlab function with Trust Region algorithm and tolerance for the first order optimality of $1.0e-6$. The absolute difference of the two configurations obtained with the two different initial guesses is of the order of 10^{-2}. In particular since the two Hessians don't have non-positive eigenvalues we can state that the optimum is a local minimum in both cases (Link number is $m = 32$).

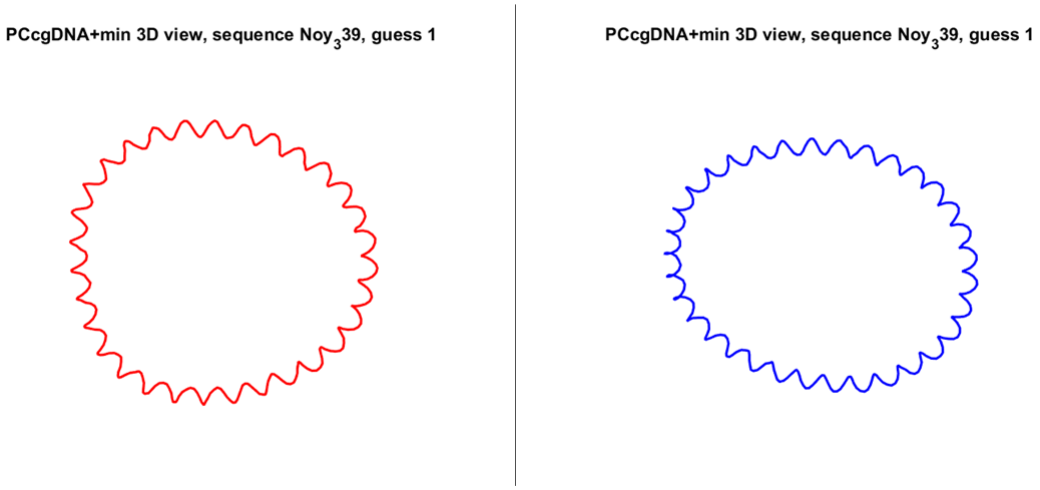


Figure 9: Phosphates line for Pyre 339 sequence, Guess 1.

	bBDNA	bBDNA with gd-intras
Energy	21.2955	21.1221
Norm of gradient	0.0767	0.0557
Min Hessian Eigenvalue	6.4206e-08	1.1106e-07
Max Hessian Eigenvalue	6.6768e+04	6.6768e+04
Median Hessian Eigenvalue	11.6666	11.6665
Number of negative eigenvalues	0	0
Hessian Condition number	2.7352e+12	1.5177e+12

GUESS 9: we used *fminunc* Matlab function with Quasi-Newton algorithm and tolerance for the first order optimality of $1.0e-6$. The absolute difference of the two configurations obtained with the two different initial guesses is of the order of 10^{-3}. The results are shown in the table below.

	bBDNA	bBDNA with gd-intras
Energy	604.9322	604.1401
Norm of gradient	35.8213	37.0695
Min Hessian Eigenvalue	-7.8645e-04	-8.2728e-04
Max Hessian Eigenvalue	6.4288e+04	6.4294e+04
Median Hessian Eigenvalue	11.6202	11.6187
Number of negative eigenvalues	+5	+5
Hessian Condition number	2.3881e+10	2.3640e+10

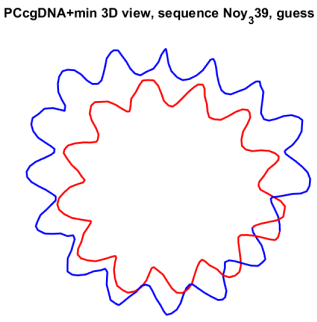


Figure 10: Line linking Watson phosphate of Pyre 339bp, initial guess 9. Number of links: 28. The two colours aim to help counting the peaks.

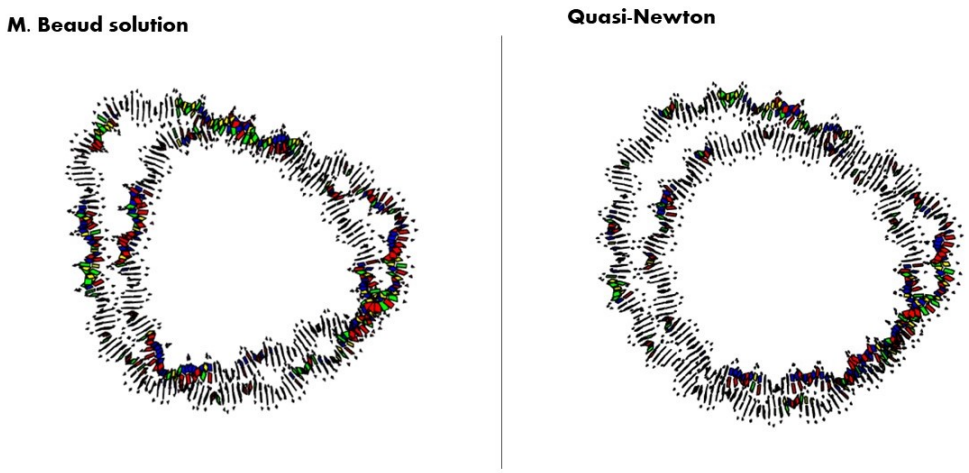


Figure 11: Two different solutions of minimization starting in Guess 9 of Pyre 339. In the right side is shown the solution obtained by M. Beaud in [1]; the one in the left side is obtained using Quasi-Newton method.

If we start running Trust Region method using as initial point the "optimum" found with Quasi-Newton method we obtain a different solution. In particular we obtain a solution similar to the one obatained with initial guess 1. Furthermore, the absolute difference of the two solutions becomes of the order of 10^{-4}. In the table below we show the new solutions data:

	bBDNA	**bBDNA with gd-intras**
Energy	15.6406	15.6078
Norm of gradient	0.2485	0.0364
Min Hessian Eigenvalue	2.9066e-07	1.5568e-07
Max Hessian Eigenvalue	6.7428e+04	6.7428e+04
Median Hessian Eigenvalue	11.6693	11.6692
Number of negative eigenvalues	0	0
Hessian Condition number	7.9021e+11	1.0208e+12

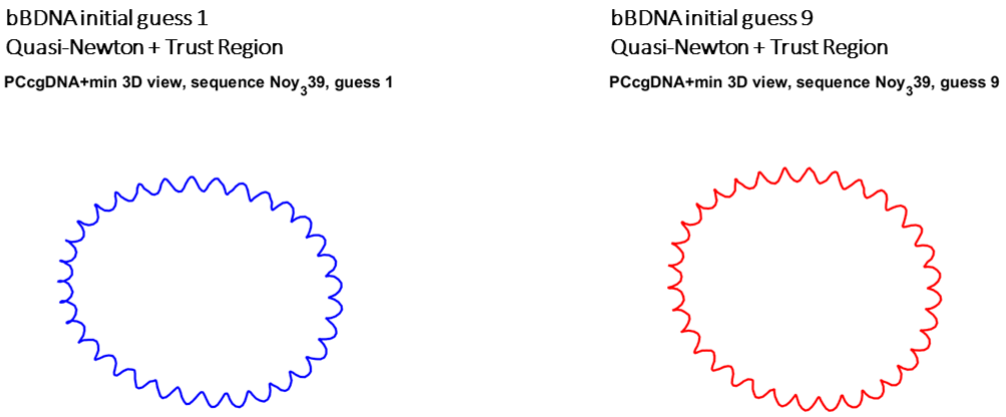


Figure 12: Line linking Watson phosphate of Pyre 339bp, initial guess 1, left side, initial guess 9, left side.

3.3 Conclusion on convergence

We analyzed a total of sixteen bBDNA initial guesses (four sequences) and for each of those initial guesses we carried out a minimization algorithm, and compared it with the optimization of the same initial guess but with the ground-state intra variables. What we obtained is:

- Kahn-Crothers, they all converge to the same solution, and they are all local minima;
- Widom 601, Guess 1, have a really small absolute difference, but one is a saddle point and the other one a local minimum;
- Widom 601, Guess 2, converges to the same local minimum;
- Widom 601, Guess 3, converges to the same saddle point;
- Widom 601, Guess 4, converges to the same local minimum;
- Pyre 251, they all converge to the same solution, and they are all local minima;
- Pyre 339, they all converge to the same solution, and they are all local minima.

We used two different optimization methods, and we can confirm that Trust Region method is more accurate than Quasi Newton one, because, as we stated before, Trust Region methods have way better convergence properties than Quasi-Newton method. However, Trust Region method locally approximates the objective function with a quadratic model in a region in which this accuracy is considered as adequate (for some tolerance). It may happen that for some steps the objective function is far away from being quadratic hence the considered region in those steps is really small. As a consequence if the iteration ends in one of these steps the algorithm starts moving really slow. Meanwhile, the main advantage of using Trust Region method is that it always converges to a stationary point, even if it goes slow. We used Quasi-Newton with some of the initial guesses in order to speed up the algorithm, in fact we discovered that doing a few steps of Quasi-Newton before applying Trust Region method could make the algorithm faster. For example, in Pyne 339bp initial guess 1, using Trust Region after 52 hours the algorithm didn't get to an end, while using Quasi-Newton + Trust Region, after one hour, the algorithm got the optimum value which was the same M. Beaud found in [1] using only Trust Region method.

The rate between the maximum eigenvalue and the minimum one is always between $1.0e+09$ and $1.0e+14$, in particular it corresponds to the condition number of the Hessian matrix. So the minimum eigenvalue is usually really small with respect to the biggest one, hence, the Hessian matrix has a significantly big condition number. So the problem regarding the Hessian matrix is ill-conditioned, and the propagation of an alteration may be really huge, therefore the numerical error might be significant. This leads to oscillations and may be the reason why the convergence to the exact solution of the optimization algorithm is really slow (Trust Region needs the evaluation of the Hessian matrix). Furthermore, this can have brought to numerical issues, for example the fact that in Widom sequence initial guess 1 we get to two really similar solutions but with different number of negative eigenvalues might be the consequence of numerical approximations. Despite that, we had positive results on the convergence to the same point given the two different starting configurations. So, in conclusion, we can say that our analysis confirm the statement that is sufficient to change the inter variables of the periodic ground-state in order to obtain a good initial guess.

4 Uniform inter variables in helicoidal DNA.

The main goal of this second part of the study is to generate an initial guess for covalently closed (both back bones closed) minicircles configurations for a given sequence of n base pairs and a range of integer linking numbers m ($\approx \frac{n}{10.5}$). In order to achieve it we divide the study in two steps:

1. generate special helicoidal configuration that have specific integer link number m;
2. deform the helicoidal equilibrium into a twisted circle;

In this report we focused on the first step, while the second one goes beyond the scope of this semester project.

4.1 Helicoidal configuration of link m

Once we got to the conclusion that if we only change the inter variables of the periodic ground-state we obtain a good initial guess for the energy minimization. The aim of the second part of the study is to construct a minicircle initial guess configuration. We first construct a helicoidal periodic DNA configuration with uniform inter variables, and periodic intras and phosphates. In particular, we will consider a configuration:

$$w = (x_1, u, v, x_2, u, v, \dots, x_n, u, v) \in \mathbb{R}^{24n}$$

where $x_i \in \mathbb{R}^{18}$ $\forall i = 1, ..., n$, represent intras and phosphate coordinates of each base pairs, n is the number of base pairs, $u \in \mathbb{R}^3$ and $v \in \mathbb{R}^3$ are respectively the Cayley vector and the translation vector representing the inters coordinates. It is noticeable that those vectors are the same for each base pairs couple, it means that the relative relationship between two consecutive base pairs is always the same.

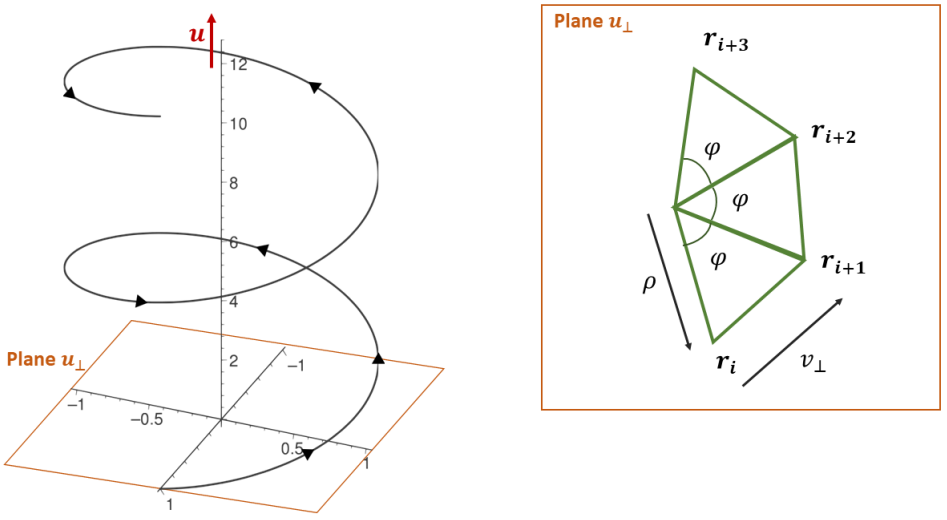


Figure 13: Helix structure as explained in [3].

As we can notice in *Figure 11*, given the base pairs rigid body $X_i = [R_i; r_i]$, the rotation between two base pairs has the same angle φ and the same axis n which is parallel to the Cayley vector u. Furthermore, the helix axis is parallel to u and the distance between the base pairs origin and the center line of the helix is always the same (the radius ρ). In particular we now want to find the equilibrium configuration values $\{x_1, ...x_n, u, v\}$, for a given sequence S. We first define:

$$z = (x_1, x_2, ..., x_n, u, v) \in \mathbb{R}^{18n+6}$$

There exists a matrix P so that $Pz = w$, and P has the following shape:

$$P = \begin{bmatrix} I_{18} & 0 & \dots & 0 & 0 \\ 0 & 0 & \dots & 0 & I_6 \\ 0 & I_{18} & \dots & 0 & 0 \\ 0 & 0 & \dots & 0 & I_6 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & \dots & I_{18} & 0 \\ 0 & 0 & \dots & 0 & I_6 \end{bmatrix} \in \mathbb{R}^{24n \times 18n+6}$$

where I_m is the identity matrix of dimension $m \times m$. We now can rewrite the energy function depending on this new variable z:

$$\begin{aligned} U(w) &= \frac{1}{2}(w-\mu)^\top K(w-\mu) \\ \iff U^*(z) &= \frac{1}{2}(Pz-\mu)^\top K(Pz-\mu) \\ \iff U^*(z) &= \frac{1}{2}\Big((Pz)^\top KPz - 2(Pz)^\top K\mu + \mu^\top K\mu\Big) \end{aligned}$$

If we want to find the vector z that satisfies the energy U^* is sufficient first to evaluate its gradient:

$$\begin{aligned} \lim_{t\to 0} \frac{U^*(z+tv) - U^*(z)}{t} &= \lim_{t\to 0} \frac{\frac{1}{2}\Big((P(z+tv))^\top KP(z+tv) - 2(P(z+tv))^\top K\mu + \mu^\top K\mu\Big) - \frac{1}{2}\Big(Pz^\top KPz - 2Pz^\top K\mu + \mu^\top K\mu\Big)}{t} \\ &= \lim_{t\to 0} \frac{1}{2} \frac{2t(Pz)^\top KPv - 2t(Pv)^\top K\mu + t^2(Pv)^\top KPv}{t} \\ &= (Pz)^\top KPv - v^\top P^\top K\mu = \langle P^\top KPz - PK\mu, v\rangle = \langle \nabla U^*(z), v\rangle \end{aligned}$$

where $\langle\cdot,\cdot\rangle$ is a scalar product. Therefore we should impose $\nabla U^*(z) = 0 \iff P^\top KPz = PK\mu$ in order to find a stationary point and then check the Hessian in order to analyze if it is a local minimum, a global minimum or a saddle point. Since P has rank $18n+6$ and P^TKP is SPD then the problem has a unique solution, hence a unique global minimum.

However, the aim of the study is to create an helicoidal periodic ground-state that can be closed into a minicircle shape.In order to close it the two end parts must interact appropriately, hence the number of links between the two ends must be an integer number m. But with a generic sequence S there is no reason that φ will lead to a complete number of turns m, so if φ is the angle in *rad* associated with the Cayley vector u, we must have:

$$\varphi n = 2\pi m \tag{20}$$

with n the number of base pairs and m the number of links. So the energy minimization is limited to an equality constraint. In fact we know that if u is the Cayley vector, the angle associated to its rotation is given by:

$$\|u\| = 10 \tan \frac{\varphi^*}{10} \tag{21}$$

where $\varphi^* = 5\varphi$ is the angle in $rad/5$. So, given *Equation 17.* and *Equation 18.*, we can conclude:

$$\|u\| = 10 \tan \frac{\pi m}{n} \tag{22}$$

So from now on we will consider the constraint $h(z) = \frac{1}{2}(k - z^T Ez)$, where $k = (10 \tan \frac{\pi m}{n})^2$ and E is the matrix construct as follow, so that $z^T Ez = \|u\|^2$:

$$E = \begin{bmatrix} 0 & \dots & 0 & 0 \\ 0 & \dots & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots \\ 0 & \dots & I_3 & 0 \\ 0 & \dots & 0 & 0 \end{bmatrix} \in \mathbb{R}^{18n+6 \times 18n+6}$$

We can then construct the Lagrangian function for $U^*(z)$ and its equality constraint $h(z)$:

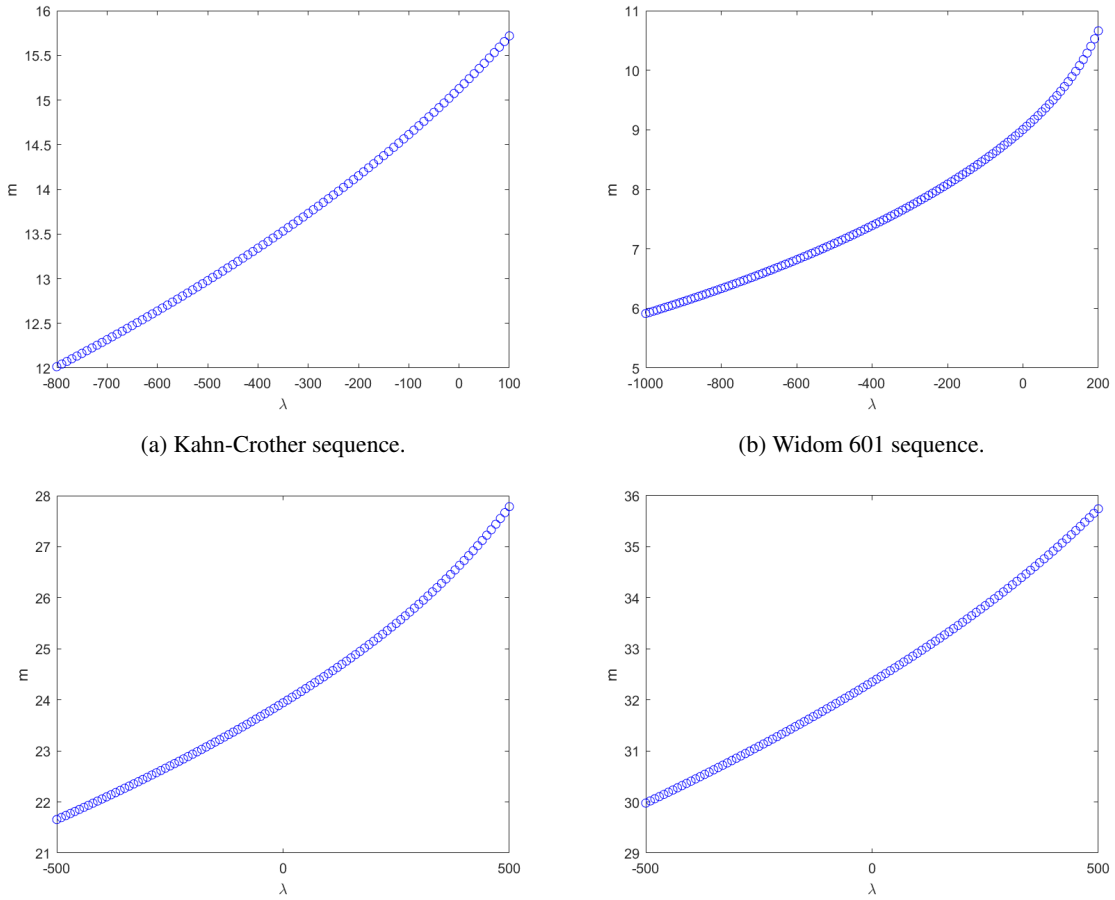
$$\mathcal{L}(z;\lambda) = U^*(z) + \lambda h(z) \tag{23}$$

as a consequence its gradient is of the form:

$$\nabla\mathcal{L}(z;\lambda) = \nabla U^*(z) + \lambda\nabla h(z) = (P^\top KP - \lambda E)z - PK\mu \tag{24}$$

Therefore, if for a range of λ's we find the values of z that solve $\nabla\mathcal{L}(z,\lambda) = 0 \iff (P^T KP - \lambda E)z = PK\mu$ we should check which of those λ verify also the constraint of *Equation 19.*

We solved this equation on *Matlab* for four different sequences, in particular we used *mldivide* solver which has specific algorithms for sparse matrix, as the one we have. Not all λ's are allowed, in fact if λ is too big the matrix $P^T KP - \lambda E$ becomes non SPD (symmetric and positive definite) and as a consequence we cannot solve the system. However, for the specific purpose of the problem, those λ's never happen to verify the constraint, so we will not consider the case in which the matrix is not positive definite. In *Figure 12* below we show, for each sequence, the different λ's that satisfy *Equation 19.* for some specific number of links m. In particular, we can notice that the λ's are all quite small.



(a) Kahn-Crother sequence.

(b) Widom 601 sequence.

(c) Pyne 251 sequence.

(d) Pyne 339 sequence.

Figure 14: Plots weight λ vs number of links m.

Furthermore in *Figure 13*, we show the 3*D* plot of Kahn-Crothers sequence, obtained using $\lambda = -493.91$, which brings to 13 links. In fact if we count the blue segments in the right side of the picture, representing each time the helix passes over the line linking the first and the last base pair, we obtain 13.

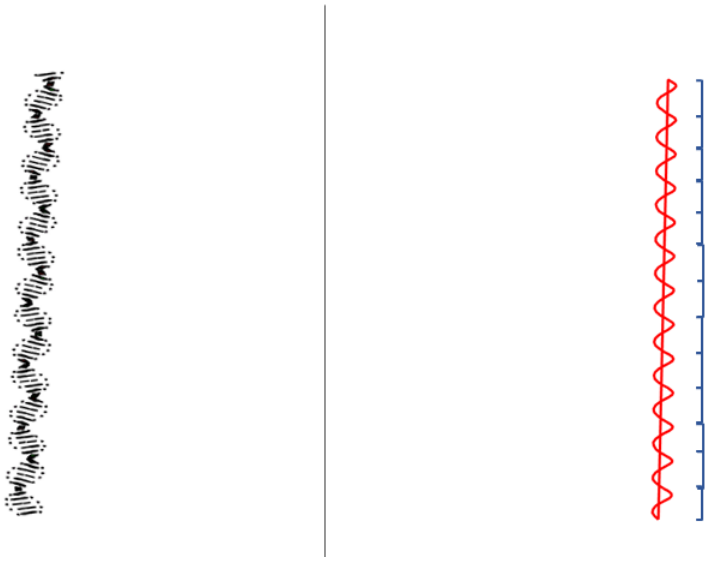


Figure 15: Kahn-Crothers with $m = 13$ and $\lambda = -493.91$. In the left side is shown the $3D$ plot of the rigid bodies; in the right side is shown the line linking all the Watson phosphates.

References

[1] M. Beaud. *Using the cgDNA+ model to compute sequence-dependent shapes of DNA minicircless*. PhD thesis, EPFL, 2021.

[2] BioNinja. DNA Structure, 2019. URL `https://ib.bioninja.com.au/standard-level/topic-2-molecular-biology/26-structure-of-dna-and-rna/dna-structure.html`.

[3] N. Chouaieb, A. Goriely, and J. H. Maddocks. Helices. *Proceedings of the National Academy of Sciences (USA)*, 103:9398 – 9403, 2006.

[4] T. E. Cloutier and J. Widom. Spontaneous Sharp Bending of Double-Stranded DNA. *Molecular cell*, 14 (3):355–362, 2004.

[5] D. M. Crothers, J. Drak, J. D. Kahn, and S. D. Levene. DNA bending, flexibility, and helical repeat by cyclization kinetics. *Methods in Enzymology*, 212:3–29, 1992.

[6] Y.-H. Dai. Convergence Properties of the BFGS Algoritm. *SIAM Journal on optimization*, 2002.

[7] R. Fletcher. Practical Methods of Optimization. *New York: John Wiley Sons*, 1987.

[8] A. Grandchamp. *On the statistical physics of chains and rods, with application to multi-scale sequence dependent DNA modelling*. PhD thesis, EPFL, 2016.

[9] J. Głowacki. *Computation and Visualization in Multiscale Modelling of DNA Mechanics*. PhD thesis, EPFL, 2016.

[10] A. A. Jejoong Yoo, David Winogradoff. Molecular dynamics simulations of DNA–DNA and DNA–protein interactions. *Current Opinion in Structural Biology*, 2020.

[11] F. Lankaš, R. Lavery, and J. H. Maddocks. Kinking Occurs during Molecular Dynamics Simulations of Small DNA Minicircles. *Structure (London)*, 14(10):1527–1534, 2006.

[12] R. S. Manning. Notes on cgDNAmin, Discrete-birod DNA Cyclization, 2017. "unpublished".

[13] I. Miller. The structure of DNA and RNA in the water-mercury interfaces. *Journal of Molecular Biology*, 1961.

[14] A. S. Patelli. *A sequence-dependent coarse-grain model of B-DNA with explicit description of bases and phosphate groups parametrised from large scale Molecular Dynamics simulations*. PhD thesis, EPFL, 2019.

[15] A. L. B. Pyne, A. Noy, K. H. S. Main, V. Velasco-Berrelleza, M. M. Piperakis, L. A. Mitchenall, F. M. Cugliandolo, J. G. Beton, C. E. M. Stevenson, B. W. Hoogenboom, A. D. Bates, A. Maxwell, and S. A. Harris. Base-pair resolution analysis of the effect of supercoiling on DNA flexibility and major groove recognition by triplex-forming oligonucleotides. *Nature communications*, 12(1):1053–1053, 2021.

[16] J. D. Watson and F. H. Crick. Molecular structure of nucleic acids : a structure for deoxyribose nucleic acid. In *Nature*. 1953.