Stochastic Nonlinear Models of DNA Breathing at a Defect

Ciprian Ionuţ DUDUIALĂ

Thesis submitted to The University of Nottingham
for the degree of Doctor in Philosophy of Mathematics

December 2009

Dedicated to my family and to my girlfriend Ana.

Thank you for your support!

Abstract

DEOXYRIBONUCLEIC ACID (DNA) is a long polymer consisting of two chains of bases, in which the genetic information is stored. A base from one chain has a corresponding base on the other chain which together form a so-called base-pair. Molecular-dynamics simulations of a normal DNA duplex show that breathing events – the temporary opening of one or more base-pairs – typically occur on the microsecond time-scale. Using the molecular dynamics package AMBER, we analyse, for different twist angles in the range 30°-40°, a 12 base-pair DNA duplex solvated in a water box, which contains the 'rogue' base difluorotoluene (F) in place of a thymine base (T). This replacement makes breathing occur on the nanosecond time-scale. The time spent simulating such large systems, as well as the variation of breathing length and frequency with helical twist, determined us to create a simplified model, which is capable to predict with accuracy the DNA behaviour.

Starting from a nonlinear Klein-Gordon lattice model and adding noise and damping to our system, we obtain a new mesoscopic model of the DNA duplex, close to that observed in experiments and all-atom MD simulations. Defects are considered in the inter-chain interactions as well as in the along-chain interactions. The system parameters are fitted to AMBER data using the maximum likelihood method. This model enables us to discuss the role of the fluctuation-dissipation relations in the derivation of reduced (mesoscopic) models, the differences between the potential of mean force and the potential energies used in Klein-Gordon lattices and how breathing can be viewed as competition between the along-chain elastic energy, the inter-chain binding energy and the entropy term of the system's free energy.

Using traditional analysis methods, such as principal component analysis, data

autocorrelation, normal modes and Fourier transform, we compare the AMBER and SDE simulations to emphasize the strength of the proposed model. In addition, the Fourier transform of the trajectory of the A-F base-pair suggests that DNA is a self-organised system and our SDE model is also capable of preserving this behaviour. However, we reach the conclusion that the critical DNA behaviour needs further investigations, since it might offer some information about bubble nucleation and growth and even about DNA transcription and replication.

Published papers

An article named "Nonlinear breathing modes at a defect site in DNA" containing the results from Chapters 2 to 5, has been submitted to Physical Review E. This article, having as authors Dr. Ciprian Ionuţ DUDUIALĂ, Dr. Jonathan A.D. Wattis, Dr. Ian L. Dryden, and Dr. Charles A. Laughton, was accepted for publication in November 2009.

The four authors also intend to write another article presenting the results from Chapters 7 and 8.

Acknowledgements

I would like to thank the European Committee for founding my project as part of MMBNOTT – an Early Training Research Programme in Mathematical Medicine and Biology, hosted by the University of Nottingham.

I would also like to thank Dr. Jonathan A.D. Wattis, Dr. Ian L. Dryden, and Dr. Charles A. Laughton, the three supervisors that helped me at each step of this project.

Finally, I would like to thank Steaua Bucureşti for making me happy by qualifying in Champions League groups stage three consecutive years during my PhD studies.

Contents

CHAPTER 1

Introduction

Nature represents a challenge for the scientific community nowadays. Natural processes, such as the wind or the rain, natural resources, for example, coal or oil, and living organisms present interesting phenomena which need further investigations to be explained. Researchers all over the world study these phenomena and create and analyse models of the system, which sometimes reveal hidden features.

Applied mathematics is one of the research fields that developed over the last few thousands years and still continues to develop. Mathematical models allow researchers to analyse a simplified structure of a biological system and predict its behaviour. In fact, interdisciplinary research can offer answers to several unexplained phenomena and mathematical biology, in particular, allows the analysis of living organisms. Such analysis might involve the appearance, the development or even the death of the organisms, or simply explain the causes and the conditions in which a process takes place.

The goal of mathematical biology is to analyse biological systems, using mathematical tools and techniques. Based on the techniques applied in biology and medicine, mathematical biology can be classified into: biological mathematical modelling, complex systems biology, bioinformatics and biocomputing. The first two fields require analytical mathematical knowledge, while the latter two also require computational resources. However, sometimes these fields overlap and a biological application can be considered part of two or more branches of mathematical biology.

A model of a system actually consists of an algorithm or a set of equations that are solved using analytical or numerical methods. These equations allow the imposition of some conditions on the system's behaviour, which influence the mathematical solution. The conditions imposed cover a wide range of system properties, such as equilibrium, non-equlibrium or transition properties.

Recently, many research projects focus on microscopic modelling. Existing techniques are, in many cases, incapable of providing a full analysis at the microscopic level, which explains the need for the development of mathematical models. An example of such a system is *deoxyribonucleic acid* (DNA) in which most processes take place at the Ångstrom level and on a timescale smaller then the microsecond timescale, which is inaccessible even for electron microscopes, such as Scanning Tunnelling Microscope (STM).

1.1 DNA background

Deoxyribonucleic acid (DNA) is a nucleic acid that contains genetic instructions for the development and functioning of living organisms. Note that viruses contain RNA genomes instead of DNA and are not normally considered living organisms. The main role of DNA is the long-term storage of information. The DNA segments which carry genetic information are called genes. There are also DNA sequences with structural purposes and those involved in regulating the expression of genetic information, as well as many redundant and repetitive unused sequences.

From a structural point of view, DNA is a long polymer composed of simple units called nucleotides, which are held together by a backbone of sugars and phosphate groups. The nucleotides composing a DNA sequence differ in their bases, which encode the genetic information copied by cells from DNA into RNA in order to use. These bases are of four types, from two different categories: the purines Adenine (A) and Guanine (G) – having two organic cycles – and the pyrimidines Cytosine (C) and Thymine (T) – with only one organic cycle. Note that nucleotides are structural units for both, DNA and RNA, and have several purposes. Nucleotides not only participate in enzymatic reactions, but also in cellular signalling and they can be sources of chemical energy.

Watson & Crick [119] first introduced, in 1953, the molecular structure of a DNA sequence. A DNA duplex is composed of two chains of bases. A base from one chain has a corresponding base on the other chain which together form a so-called *base-pair*. Adenine (A) forms a base-pair with Thymine (T), while Guanine (G) pairs with Cytosine (C). The bases are linked by covalent bonds along the chains, while the bases of each pair are linked together as follows: A-T pairs by two hydrogen bonds and C-G pairs by three hydrogen bonds [130]. The distance between the bases of a pair is approximately 2 Å and the distance between bases on the same strand is 3.4 Å. In addition, the double stranded DNA is twisted around its central axis. The twist is typically 36° per base-pair – see Figure 1.1.

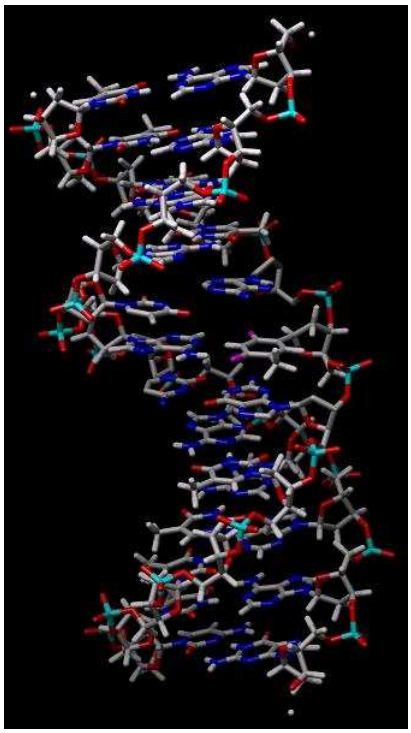

Figure 1.1: Illustration of a 12 base-pairs DNA sequence created using AMBER, for a twist of 36°.

The two strands of DNA twist around the helical axis about once every 10.5 base-pairs. However, undertwisting and overtwisting changes the DNA shape. Topoisomerase enzymes are adding or subtracting helical twist when altering DNA topology. The total DNA length is many times larger than the length of a cell, hence DNA supercoil is necessary to modify its shape such that it fits into the cell.

In a cell, DNA is stored in the nucleus and in mitochondria. The nucleus of human cells is arranged into 46 chromosomes (23 pairs). From a biological point of view, DNA is interesting as part of these chromosomes, which are composed of DNA and proteins. Enzymes are important in DNA lifecycle, since they control most processes involving DNA, such as breathing events, DNA replication, as well as transcription. Most of the enzymes are proteins and represent catalysts for chemical reactions, increasing their rate.

Breathing represents the opening of one or more base-pairs. In other words, a breathing event means the temporary breaking of the hydrogen bonds between complementary bases. The structure formed when at least two consecutive base-pairs are open is called bubble. A bubble moving along the DNA sequence is known as a travelling wave. When enzymes called helicases break the hydrogen bonds linking the two strands of a DNA molecule, a structure called a replication fork is created. This "Y"-shaped structure contains two single-stranded DNA sequences, as well as a double strand, and can move along the chain zipping or unzipping the DNA. At this point DNA replication takes places, a process through which a double-stranded DNA sequence is multiplied, resulting two identical DNA molecules. Another enzyme, known as DNA polymerase, adds matching nucleotides to the two single-stranded sequences and synthesizes the new DNA molecules.

RNA synthesis or transcription is another process controlled by enzymes, more precisely by RNA-polymerase. This enzyme uses the genetic information in DNA to create a messenger RNA (mRNA) sequence, which carries this genetic information to cell's ribosomes, where protein synthesis takes place. Each mRNA molecule is constructed based on a sequence of bases along a DNA strand.

Having this information, computer simulations of the DNA structure can be carried out at different levels of spatial and temporal resolution [129]. In what follows, we present some of the methods used to simulate and investigate processes taking place at the atomic level in DNA. We focus mainly on nucleation of open bubbles, which are at the origin of replication and transcription, and discuss how DNA bending and twisting influence bubble formation.

1.2 DNA mathematical models

One of the techniques used to investigate DNA processes is molecular dynamics (MD) simulations, using computer programs such as AMBER [142]. The biggest inconvenience with such an approach is the time spent simulating a process. The DNA sequence cannot be analysed alone and the solvent surrounding the DNA molecules, which in our case is water, needs to be taken into account. For this reason, during MD simulations a lot of time is lost analysing the solvent containing many times more atoms than the DNA sequence under study, resulting in the overall time needed for just one simulation of a few nanoseconds to be of weeks or months, even when several processors work in parallel. This is why simplified dynamic models of DNA are needed.

Recently, mathematical models of processes that take place in a DNA sequence have been developed. These models can be used to predict the behaviour of DNA and many of them can be used to study DNA denaturation and unzipping – see [68], for example, in which Kafri et al. show that the melting transition, as well as the unzipping transition, are first-order phase transitions. The DNA molecule studied can be viewed as an alternating sequence of denaturated loops and noninteracting bound segments.

Mathematical models can also be used to analyse breathing modes, which represent the starting point for DNA melting and unzipping. Such an event can be examined at both macro and micro-scale. This means, we either observe how the breathers move along the DNA double helix (from one breathing base-pair to a neighbouring base-pair) or we analyse what is happening before, during and after the opening of a single base-pair. The simplest model of DNA breathing consists of an alternating sequence of $0s$ and $1s$, each entry specifying the state of a base-pair: 0 means the base-pair is in equlibrium state, while 1 indicates the open state. Two or more consecutive entries with value 1 represent a bubble and if this bubble travels along the sequence, then we have a travelling wave. More complicated models allow a more detailed analysis of several system properties. For example, Mendes and Laughton [83] describe a way of simulating breathing events that occur when proteins scan a DNA sequence.

1.2.1 Geometrical approaches

A geometrical model of DNA is useful, since it allows multi-directional analysis. One of the first geometrical approaches, introduced by Yomosa [132], considers the projection of each base-pair onto the (x, y)-plane – see Figure 1.2. The direction of the two complementary bases B and $B^{'}$ is given by the angles θ and $\theta^{'}$, respectively, determined by the parallel to Ox axis and the line specific to the hydrogen bonds. One might think that $\theta + \theta^{'} = 2\pi$, but this does not necessarily hold, since the two bases are projected with small deviations from the hydrogen axis that are included in the rotational angles.

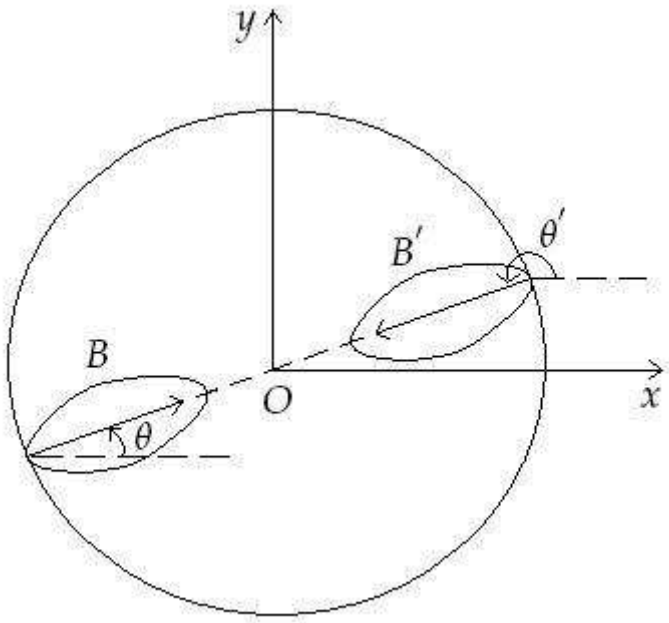

Figure 1.2: Illustration of Yomosa's model.

Using this representation, Yomosa defines the system's Hamiltonian as the sum of the rotational kinetic energy and the inter-strand and along-chain potential energies. The kink and antikink solution of the resulting equations of motion (which have a sine-Gordon form) correspond to the open states with positive and negative helicities. The length of the open sequence and the associated energy are also analysed. In [133], Yomosa considers the rotational angle of each base to be the deviation angle of the base from the imaginary line representing the hydrogen bonds. This new representation reveals four modes of sine-Gordon solitons, describing the existence of open states in double-stranded

DNA molecules. In both papers, the results obtained are compared with experimental data.

Takeno et al. propose several geometrical approaches, in which they study the existence of topological solitons (or kinks), for example [117] and [63], the existence of nonlinear localized modes [115]. They also propose in [116] a three-dimensional harmonic-lattice model with some geometrical constraints. In [117] they propose a generalised form of the dynamic plane base-rotator developed by Yomosa in [132]. Since they are not able to determine an expression for the intra- and inter-strand potential, they use the symmetry of these potentials to determine 2π topological solitons. A similar model is used in [63] to show that, when the intra-strand interactions are much larger then the inter-chain ones, the solitons move along the helical axis.

Zhang [138] studies soliton excitations in DNA as well. The analysis starts from Yomosa's plane base-rotator model, with a modified Hamiltonian that takes into account the dipole-dipole and dipole–induced-dipole energies from his model. In a similar way, each base-pair is depicted by conjugated arrows directed inward and the angles between an arrow and the imaginary line created by the inter-chain hydrogen bonds are measured. The solution of the equations of motion (which form a set of coupled sine-Gordon equations) is compared to experimental data from H-D exchange measurements.

Hennig et al. [57, 60] describe the DNA double helix structure in a Cartesian coordinate system, where the z-axis points along the centre of the helix. The base-pairs are situated into equally spaced planes perpendicular to the central helix axis. They also consider the rotation of each base around the central axis by an angle θ, different for each base. They use this model to study the initiation of the bubble formation process associated with structural deformations of the double helix. In [58] and [59] they focus on the energy exchange processes and the relaxation dynamics in DNA molecules in a nonequilibrium conformation.

1.2.2 Sequence dependent models

Simple nonlinear models allow relevant modes to be analysed. Salerno [105] suggested that sine-Gordon kinks are set in motion in certain regions of a DNA

sequence that include promoters. He analyses nonlinear wave dynamics in the $T7A_1$ DNA promoter region using a model based on the following equations of motion

$$I\frac{d^2\psi_i}{dt^2} = K(\psi_{i+1} - 2\psi_i + \psi_{i-1}) - \frac{\beta}{2}\lambda_i \sin(\psi_i - \theta_i), \tag{1.2.1}$$

$$I\frac{d^2\theta_i}{dt^2} = K(\theta_{i+1} - 2\theta_i + \theta_{i-1}) - \frac{\beta}{2}\lambda_i \sin(\theta_i - \psi_i), \tag{1.2.2}$$

where θ_i and ψ_i represent the deflection angles that two complementary bases form with the imaginary line connecting them, while K is the backbone spring constant, I is the moment of inertia of a base, β is a parameter, describing the strength of the base-pair interaction, and λ_i represents the number of hydrogen bonds involved in pairing the bases ($\lambda_i = 2$ or 3, depending on whether the base-pair is A-T or C-G, respectively).

Salerno's idea was later used by Lennholm and Hornquist [77] to perform a genome-wide study of promoters as dynamical active regions, but they could not prove the existence of a kink-like travelling wave distortion along the DNA chain, since they used the same width for the active regions for all promoters and biological systems are not that regular. In fact, their results are disproved by a recent study of Cuenda et al. [39] who find that kinks move along inhomogeneous sequences in a similar way to those developed by Salerno, which depend on the sequence under study. Moreover, they show that the behaviour observed in Salerno's model is not generated by promoters, but originates from the bases at the boundary. They conclude that this simple model cannot provide relevant information about kinks and breathers. In this way, they also disprove the work of Bashford [14], who also analyses Salerno's model and suggests a relationship between planar moving breather solitons and the helical motion of a sliding protein "particle" about a bent DNA axis. He claims that the solitons he analyses are not thermally-driven, instead base-pair opening is caused by protein-DNA interactions. He also discusses the relationship between transcription and DNA sequences rich in A-T base-pairs.

Alvarez et al. [1] study breather trapping – cessation of breather propagation through the lattice due to lattice parameters varying along the DNA double helix – and breather transmission in a DNA chain in which all base-pairs are identical appart from an interface across which the base-pairs dipole moments

change to the opposite direction. Even if their model is sequence-dependent, they prove that a simple local inhomogeneity creates a mechanism for trapping energy. Nevertheless, Rapti et al. show in [101] that the probability of the formation of a bubble is regulated by the number of A-T pairs in specific regions and the size of the bubble depends on the size of the region which is rich in A-T pairs. This means that a DNA model studying bubbles needs to take into account the number of A-T base-pairs.

1.2.3 One-dimensional models

Both, linear [121] and nonlinear [93] models have been created to analyse DNA denaturation. Zandt analyzes only the transverse displacements in [135], taking into account both the elastic restoring force between neighbours on the same strand and an intra-strand force between complementary bases. If, for the longitudinal interactions, purely harmonic forces are considered, the nonlinear force between chains is the product of the ordinary Hook's law harmonic force and a term causing hard-core repulsion and large separation softening of the force.

Even though some papers study multi-dimensional models of DNA sequences, most DNA models reduce to an one-dimensional system by taking into account only the transverse displacements, as Zandt did. In addition, many models describe how the distances between the bases of each pair vary in time, instead of computing the actual position of each base. Such models are also used to emphasize the links with breather modes or solitons – see [120], for example, in which Wattis studies the form of stationary breather modes in generalised discrete nonlinear Klein-Gordon equations, with symmetric and non-symmetric potential energy functions. The breather solutions are obtained using an asymptotic approach that reduces the system's equations to nonlinear Schrodinger equations at leading order and more complex equations at higher order. An earlier study of Schrodinger solitons in a Klein-Gordon system is presented by Remoissenet in [102], where he describes a general methodology to study breather and envelope solitons in a quasi-1D model.

Another study of Wattis et al. [121] introduces a defect site into a linear lattice

model and finds the system's normal modes by imposing some periodic boundary conditions. This last model is generalized in [122] by modelling the interchain interactions through a nonlinear force-displacement relationship. Moreover, using a change of variable, the model is reduced to one degree of freedom per base-pair. Determining the nonlinear breathing modes that appear at the defect site of the homogeneous nonlinear system created requires an asymptotic approach and multiple scales in space and time.

Peyrard and Bishop [93] proposed one of the first nonlinear models, which neglects the inhomogeneities due to the base sequence and the asymmetry of the two strands. This model ignores the longitudinal displacements, while the neighboring nucleotides of the same strand are connected by a harmonic potential to keep the model as simple as possible. Considering a common mass m for the bases and the same coupling constant k along each strand, they define the system's Hamiltonian as

$$H = \sum_n \frac{1}{2} m \left[\left(\frac{du_n}{dt} \right)^2 + \left(\frac{dv_n}{dt} \right)^2 \right] + \frac{1}{2} k (u_n - u_{n-1})^2 + \frac{1}{2} k (v_n - v_{n-1})^2 + V(u_n - v_n), \tag{1.2.3}$$

where u_n and v_n represent the bases' displacements from equilibrium. The nonlinearity is introduced via the Morse potential

$$V(u_n - v_n) = D(e^{-a(u_n - v_n)} - 1)^2, \tag{1.2.4}$$

with D and a being the depth and the inverse width of the Morse potential. This potential describes the bonds connecting the opposite parts of a base-pair, which are stretched when the double helix opens locally. The analysis of the inter-strand separation dependence on temperature suggests that energy localization might initiate denaturation.

Larsen et al. show in [76] that the bubble generation in a DNA sequence can be viewed as a mechanism in which the two strands open to allow molecule replication, with additional proteins involved in processing or completing the strand separation. Analysing the Peyrard-Bishop model, they reach the conclusion that a larger DNA twist facilitates bubble generation. Englander [46] studies open regions that contain 10 base-pairs in a pendulum-like model and

suggests that such extended open regions could represent thermally activated soliton twist excitations of the double helix.

Olson analyzes normal modes at a base-pair level [78], identifying bending, twisting and stretching modes. It can be shown that chain curvature generates bubbles, that is, increasing curvature increases the tendency for bubble generation. Although some models determine the exact cause of breather-formation, it is also interesting to predict the bubble-size and lifetime in a DNA sequence. Bending of DNA cannot easily be built into Peyrard-Bishop model, since it only considers the interactions between neighbouring pairs. However, considering bending as an inhomogeneity, as well as long range interactions for the along-chain bonds, allows Cuevas et al. [40, 41] to show that the movement of a breather depends on the bending of the chain. They use a mathematical model which is similar to that of a particle moving in a potential barrier.

Using the same Peyrard-Bishop model, Peyrard and Farago [94] prove that, at low temperature, localization is due to individual discrete breathers, while, at high temperature, large regions are involved. Ting and Peyrard [118] transform the equations of motion from Peyrard-Bishop model into perturbed nonlinear Schrodinger equations, using a multiple-scale expansion. The new representation allows them to show that the perturbation induced by a transcription enzyme is more efficient at trapping breathers than an isolated impurity. They obtain that trapping occurs when the amplitude of the incoming breather exceeds a threshold.

Using a simillar model, based on a Morse potential for the inter-chain interactions, Zdravković and Satarić [136, 137] prove that the nonlinear oscillations of DNA nucleotides of large amplitude lead to the unzipping of the DNA chain. Analysing the system for different values of the inverse width of the Morse potential, they reach the conclusion that this parameter plays an important role in the DNA opening.

1.2.4 Twist-opening interactions

Many of the existing DNA models suggest that base-pair opening, as well as bubble generation and trapping are often observed in sequences in which the

curvature is increased. In contrast with the Peyrard-Bishop model in which the bases move only in the direction of the hydrogen bonds, Barbi, Coco and Peyrard have developed in [11] a new model with two degrees of freedom per base-pair, which takes into account the twist-opening interactions – see Figure 1.3. They study analytically the small amplitude dynamics, in which the bases are allowed to move in the plane described by a radial variable r_n specific to the motion along the hydrogen bonds and an angular variable θ_n indicating the base-pair twisting. As can be seen, this system is, to some extent, a simplified version of Yomosa's model, since both bases of a pair are characterised by the same two variables.

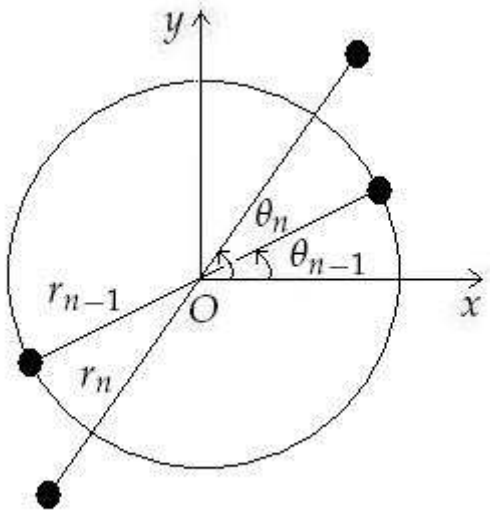

Figure 1.3: Illustration of Barbi-Cocco-Peyrard model.

Several papers analyse Barbi-Cocco-Peyrard model. Based on the derivation of a generalized multiple scale expansion for vectorial lattices [33], Barbi et al. [12] show that the small amplitude approximate solutions of the system are spatially localized and can travel along the sequence. They also study in [13], the static and dynamical properties of this model around its melting temperature. Cocco and Monasson [34] use the Barbi-Cocco-Peyrard model to describe the denaturation of the chain either thermally or mechanically by applying an external torque at the end of the DNA strands.

In another recent paper, Gaeta and Venier [50] identify the conditions for which

solitary travelling waves exist in Barbi-Cocco-Peyrard model. They show that simple asymptotic behaviour and physical values of system's parameters are not enough to satisfy wave existence conditions. In addition, they show that this model admits only solitary waves solutions. These results are compared with the ones from the model for DNA torsional dynamics proposed by Cardoni et al. [21–23], which consists of a double chain of coupled pendulums. The model actually represents a generalisation of the Yakushevich model [128], in which the rotational and torsional degrees of freedom of the DNA sequence are considered to play an important role for DNA transcription. The resulting composite Yakushevich model, as called by the authors and first introduced in [21], splits each nucleotide into several subunits, taking into account, for example, the degree of flexibility and freedom of displacements that the sugar rings exhibit. As presented in Figure 1.4, each DNA strand is considered to be an array of pairs of the form (N, B).

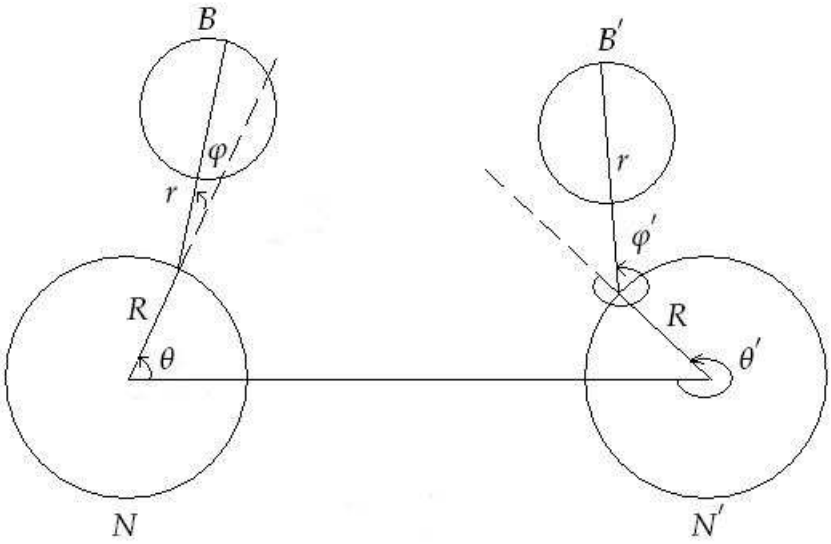

Figure 1.4: Illustration of the composite Yakushevich model.

The base B is a single unit that attaches to the nucleotide N, which is also considered to be a single unit. The attachment point, as well as the centre of the bases and nucleotides define the rotation angle φ and $\varphi^{'}$ of bases around the bond linking them to the nucleotide. Using the hydrogen bond linking the nucleotides of a base-pair we can also define, in the counterclockwise direction,

the sugar-phosphate torsion angles θ and $\theta^{'}$. Note that the system makes sense only if the distance between the nucleotides N and $N^{'}$ is to be greater than $2(R+r)$, where R is the nucleotides' radius and r is the distance from the nucleotide base attachment point to the opposite side of the base. The system's dynamics are then described through the Euler-Lagrange equations derived from a Lagrangian with five components: kinetic energy, backbone torsional potential, stacking potential, pairing potential, and helicoidal potential. Solving the equations of motions numerically, as well as solving the associated system of PDEs, representing the continuous version of the equations, emphasize that the existence of solitons is independent of the detailed modelling of DNA, since the results are similar to the ones of Yakushevich model.

Cardoni et al. [22] generalise this representation, considering the nucleotides and the bases as pendula. The result is a system of two double pendula chains, which in a certain limit reduce to a sine-Gordon equation that supports topological soliton solutions, since the non-topological degrees of freedom are frozen. Furthermore, this model is generalized in [23] to a full class of two-dimensional field theories of sine-Gordon type, which allows one to change the speed of a sine-Gordon solitons by modifying elastic coupling constants and kinematic parameter values. Moreover, breaking the Lorentz symmetry of the system does not modify important soliton properties, such as stability and existence of conserved topological charges.

Yakushevich et al. [131] consider that a model taking into account the asymmetry of the base-pairs is needed. In most of the models presented above, the bases of each pair are considered to have identical structures, having the same masses or moments of inertia, for example. They create a new model in which the two chains of the DNA molecule are two parallel lines and the base-pairs are equally spaced, being all the time perpendicular to the two chains. The bases are not treated as identical structures and have different masses. However, each base is considered a single unit and is only allowed to move around its corresponding chain. Considering that one chain consists only of adenines, while the other one only of thymines, as well as some other inhomogeneous configurations, they determine three types of topological solitons that imitate localized states with open base-pairs. They also show that the solitons can move along the macromolecule with constant velocity and are stable with respect to ther-

mal oscillations, which helps explain the long-range effects in a DNA macromolecule.

1.2.5 Stochastic models

Even with models that consider twisting and treat separately each base of a pair, random oscillations of the base-pair displacements have been observed during breathing events. Bubble lifetime and breathing frequency also exhibit random fluctuations. It is possible to model this type of behaviour using stochastic processes.

In [3], Ambjörnsson et al. show how the probability densities of bubble lifetimes and of the waiting times between successive bubble events can be obtained from a master equation for the joint probability distribution of the bubble size and position along the sequence, for an arbitrary DNA sequence. In [2] and [84] Metzler and Ambjörnsson use dynamic approaches, based on a (2+1)-dimensional master equation and on a Fokker-Planck equation respectively, to study the size fluctuations of bubbles in a DNA molecule in the presence of single-stranded DNA binding proteins (SSBs).

Hanke and Metzler [54] study the bubble dynamics of double-stranded DNA using a Fokker-Plank equation for the bubble's free energy function, which allows them to include microscopic interactions in a straightforward fashion. Another scheme, describing the temporal fluctuations of local denaturation zones in double-stranded DNA, is proposed by Banik et al. in [10]. The scheme, used to study measurable quantities like the bubble size autocorrelation function, is based on a stochastic approach and is computationally efficient, easy to implement and amenable to generalization. In fact, the stochastic approaches may represent mesoscopic models for long timescale simulations of long chains, which are inaccessible to all-atom molecular dynamics studies.

An important problem in stochastic dynamics is that the random terms can increase considerably the temperature, as well as the total energy of the system. Lennholm and Hornquist [77] use the Nosé-Hoover thermostat as the simplest version of such a model. This approach introduces an extra degree of freedom into the system, which has the role of maintaining the temperature at a certain

value. Dauxois et al. [42] reformulate Peyrard-Bishop model using the Hoover reformulation of Nose's method [64] and show that at low temperatures, extended waves interact nonlinearly, but the role of localised excitations grows as temperature rises and these excitations are responsible for DNA melting. Kalosakas and Ares in [70] study based on this model the temperature dependence of the distribution of bubble lengths in DNA segments of various C-G concentrations. In addition, this last model's stationary behavior is studied in [43] by Deng and Zhu. They analyse local DNA denaturation using the stochastic averaging method for a quasi-Hamiltonian system, described by Zhu et al. in [140]. Hien et al. [62] combine the pendulum model of Englander [46] with Peyrard-Bishop model and consider both damping and driving forces in their model. They show that the bubble's length and the kink's velocity depend on system's temperature, as well as on the along-chain coupling interactions.

Quintero et al. [100] use another stochastic approach introducing a damping term into the system, so that energy is conserved. Such an approach relates the damping to the temperature and the noise terms that simulate the random events in the system. Quintero's model allows the computation of expressions, valid up to second order in temperature, for the average and variance of the kink's position and for its mean shape. Muto et al. [86] also introduce noise and damping terms in the equations of motions to describe the system's interactions with a thermal reservoir at finite temperature. Their DNA model considers the two polynucleotide strands to be springs, with the backbone bridges described through an anharmonic Toda potential. The bases of a pair are linked together by hydrogen bonds, which are described by a Lennard-Jones potential. They obtain, from the equations of motion, the expressions for the transverse and longitudinal displacements of each base, which allows the study of wave propagation in their system. They reach the conclusion that the longitudinal anharmonicity might be important in DNA denaturation.

Finally, Cubero et al. [38] study breather nucleation using stochastic resonance in a nonlinear lattice. Their model consists of a quartic potential, that is, the so-called hard ϕ^4 lattice, and a solution is obtained by imposing periodic boundary conditions. The particles from the lattice are subject to a staggered driving force and they optimize breather formation by requiring that the average energy per particle equates to the intrinsic energy of the breather mode. They use

this model to demonstrate that the spontaneous formation and destruction of discrete breathers with a selected frequency are due to thermal fluctuations.

1.3 DNA modelling challenges

Taking all these aspects into account, mesoscopic DNA models can still be considered a challenge for nonlinear science, as discussed by Peyrard et al. in [96]. One challenge is the choice of the potentials describing the interactions from the system. For inter-strand interactions Zhang et al. [139] analyse the Toda lattice potential and the Morse potential. Using a transformation of variables and the Morse potential they prove that a solitary wave excitation with an estimated width of only one or two base-pairs can be obtained. Peyrard et al. [96] suggest that the simple Morse potential is not enough to describe all the DNA effects and proposes a more elaborate function containing a barrier for reclosing the base-pairs.

The stacking interactions (between bases situated on the same chain) are also important in such a complex system. Most papers consider harmonic coupling, but in [69], [95] and [96] it is suggested that a nonlinear stacking leads to a self-amplification process. This improved stacking potential, has the role of weakening the along chain bonds during a breathing event. Presumably, this lengthens breathing events, since it causes a weaker closing force. However, a choice of along-chain and inter-chain potentials that allows breathers to be formed in our system does not guarantee that the DNA behaviour is accurately represented by the mathematical models, unless the mathematical simulations are shown to be close to experimental data or all-atom molecular dynamics simulations.

Moreover, as already discussed, base-pair asymmetry and DNA strands inhomogeneity are not easily incorporated in simple nonlinear models. All these aspects, as well as the random properties that DNA sequences exhibit, influence the dynamics, hence, most of the DNA models developed are only able to predict the DNA behaviour for some particular types of events, such as breathing events.

Base-pair opening in DNA typically occurs on the microsecond timescale [130], which is beyond the scope of all-atom molecular dynamics simulations. However, Guckian et al. discuss in [52] the properties of a 12-mer duplex having a thymine base (T) replaced with the 'rogue' base diflourotoluene (F). They reach the conclusion that the geometry of the Watson & Crick model is not affected by this change, but it leads to the formation of weak hydrogen bonds between the A and F bases. More precisely, only one hydrogen bond links the adenine (A) to the nonpolar molecule (F), weakening the inter-chain interaction at this defect point in DNA. Several studies consider DNA sequences with such a defect to be a probe for the DNA replication mechanism – see [48], for example, in which it is suggested that conventional hydrogen bonds are not crucial for high efficiency and fidelity in DNA synthesis. Moreover, in DNA strands which incorporate a defective base, DNA breathing has been observed to occur on the nanosecond timescale, as presented in a recent study made by Cubero et al. [37].

1.4 Overview

In what follows, we focus only on stationary breathers appearing at a defect site of the lattice that we define. Our molecular dynamics simulations, obtained using AMBER [26], revealed that the frequency, amplitude, and duration of breathing events vary with helical twist, but in a complex way. We therefore seek a simpler model, with fewer variables, that reproduces this twist-dependent behaviour, in which undertwisted DNA (30°-35° degrees per base-pair) display more frequent short-duration breathing events, while overtwisted DNA (37°-40° per base-pair) exhibit fewer longer-duration breathing events. We therefore propose a mesoscopic model for this behaviour, fit it to MD data and compare the results to all-atom AMBER simulations.

The thesis is divided into three parts. The first of them is self-contained and presents the DNA AMBER simulations and details about the stochastic differential equations (SDE) model that we propose. The second one includes the system analysis and a comparison between AMBER and SDE results, while the third part contains the Appendix describing AMBER files needed to simulate

our system, some figures sustaining the ideas presented in Part II and the papers cited along the thesis.

We start Part I by briefly presenting, in Chapter 2, the AMBER package and how we create the input files (presented in details in the Appendix) needed to simulate a 12-mer DNA sequence, containing a difluorotoluene (F) base in place of a thymine (T) base. The DNA molecule is solvated in a water box and, after performing energy minimization operations, the system is simulated using AMBER's component SANDER. The Chapter ends with the methodology needed to extract the relevant information from the simulations output files.

Chapter 3 introduces a new stochastic differential equations (SDE) mesoscopic model for double-stranded DNA useful to study individual breathers appearing at the defect site of our lattice. Using a change of variables, we reduce the model to one-dimension, considering each base as a single particle. We use an harmonic stacking potential, while for the inter-chain interactions the expression of the potential is determined from the free energy of the breathing pair. Based on studies of sympletic methods capable of preserving energy-like quantities [19], we introduce noise and damping terms into the system. Next, we describe the maximum likelihood estimator (MLE) method needed to fit the unknown parameters to data obtained from AMBER simulations. In addition, we demonstrate the need of an alternative fluctuation-dissipation relation, for reduced mesoscopic models.

The system parameters values are derived in Chapter 4, which also includes a discussion on how these values vary with twist angle and influence breathing. We conclude that breathing can be viewed as competition between the along-chain elastic energy, the inter-chain binding energy and the entropic component of the free energy, which is due to the forcing and damping induced by the solvent, which slows the DNA atoms and changes the dynamics of the DNA molecule.

Next, in Chapter 5, we apply the implicit midpoint method to simulate the breathing process of a 12 base-pair DNA sequence, using the SDE model. The comparison with the simulations obtained using AMBER reveals that our results are close to all-atom MD simulations, which implies that system definition and parameters fitting methodology are consistent. Small differences can

be observed between the degree of randomness of the two methods, but we conclude that the AMBER simulations, as well as the SDE simulations, are random. Also, longer SDE simulation of 100 nanoseconds are presented for 30° and 38° twisted DNA sequences.

The second part of this thesis, starts with Chapter 6, in which we introduce three traditional methods, which can be used to analyse Hamiltonian system. We first introduce the principal component analysis (PCA) method, which is a quantitative analysis tool. Then, we discuss the data pre-processing that is sometimes required before applying PCA and define the Mahalanobis distance, which can be measured in the principal component space. Next, we present the data autocorrelation function, which gives information about the data dependence on the system's initial conditions. Finally, we describe the normal modes decomposition of Hamiltonian systems and we present a method of determining the specific frequencies and vectors using the Fourier Transform. We end Chapter 6 by applying these analytic methods to a simple example to demonstrate how can the properties of a system be retrieved from simulation data.

In Chapter 7 we apply the traditional methods discussed in Chapter 6 to the DNA trajectory data, obtained using both AMBER and SDE models. After discussing the difficulties of constructing predictive models based on principal components, we show the agreement between AMBER and SDE data in terms of principal components, autocorrelation, and, least but not last, Fourier Transform expressions.

The Discrete Fourier Transform (DFT) suggests that DNA exhibits the so-called "self-organised criticality" (SOC) property, which is discussed in detail in Chapter 8. First, we introduce basic SOC notions such as power laws, fractals, flicker noise and cellular automaton, and next, we present several self-organised systems, which link critical behaviour to $1/f$ (flicker) noise. Finally, we determine the log-representation of DFTs characteristic to our DNA datasets and we show they scale into power laws for both AMBER and SDE simulations and we conclude that DNA is a self-organised system.

We draw the conclusions of this thesis in Chapter 9, by summarising the DNA models discussed and the results obtained during the system analysis. Finally, the last part of the thesis contains the Appendix and References sections, in

which files, figures, and scientific literature papers discussed in this thesis are presented.

Part I

DNA Simulations

CHAPTER 2

The molecular-dynamics package AMBER

AMBER (*Assisted Model Building with Energy Refinement* [142]) represents one of the alternatives to simulate DNA molecules dynamics and investigate breathing events. It includes two main parts:

1. a set of molecular mechanical force fields (atom types in the system, parameters for all of the bond lengths, angles and dihedrals);
2. a package of molecular simulation programs, also known as AmberTools.

Note that the programs from AmberTools work without AMBER, but AMBER itself cannot be used without the tools package. Although the set of force fields is dispensable, it is useful when we define a new molecular structure. The predefined values of bond lengths, angles and dihedrals for different atoms shorten the time needed to determine the structure of a DNA sequence, for example.

The latest version available is Amber10, but for our simulations, we used Amber9 [26]. The main disadvantage of this MD package is the time needed to simulate a normal DNA duplex, since recent experiments show that breathing events occur on the microsecond time-scale, while the integration time-step used by AMBER is expressed in picoseconds and for best results it is recommended to use a 0.002 ps time-step. Moreover, when the input files for AMBER

are constructed, we need to take into account that the DNA sequence is surrounded by a solvent, which in our case is a water box, typically having more than ten times as many atoms as the DNA duplex.

2.1 The DNA sequence analysed

We analyse the breathing events that occur in a DNA duplex containing the 'rogue' base diflourotoluene (F) in place of a thymine base (T), as proposed by Cubero et al. in [37]. Replacing a T base with the F base, weakens the inter-chain interaction at that point (since effectively only one bond of hydrogen links the two bases), causing breathing to occur on the nanosecond time-scale, rather than microsecond, which reduces the time needed to simulate the system in order to perform a complete analysis of the breathing events by a factor of 1000. The DNA sequence analysed contains 12 base-pairs as follows:

C	T	T	T	T	G	F	A	T	C	T	T
G	A	A	A	A	C	A	T	A	G	A	A

This sequence is analysed at a constant temperature of $T = 293K$, in the presence of a surrounding water box. The box has to be taken into account because it influences the atoms interactions through the hydrogen bonds linking the bases from the same DNA strand [95]. Even if the breathing events occur on the nanosecond time-scale and the DNA sequence contains only 12 base-pairs, which together with the sugars and phosphate groups represent 763 atoms, the number of degrees of freedom in our system is actually very large (16682) due to the water box. This means most of the time is spent computing information about the solvent, even though this information is not used for our analysis, since we focus only on the DNA bases and their dynamics.

Moreover, the computations involve complex interaction potentials and therefore, require several processors working in parallel and several weeks of work. For example, for a 20 nanoseconds simulation, we needed about 10 days and 4 processors working in parallel. In order to reduce the system complexity, we

need to create a new model that incorporates the effect of the solvent, but which only deals with the DNA bases.

Our DNA sequence is analysed for different twist angles in the interval 30°-40° per base-pair, more precisely five angles for an undertwisted DNA sequence (30°, 32°, 33°, 34° and 35°), the typical twist angle of 36° and two angles (38° and 40°) for an overtwisted DNA sequence. The way in which the twist angle influences the structure of the DNA sequence can be observed in Figure 2.1, in which the eight twist angles analysed are presented.

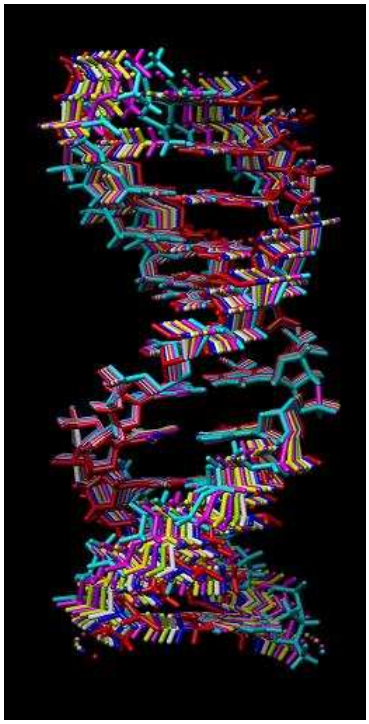

Figure 2.1: The DNA sequence under study for different twist angles in the range 30°-40°.

Note that AMBER considers that the normal twist is by default about 32.5°. In order to avoid this inconvenience, we have constructed the DNA sequence by considering the degree of twist at rest. Next, the twisting degree was preserved by imposing a harmonic restraint on the atoms at the end bases. More precisely, we have considered a constant energy (of 1 kcal mol^{-1} $Å^{-2}$) and hence a constant force acting on the end bases, in order to keep the DNA atoms close to their initial positions. However, applying this restraint only to the end bases, allowed the A-F pair to breathe by exploring a larger volume of space than the other base-pairs.

2.2 Simulating the system using AMBER

The basic MD program contained in AmberTools is called *SANDER*. It requires several input files. Three of them are crucial for a MD simulation:

- a topology file (".top") containing the type of atoms in the system (including the water box), in the order in which the information about each atom is added to the output files, and the necessary force field parameters – a description of such a file can be found in Appendix A.1;
- a coordinates file (".crd") containing on the first line the number of atoms of the analysed system, on the next lines the 3D coordinates of each atom at the initial position, in the order given by the topology file, and optionally this file may contain velocities and current periodic box dimensions – see Appendix A.2 for more details;
- MD file (".in") representing the *SANDER* input file and consisting of several namelists and control variables needed to determine the type of simulations to be processed – examples of such files can be found in Appendix A.3.

2.2.1 Creating input files

The scope of this section is not to present how to build from scratch a system, since several tutorials are available on the official AMBER website (at the address http://ambermd.org/tutorials), but the most important steps are explained. Note that a force field has to be specified in order to be able to use *LEaP*. The direct way to specify the force field is using one of the *leaprc* files, containing predefined force fields, that can be found in $AMBERHOME/dat/leap/cmd directory. For a DNA molecule, the predefined FF99SB all-atom force field can be used. Moreover, AMBER provides several water models – the default one is TIP3P – that are used for residues with name WAT. The topology and coordinates files are created using the *LEaP* command `saveamberparm`.

The program *LEaP* provides a platform for carrying out the modelling tasks. Reading in the force field, topology and coordinates, it produces the files nec-

essary for the MD simulation. First of all, we need to create the structure of our 12-pairs DNA. It is possible to use an experimentally determined structure or to create a new structure using *nucgen*, which allows generating canonical A- and B- duplex geometries of nucleic acids by specifying the base-pairs of our DNA sequence. This program produces a *pdb* file. Such a file usually contains information about each atom in the system: an unique number identifier needed for future references, the atom type, the name and the number of the residue (nucleotide or water molecules, in our case) containing the atom and the Cartesian atom coordinates. The residues from the *pdb* file are considered to be connected, in the order of their listing, and separated, when a line containing the reserved word `TER` is inserted between two residue. At the end of the structure definition process, all this information needs to be in agreement with the topology and coordinates files.

The *pdb* file produced by *nucgen* does not contain the hydrogen atoms or any of the water molecules. *LEaP* allows reading and writing *pdb* files, constructing new residues and molecules, linking together residues, creating nonbonded complexes of molecules, solvating molecules in arbitrary solvents, modifying internal coordinates within a molecule or generating topology files. Performing such operations is helpful in adding new residues, like neutralizing counterions or solvents, and the result is the specification of the complete force field and of the complete DNA sequence structure, as well as the creation of the MD simulation input files. Appendix A.4 contains an example of a *pdb* file containing the final structure of a DNA sequence.

The *pdb*-format files are also used for visual analysis of the system and are not involved in the actual simulation of our system. We use *pdb* files to obtain coordinates and topology files, but the inverse operation is also possible, using *ambpdb* filter, which transforms a coordinate file into a *pdb* file, using the information contained in a topology file. Such operations are needed when we use a predefined DNA structure, for example. During visual analysis, this type of file is usually used in conjunction with a trajectory file, which is one of the possible output files of a MD simulation.

The last step in the file preparation process is the creation of the MD input files. Note that *SANDER* can also be used for energy minimization, which involves

a structure relaxation. The coordinates file contains some initial values that do not guarantee a minimum of the energy, which reduces the possibility of having conflicts or atoms overlaps. The actual MD simulation is based on the integration of Newton's equations of motion, allowing, at the same time, the structure to cross over small potential energy barriers. *SANDER* also provides a mechanism of saving configurations during the simulation at regular intervals, as well as adding constraints to the force field. In order to accomplish these tasks we define several *SANDER* input files, which will specify the values of several parameters, depending on the operation performed (energy minimization or MD simulation – see Appendix A.3 for examples for both cases).

We decided to split the energy minimization into two steps: one in which only the water box energy is minimized and another one in which all molecules, except *Carbon*, are taken into account. The input file for the first minimization step, contains the following section:

```
&cntrl
      IMIN=1, MAXCYC=5000,
      NCYC=50, DRMS=0.5,
      IBELLY=1, NTB=1
&end
```

The variable `IMIN` specifies that we perform minimization, not molecular dynamics. `MAXCYC` represents the maximum number of minimization cycles. The method of minimization will be switched from steepest descent to conjugate gradient after `NCYC` cycles and the convergence criterion for the energy gradient is given by the root-mean-square of the gradient, which has to be less than `DRMS`. Finally, `IBELLY` shows that only a subset of the atoms in the system is allowed to move, and the coordinates of the rest are frozen, while `NTB` specifies the periodic boundaries conditions used (in our case, the volume is considered to be constant). After the parameters section, the residues allowed to move are specified using `RES` directive followed by two lines containing the keyword `END`.

The second minimization input file has the same structure, but the `IBELLY` parameter is not needed and `NTR` is used instead of `NTB` by setting its value to be

1, which turns on Cartesian restraints. Also instead of allowing the residues to move, we only specify the constrained atoms, using the directive ATOM.

Next, we define several input files that are used in cascade in order to simulate the system. The files structure contains the following namelist:

```
&cntrl
      IREST=1, NTX=7,
      NTF=2, NTB=2, SCEE=1.2, CUT=9.0,
      NTR=1,
      NSTLIM=500000, DT=0.002,
      TEMP0=300.0, NTT=1,
      NTWX=500, NTWE=500, NTWV=500
      NTP=1,
      NTC=2,
&end
```

The parameters IREST and NTX indicate that the simulation is restarted using the output coordinates file from the previous step. Observe that for the first MD simulation, the coordinates file is the output file of the minimization process. NTF specifies that the bond interaction involving H atoms are ignored and NTB value shows that constant pressure dynamics theory is used. CUT is used to specify the nonbonded cutoff, in Angstroms, used to limit direct space sum, while SCEE describes the electrostatic interactions. If NSTLIM represents the number of MD steps performed and DT is the time-step in picoseconds, NTT is a variable showing that temperature scaling is used in order to keep the system in equilibrium – in our case the temperature is considered constant with value TEMP0 and the weak-coupling algorithm is used for rescaling. Moreover, NTP shows that MD with isotropic position scaling is used, while NTC indicates that the length of bonds involving hydrogen are constrained. Information about every NTWX, NTWE and NTWV steps will be written in the output files concerning the trajectory, energy and velocity, respectively.

In addition, the first MD file needs to specify, in the Section &cntrl, the value of IG (the seed for the random number generator, on which the MD starting velocity is dependent). Also, some of the first MD simulations can be considered

part of the system equilibration process. They will be shorter then a normal simulation and a useful technique is to start simulating the system at a temperature of $100K$ and to increase the temperature to a value around $300K$, to allow breathing events, while ensuring the DNA does not melt. Moreover, the MD will be performed only on water or water and ions, for example, in order to reach equilibrium more easily. Also some other sections can be included in the input file, such as the weight change information section, which is repeatedly read (if `NMROPT`>0 is specified in the `&cntrl` section) as a series of namelist specifications, until a namelist `&wt` statement is found with `TYPE='END'`. This section is useful when the system temperature is changed, as previously suggested:

```
&wt
   TYPE='TEMP0',
   ISTEP1=0, ISTEP2=4999,
   VALUE1=100.0, VALUE2=300.0
&end
&wt
   TYPE='END'
&end
```

The temperature value `VALUE1` is replaced with `VALUE2` and the change takes place between time-steps `ISTEP1` and `ISTEP2`.

When all these goals are achieved, we start the actual MD simulation of the system, during which, in our MD files, the `&cntrl` structure is followed by a directive that specifies a Cartesian restraint on the four terminal base atoms:

```
10.0
ATOM 11 22 360 373 391 404 742 756
END
END
```

This section is different for the first few MD files, involved in the system equilibration, depending on the specific task performed. For example, we can specify the atoms that are going to be tightly restrained in the MD equilibration simulation.

2.2.2 System simulation

Preparing the input files may require as much time as simulating the system. Any mistake made while creating the DNA structure and the input files can generate an error propagated in the output files of the MD simulation. In other words, if the DNA structure does not have consistency and if the minimization and simulation processes are not planed correctly, the analysis of the results obtained is meaningless. Using the topology, coordinates and MD input files, we can proceed with the system energy minimization and the actual DNA system simulation.

The sequence of *SANDER* commands used is the following:

a) perform minimization

```
$AMBERHOME/exe/sander -O -i min1.in -o min1.out -inf
min1.inf -c DNA.crd -ref DNA.crd -r DNA.min1 -p DNA.top

$AMBERHOME/exe/sander -O -i min2.in -o min2.out -inf
min2.inf -c DNA.min1 -ref DNA.min1 -r DNA.min2 -p DNA.top
```

b) perform few equilibration and temperature changing MD simulations

```
$AMBERHOME/exe/sander -O -i md1.in -o md1.out -inf
md1.inf -c DNA.min2 -ref DNA.min2 -r DNA.md1 -p DNA.top

$AMBERHOME/exe/sander -O -i md2.in -o md2.out -inf
md2.inf -c DNA.md1 -ref DNA.md1 -r DNA.md2 -p DNA.top

.......

$AMBERHOME/exe/sander -O -i md10.in -o md10.out
-inf md10.inf -c DNA.md9 -ref DNA.md9 -r DNA.md10
-p DNA.top -x DNA.md10.x -e DNA.md10.ene -v DNA.md10.v
```

c) perform MD simulations

```
$AMBERHOME/exe/sander -O -i md11.in -o md11.out
-inf md11.inf -c DNA.md10 -ref DNA.md10 -r DNA.md11
-p DNA.top -x DNA.md11.x -e DNA.md11.ene -v DNA.md11.v

$AMBERHOME/exe/sander -O -i md12.in -o md12.out
-inf md12.inf -c DNA.md11 -ref DNA.md11 -r DNA.md12
-p DNA.top -x DNA.md12.x -e DNA.md12.ene -v DNA.md12.v

.......
```

d) continue until the desired number of data points is obtained

The files with extension ".x", ".ene" and ".v" represent the trajectory, energy and velocity files, respectively, while the ".md*" represent the restarting coordinates file. Also, information about the simulation in progress are obtained in ".out" and ".inf", which are the log file and the summary file.

Note that the topology file is very important for this process, since it does not modify during the minimization process or during the simulation process. Indeed, the structure of the DNA sequence is not modified by our computation. The measurable quantities like the system energy or atoms coordinates and velocities will modify, but they will not affect the DNA structure, since the temperature is considered to be constant and hence, the DNA melting point is not reached.

Finally, after obtaining the velocities and trajectory files, we will eliminate the information that we do not need in order to analyse the system.

2.3 Interpreting AMBER results

The files generated using AMBER contain the coordinates (x_1, x_2, x_3) of each atom of each base at every time step, as well as the velocity values (v_1, v_2, v_3)

for each atom. We only need information about the atoms from the extremities of our bases, which is then used to compute the distance and the velocity corresponding to each base-pair. For example, for the A-F pair, we measure the distance between the N1 atom of the A base and the H3 atom of the F base, having the unique identifier number 213 and 561, respectively – see Appendix for more details. As can be seen in Figure 2.2, the choice of the atoms between which the distance is measured is not unique. It is possible to measure the distance between the centers of mass or geometrical centers of the two bases, but these methods require more time and more resources. Hence, to measure the distance between two base-pairs, choosing the two atoms between which the distance is minimum, seems to be the most reasonable thing to do.

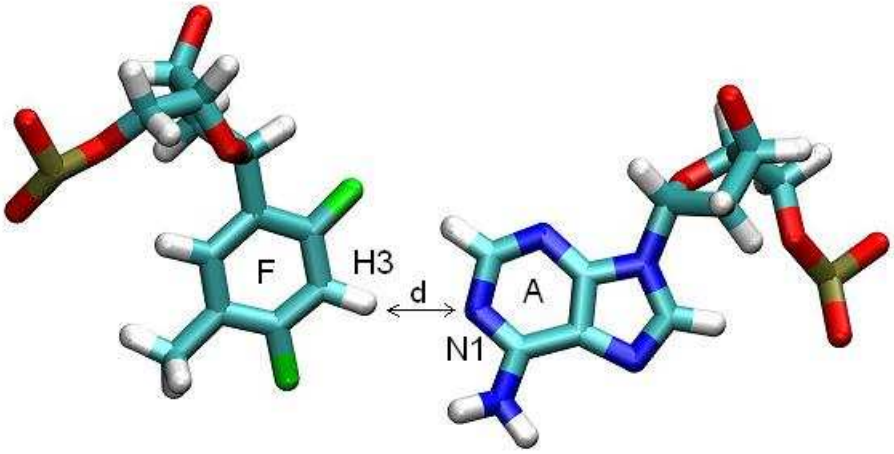

Figure 2.2: Illustration of the A-F base-pair.

The program *ptraj* contained in AmberTools is an alternative for processing coordinates/trajectories. Using it, other information can be obtained, such as the center of geometry of a group of atoms, the angle between three atoms or the distance between two atoms. We used *ptraj* to strip off information about atoms not needed for our computations.

A file designed to process such a task has the following structure:

```
#!/bin/sh
ptraj DNA.top << EOF
     trajin DNA.md11ls.x
```

```
        trajin DNA.md12ls.x
        trajin DNA.md13ls.x
        trajin DNA.md14ls.x
        trajin DNA.md15ls.x
        trajin DNA.md16ls.x
        trajin DNA.md17ls.x
        trajin DNA.md18ls.x
        trajin DNA.md19ls.x
        trajin DNA.md20ls.x

        trajout DNA.x nobox
        strip @1-212,@214-560,@562-16682
EOF
```

The topology file is again, very important for our analysis, since it specifies the order in which the atoms coordinates are represented in the trajectory files and thus, offers a way of determining the coordinates needed. The commands *trajin* and *trajout* specify the input and output files, respectively. We can have several input files, but we have to be careful about the order in which they are analysed, because the final result depends on the processing order. Inserting the keyword *nobox* after the output file name specifies that the water box is ignored, otherwise another three coordinates, representing the center of mass of the water box, are added to the final trajectory file.

Next, we repeat the stripping procedure, for the velocities files. As will be explained in the next chapter, it is not enough to know the distances between the bases of the DNA pairs, we also need their velocities, in order to obtain an accurate fit of the parameters for the reduced model proposed.

After obtaining the trajectory and velocity values for the extremities of each base-pair we have four arrays containing information about the position and velocity of each base of a pair: $\mathbf{x^1} = (x_1^1, x_2^1, x_3^1)$, $\mathbf{v^1} = (v_1^1, v_2^1, v_3^1)$, $\mathbf{x}^2 = (x_1^2, x_2^2, x_3^2)$ and $\mathbf{v}^2 = (v_1^2, v_2^2, v_3^2)$. In this case the displacement vector is $\mathbf{d} = \mathbf{x^1} - \mathbf{x^2}$ and hence the distance is

$$d = \sqrt{(x_1^1 - x_1^2)^2 + (x_2^1 - x_2^2)^2 + (x_3^1 - x_3^2)^2}, \tag{2.3.1}$$

while the velocity is $\mathbf{v} = \mathbf{v^1} - \mathbf{v^2}$ and hence the speed in the direction $\widehat{\mathbf{d}} = \mathbf{d}/d$ is $v = \mathbf{v} \cdot \widehat{\mathbf{d}}$, which is given by

$$v = \frac{(x_1^1 - x_1^2)(v_1^1 - v_1^2) + (x_2^1 - x_2^2)(v_2^1 - v_2^2) + (x_3^1 - x_3^2)(v_3^1 - v_3^2)}{d}. \tag{2.3.2}$$

Next, observe that d represents the real distance between the two bases of a pair and we are interested in how this distance evolves over in time. Hence, we need to subtract the mean value, which we take to be a good approximation to the equilibrium displacement, of the distance vector from each value of the vector: this is about 2.6 Å for the A-F base-pair and 1.96 Å for all the other pairs.

Having completed all the steps presented above, we have prepared all the files needed to analyse breathing for our 12-mer DNA sequence. The results obtained, revealed that the frequency, amplitude, and duration of breathing events varies with helical twist, but in a complex way that at present we do not fully understand. More precisely, an undertwisted DNA molecule (30°-35° per base-pair) displays short, but frequent breathing events, while the over-twisted DNA sequences (37°-40° per base-pair) breathing lasts longer, but is less frequent. Therefore, we developed a reduced model, with fewer variables, capable of reproducing this twist-dependent behaviour. We present AMBER results in the Chapter 5, where a comparison with the proposed mesoscopic model simulations is made.

We fit the parameters of the reduced system to AMBER data, such that it incorporates the water contribution to the potential energy, even thought we only consider the bases of the DNA sequence. This coarse-grained model will reduce the time needed to simulate the system, as well as the resources needed to store the information (during and after the simulation), but will also explain breathing through the set of system parameters.

2.4 Summary

This chapter presents the steps that have to be completed to simulate a DNA sequence using AMBER. After introducing the defective DNA sequence that we analyse, we show how to create the DNA molecule using *LEaP* and the input

files required by *SANDER* to simulate the system. After completing the system energy minimization phase and the DNA simulation, we process the data into a form that allows us to measure the oscillations from equilibrium of the bases. Finally, the simulation results, that we discuss in detail in Chapter 5, show the need for a reduced model to explain the breathing length and frequency twist dependence.

CHAPTER 3

Model

The reduced model that we propose is able to reproduce the behaviour of a DNA sequence containing an A-F base-pair. We model the DNA molecule through a lattice consisting of two chains of bases, which contains a defect at the middle site of the lattice. This defect is considered in both, along-chain and inter-strands interactions, and represents the only base-pair of the sequence expected to breathe. Note that we create a model that simulates with accuracy not only breathing events taking place at the defect site, but also the behaviour of the neighbouring base-pairs, compared to AMBER simulations.

3.1 Preliminaries

We consider each nucleotide in the DNA strands to be a separate point mass linked to three other bases: one in each direction along the same chain and one on the complementary chain, as in Figure 3.1. The inter-chains bonds are modelled by nonlinear force-displacement relationships, while the intra-chain bounds are modelled as a linear spring with constant k as shown in [122]. Although we construct a model with $4N$ bases – this means $2N$ base-pairs – which can be viewed as a lattice of order N, we want to have a similar system as in the microscopic case. Hence, we use $N = 6$ for our simulations and we consider the lattice system to be recursive, that is base-pair -N is considered to be the same as base-pair N.

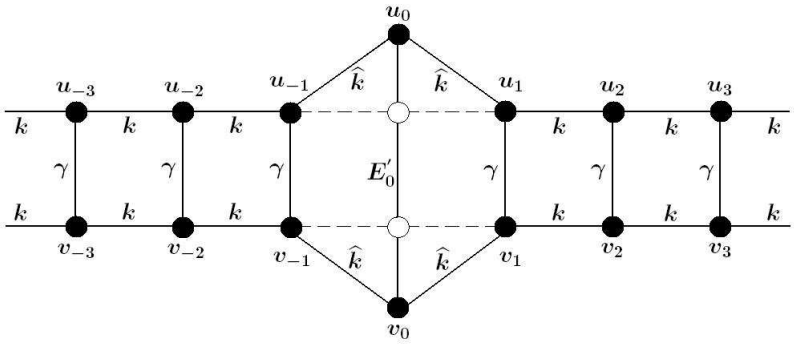

Figure 3.1: Illustration of the DNA model.

The energy associated with a breathing event is expressed by the following Hamiltonian [122]

$$H = \sum_n \frac{1}{2} m_n \left(\frac{du_n}{dt}\right)^2 + \frac{1}{2} m_n \left(\frac{dv_n}{dt}\right)^2 + \frac{1}{2} k^{(u)}_{n+\frac{1}{2}} (u_{n+1} - u_n)^2 + \\ + \frac{1}{2} k^{(v)}_{n+\frac{1}{2}} (v_{n+1} - v_n)^2 + \frac{1}{2} V_n (u_n - v_n), \tag{3.1.1}$$

where $u_n(t)$ and $v_n(t)$ denote the transverse displacements from equilibrium of the two chains. No longitudinal displacements are taken into consideration. Using the Hamiltonian we obtain the equations of motion

$$m_n \frac{d^2 u_n}{dt^2} = k^{(u)}_{n+\frac{1}{2}} (u_{n+1} - u_n) - k^{(u)}_{n-\frac{1}{2}} (u_n - u_{n-1}) - \frac{1}{2} F_n (u_n - v_n), \tag{3.1.2}$$

$$m_n \frac{d^2 v_n}{dt^2} = k^{(v)}_{n+\frac{1}{2}} (v_{n+1} - v_n) - k^{(v)}_{n-\frac{1}{2}} (v_n - v_{n-1}) + \frac{1}{2} F_n (u_n - v_n), \tag{3.1.3}$$

for the atoms on each chain of the double helix, where $F_n(y) = \frac{dV_n}{dy}(y)$. The model can be simplified, fully separating the equations, using the substitution $u_n = \frac{1}{2}(x_n + y_n)$ and $v_n = \frac{1}{2}(x_n - y_n)$ equivalent to $x_n = u_n + v_n$ and $y_n = u_n - v_n$. We also impose the condition that the spring constants in the two chains, at the same site, are the same, that is, $k^{(u)}_n = k^{(v)}_n = k_n$ for all n. The system becomes

$$m_n \frac{d^2 x_n}{dt^2} = k_{n+\frac{1}{2}} (x_{n+1} - x_n) - k_{n-\frac{1}{2}} (x_n - x_{n-1}), \tag{3.1.4}$$

$$m_n \frac{d^2 y_n}{dt^2} = k_{n+\frac{1}{2}} (y_{n+1} - y_n) - k_{n-\frac{1}{2}} (y_n - y_{n-1}) - F_n (y_n). \tag{3.1.5}$$

Furthermore, we can simplify our model by considering that all bases have approximately the same mass, as can be observed in Table 3.1. Given that a

nucleotide is composed of a nucleobase, a five-carbon sugar, and one to three phosphate groups, we may consider $m_n = m, \forall n$, where m represents the average value of nucleotides masses, that is, $m = 0.5098 \times 10^{-24}$ kg.

base	mass
Adenine (A)	0.2243×10^{-24} kg
Guanine (G)	0.2094×10^{-24} kg
Cytosine (C)	0.1845×10^{-24} kg
Thymine (T)	0.2509×10^{-24} kg
Diflourotoluene (F)	0.2125×10^{-24} kg

Table 3.1: Mass values for the five types of bases composing the nucleotides of our DNA sequence.

Moreover, we analyze a particular case of this system by removing the mass m from the equations and redefining the spring constant as follows: $k_{n+\frac{1}{2}} = m(k + k^{'}\delta_{n,0})$, where $\delta_{i,j}$ is the Kronecker delta function satisfying $\delta_{i,j} = 0$ if $i \neq j$ and $\delta_{i,j} = 1$ if $i = j$. In addition, we consider $V_n(y) = \frac{1}{2}m\gamma_n y^2$ for $n \neq 0$, with $\gamma_n = \gamma$ for all n, and $V_0(y) = mE_0(y)$, where E_0 is the energy function for the middle base-pair, which will be discussed later. We obtain a linear system of differential equations for x_n which can be solved analytically

$$\frac{d^2x_n}{dt^2} = k(x_{n+1} - 2x_n + x_{n-1}),\ \forall n \text{ with } |n| > 1, \tag{3.1.6}$$

$$\frac{d^2x_{-1}}{dt^2} = \widehat{k}(x_0 - x_{-1}) - k(x_{-1} - x_{-2}), \tag{3.1.7}$$

$$\frac{d^2x_0}{dt^2} = \widehat{k}(x_1 - 2x_0 + x_{-1}), \tag{3.1.8}$$

$$\frac{d^2x_1}{dt^2} = k(x_2 - x_1) - \widehat{k}(x_1 - x_0), \tag{3.1.9}$$

where $\widehat{k} = k + k^{'}$. Similarly, for y_n we have

$$\frac{d^2y_n}{dt^2} = k(y_{n+1} - 2y_n + y_{n-1}) - \gamma y_n,\ \forall n \text{ with } |n| > 1, \tag{3.1.10}$$

$$\frac{d^2y_{-1}}{dt^2} = \widehat{k}(y_0 - y_{-1}) - k(y_{-1} - y_{-2}) - \gamma y_{-1}, \tag{3.1.11}$$

$$\frac{d^2y_0}{dt^2} = \widehat{k}(y_1 - 2y_0 + y_{-1}) - \frac{dE_0}{dy}(y_0), \tag{3.1.12}$$

$$\frac{d^2y_1}{dt^2} = k(y_2 - y_1) - \widehat{k}(y_1 - y_0) - \gamma y_1. \tag{3.1.13}$$

The Hamiltonian which generates the latter system of equations is

$$H_y = \sum_n \left[\frac{1}{2} \left(\frac{dy_n}{dt} \right)^2 + \frac{1}{2} k (y_{n+1} - y_n)^2 + \frac{1}{2} \gamma y_n^2 \right] + E_0(y_0) \\ -\frac{1}{2} \gamma y_0^2 + \frac{1}{2} (\widehat{k} - k) \left[(y_1 - y_0)^2 + (y_0 - y_{-1})^2 \right]. \tag{3.1.14}$$

As it can be seen, except for $n = 0$, where our system of differential equations in y_n is nonlinear – see (3.1.12) – the inter-chains bonds are modelled by linear force-displacements relationship with coefficient γ.

3.2 Proposed model with white noise

A more realistic model of a natural process is obtained by allowing some randomness in the terms or coefficients of a differential equation [90]. Newton's second law of motion relates force to acceleration through a second-order differential equation.

Øksendal [90] analyzes equations like

$$dX/dt = b(t, X_t) + \sigma(t, X_t) \cdot W_t, \tag{3.2.1}$$

where W_t is a stochastic process that represents the noise term, which he considers to be the small Δt limit of the discrete equation $X_{i+1} = X_i + b(t_i, X_i)\Delta t_i + \sigma(t_i, X_i)\Delta B_i$, with $X_i = X(t_i)$ being a random variable, $\Delta t_i = t_{i+1} - t_i$ and $\Delta B_i = W_{t_i} \Delta t_i$, with B_t representing the Brownian motion, which is a stochastic process with stationary independent increments with mean zero and with continuous paths [90].

To solve (3.2.1) we have to choose between the Itô and Stratanovich integrals. The difference between the two methods is that Itô integrals are "not looking into the future", but if $\sigma(t, x)$ is a function that does not depend on x the two approaches are similar, as explained in [90]. In our case, $\sigma(t, x)$ is function independent of x and t and we will use the Itô integral to solve our system of equations.

Since we are interested in preserving the energy in our system, the stochastic differential equations used will contain parameters known as damping. Burrage et al. [19] analyze the stochastic differential equation

$$\ddot{x} = f(x) - \eta s^2(x)\dot{x} + \epsilon s(x)\xi(t), \tag{3.2.2}$$

which describes the position of a particle subject to a deterministic forcing $f(x)$, related to the potential function $V(x)$ by $f(x) = -V'(x)$, and a random forcing $\xi(t)$ such that $\left\langle \xi(t)\xi(t') \right\rangle = \delta(t - t')$. The damping term is η, while ϵ is the amplitude of the random forcing. Note that the noise coefficient ϵ is related to the damping coefficient η by the fluctuation-dissipation relation, which will be introduced in Section 3.3 and discussed in details in Section 3.4. Equation (3.2.2) can be rewritten as

$$dX_t = V_t dt, \tag{3.2.3}$$

$$dV_t = -\eta s^2(X_t)V_t dt + f(X_t)dt + \epsilon s(X_t)dW_t, \tag{3.2.4}$$

which shows that the noise term directly influences the velocity and only indirectly the displacement.

For $s(x) = 1, \forall x \in \mathbb{R}$, numerical analysis of equations (3.2.3)–(3.2.4), given in [19], shows that several integration methods can be used to obtain their solution, for example the forward Euler method, Heun's method or leapfrog method, but the best results are obtained using the implicit midpoint method. As already stated, we have no need to let s depend on x, so we simply take $s(x) = 1, \forall x$. Hence, we use the implicit midpoint method for the numerical simulations presented in Chapter 5.

Taking into consideration the above observations, we add noise and damping terms in the system of equations (3.1.10)–(3.1.13) to obtain

$$\frac{d^2y_n}{dt^2} = k(y_{n+1} - 2y_n + y_{n-1}) - \gamma y_n - \eta\frac{dy_n}{dt} + \epsilon\xi_n(t), \ \forall |n| > 1, \tag{3.2.5}$$

$$\frac{d^2y_{-1}}{dt^2} = \widehat{k}(y_0 - y_{-1}) - k(y_{-1} - y_{-2}) - \gamma y_{-1} - \eta\frac{dy_{-1}}{dt} + \epsilon\xi_{-1}(t), \tag{3.2.6}$$

$$\frac{d^2y_0}{dt^2} = \widehat{k}(y_1 - 2y_0 + y_{-1}) - \frac{dE_0}{dy}(y_0) - \eta_0\frac{dy_0}{dt} + \epsilon\xi_0(t), \tag{3.2.7}$$

$$\frac{d^2y_1}{dt^2} = k(y_2 - y_1) - \widehat{k}(y_1 - y_0) - \gamma y_1 - \eta\frac{dy_1}{dt} + \epsilon\xi_1(t), \tag{3.2.8}$$

The random forcing in our system $\xi_n(t)$ can be represented as a generalized stochastic process called *white noise* [61], in which $\xi_n(t) = dB_n(t)$ and $B_n(t)$ is continuous in time. Since we are primarily concerned with simulations, we work with the discrete-time version of the system, which suggests the replacement of ξ_n by proper stochastic processes. We apply the Itô integrals theory to solve the system of stochastic differential equations (3.2.5)-(3.2.8) and discretising we replace $y_n(t)$ with $y_n = y_n(t_i)$, where $t_i = i\Delta t$, and hence obtain

$$
\begin{aligned}
(3.2.9)\quad & y_n^i = y_n^{i-1} + v_n^{i-1}\Delta t_i, \ \forall n \text{ with } |n| > 1, \\
(3.2.10)\quad & v_n^i = v_n^{i-1} + (k(y_{n+1}^{i-1} - 2y_n^{i-1} + y_{n-1}^{i-1}) - \gamma y_n^{i-1})\Delta t_i - \\
& \quad -\eta v_n^{i-1}\Delta t_i + \epsilon \Delta B_n^i, \ \forall n \ with \ |n| > 1, \\
(3.2.11)\quad & y_{-1}^i = y_{-1}^{i-1} + v_{-1}^{i-1}\Delta t_i, \\
(3.2.12)\quad & v_{-1}^i = v_{-1}^{i-1} + (\widehat{k}(y_0^{i-1} - y_{-1}^{i-1}) - k(y_{-1}^{i-1} - y_{-2}^{i-1}) \\
& \quad -\gamma y_{-1}^{i-1})\Delta t_i - \eta v_{-1}^{i-1}\Delta t_i + \epsilon \Delta B_{-1}^i, \\
(3.2.13)\quad & y_0^i = y_0^{i-1} + v_0^{i-1}\Delta t_i, \\
(3.2.14)\quad & v_0^i = v_0^{i-1} + (\widehat{k}(y_1^{i-1} - 2y_0^{i-1} + y_{-1}^{i-1}) - \frac{dE_0}{dy}(y_0^{i-1}))\Delta t_i - \\
& \quad -\eta v_0^{i-1}\Delta t_i + \epsilon \Delta B_0^i, \\
(3.2.15)\quad & y_1^i = y_1^{i-1} + v_1^{i-1}\Delta t_i, \\
(3.2.16)\quad & v_1^i = v_1^{i-1} + (k(y_2^{i-1} - y_1^{i-1}) - \widehat{k}(y_1^{i-1} - y_0^{i-1}) \\
& \quad -\gamma y_1^{i-1})\Delta t_i - \eta v_1^{i-1}\Delta t_i + \epsilon \Delta B_1^i.
\end{aligned}
$$

Here, for each time step i and each lattice site n, ΔB_n^i is an independent normally distributed random variable with zero mean and standard deviation $\sqrt{\Delta t_i}$.

3.3 Parameter fitting

The system of equations (3.2.9)-(3.2.16) contains several terms and coefficients, namely η, ϵ, k, $\widehat{k}$, γ and the energy function $E_0(y_0)$, whose values influence the system solution. For this reason, their values have to be chosen carefully so that our model behaves in a similar manner to the experimentally observed systems and all-atom molecular dynamics (MD) simulations.

The system's temperature, $\widetilde{T}$, is related to η and ϵ by a fluctuation-dissipation relation, which is defined as $\epsilon^2 = 2\eta k_B \widetilde{T}$, where $k_B = 1.38 \times 10^{-23}$ JK^{-1} is Boltzmann's constant (see [51] for details). In our case, the temperature is $T = 293$ K and hence $k_B\widetilde{T} = 4.1 \times 10^{-21}$ J [121].

Note that, before introducing noise and damping in our system, we have divided each equation by the mass of a nucleotide, that is, $m = 0.5098 \times 10^{-24}$ kg. We also consider $\epsilon = \widetilde{\epsilon}/m$ and $\eta = \widetilde{\eta}/m$, which implies $k_B T = k_B\widetilde{T}/m$, that is $k_B T = 0.8125$ Å^{2}ps^{-2}.

Next, we observe that the value of η is not directly fitted to AMBER data. On the contrary, it is based on the fitted value of ϵ, and then η is computed using the fluctuation-dissipation relation.

Most papers in the literature assume that all the along-chain interactions are identical – see [76] for example – and assume that defects only influence the coupling between the two chains (γ and E_0). Our model enables us to test the effects of defects in the along-chain interactions, for example, $k = \widehat{k}$ and later results suggest, that for the diflourotoluene base $\widehat{k} < k$, hence we treat k and $\widehat{k}$ as two distinct parameters.

Using data from AMBER simulations, it is possible to determine the form of the force-distance relationship for the interchain separations and the associated energy function, known as "potential" of mean force (PMF), which can be used to determine $E_0(y_0)$. The standard procedure is as follows:

- determine the minimum min and the maximum max displacements from some reference distance between the bases of the breathing pair, for example, for a 30° twisted DNA sequence we typically have $min = -0.5998$ Å and $max = 5.1437$ Å;
- split the interval $[min, max]$ into several bins of equal size s (typically 20, of size 0.3 Å, but possibly 6-600 of size 0.01-1 Å);
- let f_{tot} be the total number of data points available for base-pair opening distances;
- from the base-pair opening distances represented in the AMBER data, count the frequency f_i of each bin;

- as a first approximation, we have that for each bin i the corresponding value for the $PMF(y_0)$ is $-k_B T log(f_i/f_{tot})$;
- use spline interpolation to determine an expression for the potential of mean force $PMF(y_0)$, as illustrated in Figure 3.2.

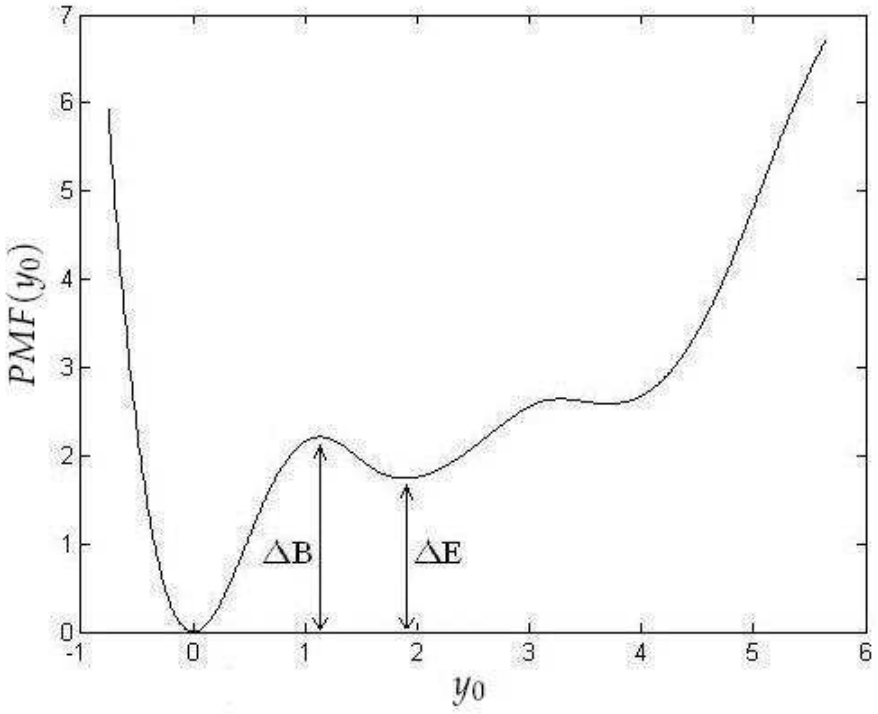

Figure 3.2: Illustration of the potential of mean force $PMF(y_0)$ (Å^2 ps^{-2}) as a function of displacement y_0 (Å), for a bin size of $s = 0.5$ and a twist of 30°.

Figure 3.2 defines two important quantities for the potential of mean force, that is, the breathing barrier ΔB and the energy difference ΔE between the breathing and the normal states. The size s of the bins significantly influences the expression of the energy function. The number of bins N_{bin} depends on s. In fact we have $N_{bin} = [(max - min)/s] + 1$, which means that for $s = 0.1$ we have $N_{bin} = 58$, while for $s = 0.5$ we have $N_{bin} = 12$, for example. We have tested a wide range of bin sizes from $s = 0.01$ up to $s = 1$ to investigate the effect of s on $PMF(y_0)$. Figure 3.3 emphasizes the difference between the small and large bin sizes. As can be seen, there are significant changes in ΔB and ΔE values when we vary the bin size s from a value of 0.1, as in Figure 3.3(a), to a value of 0.9, as in Figure 3.3(b).

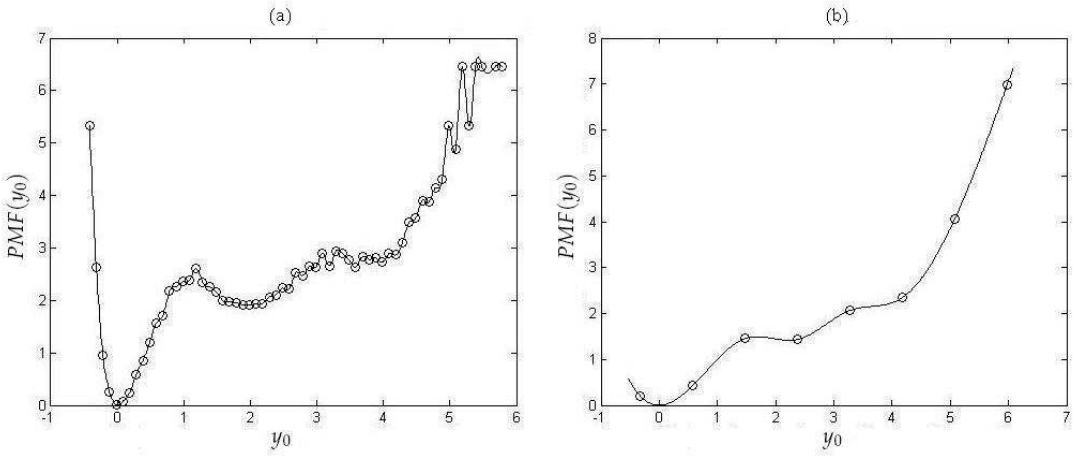

Figure 3.3: Illustration of the potential of mean force $PMF(y_0)$ (Å^2 ps^{-2}) as a function of displacement y_0 (Å), for a twist of 30° and a bin size of (a) $s = 0.1$ and (b) $s = 0.9$. The small circles represent the bin points.

In Figure 3.4, we illustrate how the barrier ΔB for the base-pair breathing varies when the bin size is changed.

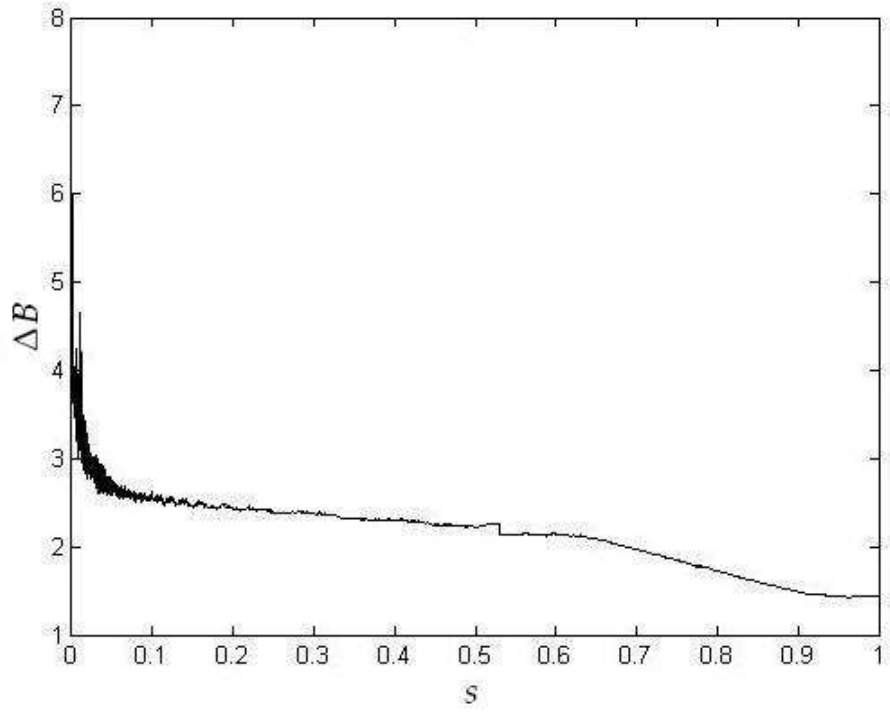

Figure 3.4: Illustration of the breathing barrier ΔB (Å^2 ps^{-2}) against the bin size (Å) – see Figure 3.2 for the definition of ΔB.

This analysis shows that s should not take values below 0.2 or above 0.5, since in such cases the barrier variation with bin size is too large. In addition, when s is too large the bins are so coarse-grained that the barrier is not resolved at all, leading to underestimates of ΔB, whereas when s is very small, there are so few plane paths in each bin that ΔB varies wildly with s.

In what follows, the system parameters are fitted to data obtained from the molecular dynamics (MD) package AMBER using the maximum likelihood method.

3.3.1 The maximum likelihood method

We use the maximum likelihood method (MLE) to determine k, $\widehat{k}$, γ, $E_0(y_0)$, ϵ and implicitly η, since they are correlated by the fluctuation-dissipation relation, using data obtained during AMBER simulations. Note that the time step in AMBER simulations is constant, thus $\Delta t_i = \Delta t$, $\forall i$.

Taking into account that the nonlinearity of the system is generated by the breathing pair, we will first apply MLE method for y_1, which involves only linear terms in y_0, y_1, v_1 and y_2, to obtain the parameters k, $\widehat{k}$, γ and ϵ.

From the system of equations (3.2.9)-(3.2.16), we have that the speed is normally distributed, thus $v_1^{i+1} \approx N(\mu_{i+1}, \sigma^2)$, with

$$\mu_i = v_1^{i-1} - (\eta v_1^{i-1} + \gamma y_1^{i-1} - k(y_2^{i-1} - y_1^{i-1}) + \widehat{k}(y_1^{i-1} - y_0^{i-1}))\Delta t \tag{3.3.1}$$

and $\sigma^2 = \epsilon^2 \Delta t$. This implies that the log-likelihood is

$$\begin{aligned} l_1(\epsilon, k, \widehat{k}, \gamma) &= \log(L_1(\epsilon, k, \widehat{k}, \gamma)) \\ &= -\frac{n}{2}\log(\sigma^2) - \frac{1}{2\sigma^2}\sum_{i=1}^{n}(v_1^i - \mu_i)^2 \\ &= -\frac{n}{2}\log(\epsilon^2\Delta t) - \frac{1}{2\epsilon^2\Delta t}\sum_{i=1}^{n}[v_1^i - v_1^{i-1} + (\eta v_1^{i-1} \\ &\quad + \gamma y_1^{i-1} - k(y_2^{i-1} - y_1^{i-1}) + \widehat{k}(y_1^{i-1} - y_0^{i-1}))\Delta t]^2. \end{aligned} \tag{3.3.2}$$

After computing the parameters values for which the likelihood function is maximum, we compute the 95% confidence intervals for them, in order to determine the permitted ranges for each parameters.

Let θ be a column vector of q parameters. We denote the information (a $q \times q$ matrix) [56] by

$$I(\theta)_{ij} = \left(\mathbb{E}_x \left[-\frac{\partial^2 l_1}{\partial \theta_i \partial \theta_j}(\theta) \right] \right), \tag{3.3.3}$$

where x is a vector of data and $1 \leq i, j \leq q$. Then the estimate of θ using MLE method is given by $\widehat{\theta} \approx N(\theta, \beta)$, where β contains the elements of the main diagonal of $I^{-1}(\theta)$. Instead of I, we can use the observed information $I_{obs}(\theta) = H(\theta)$, where H is the Hessian matrix of l_1 (3.3.2) and the variance of θ_i will then be $(I_{obs}^{-1}(\theta))_{ii}$.

Finally, the 95% confidence interval for $\widehat{\theta_i}$ is

$$\left[\widehat{\theta_i} - 1.96\sqrt{(I_{obs}^{-1}(\widehat{\theta}))_{ii}}, \widehat{\theta_i} + 1.96\sqrt{(I_{obs}^{-1}(\widehat{\theta}))_{ii}} \right]. \tag{3.3.4}$$

Maximizing l_1, using data obtained from AMBER simulations, we determine values of the parameters k, $\widehat{k}$, γ and ϵ of our system. Notice that maximizing the likelihood function is equivalent to finding values of k, $\widehat{k}$, γ and ϵ, for which the partial derivatives of the function l_1 vanish, i.e.

$$\frac{\partial l_1}{\partial k}(\theta) = 0, \tag{3.3.5}$$

$$\frac{\partial l_1}{\partial \widehat{k}}(\theta) = 0, \tag{3.3.6}$$

$$\frac{\partial l_1}{\partial \gamma}(\theta) = 0, \tag{3.3.7}$$

$$\frac{\partial l_1}{\partial \epsilon}(\theta) = 0, \tag{3.3.8}$$

where $\theta = (k, \widehat{k}, \gamma, \epsilon)$.

Note that in some cases the parameter values might be highly correlated and we need to compute the confidence region [111] based on the probability density function for θ, that is,

$$f(\theta) = \frac{1}{(2\pi)^{q/2} \det(I_{obs})^{1/2}} e^{-\frac{1}{2}(\theta - \widehat{\theta})^T I_{obs}^{-1}(\theta - \widehat{\theta})}, \tag{3.3.9}$$

where $\det(I_{obs})$ represents the determinant of I_{obs}, while $(\theta - \widehat{\theta})^T$ is the transpose of $(\theta - \widehat{\theta})$.

Next, we consider

$$\rho^2 = (\theta - \widehat{\theta})^T I_{obs}^{-1} (\theta - \widehat{\theta}) \tag{3.3.10}$$

and for each $0 \leq \alpha \leq 1$ we define l_α such that

$$\int_{\rho^2 \leq l_\alpha^2} f(\theta) d\theta_1 ... d\theta_q = 1 - \alpha. \tag{3.3.11}$$

Then, the ellipsoid defined by $\rho^2 \leq l_\alpha^2$ represents the $100(1-\alpha)\%$ confidence region. However, we will use, for our SDE simulations, parameter values close to the center $\widehat{\theta}$ of the ellipsoid and therefore, to simplify the parameter analysis, we will discuss the rectangular confidence regions obtained using (3.3.4).

In what follows, we will present the results obtained for a 30° undertwisted DNA, containing 12 base-pairs. First we applied the MLE method using information taken each $\Delta t = 1$ps and we obtained $k = 0.0028$, $\widehat{k} = 0.0033$, $\gamma = 0.0646$ and $\epsilon = 0.5367$, with negative values at the beginning of the confidence intervals for k and $\widehat{k}$. This shows that the method used to fit our parameters is not particularly accurate, or that we could use larger confidence intervals.

Moreover, there were significant differences between the SDE simulation and AMBER results. At the defect site, for example, the expected breathing frequency was different for the two cases. For the defect site neighbouring base-pairs we have important differences in average displacement from equilibrium – see Figure 3.5. Note the different scale on the vertical axis for $y_1(t)$. The cause of the differences is the integration step: for AMBER simulation a $\Delta t = 2$ fs time step was used, while for the MLE method only information taken each $\Delta t = 1$ ps was used. When the correct value of $\Delta t = 2$ fs is used, an agreement between AMBER and SDE simulations is achieved, as we will show in Chapter 5.

These results show that we need all the intermediary data that AMBER generates while simulating the system. We cannot store all this data because we would need more than 8000 GB to represent 20 ns and it is impossible to achieve this goal using existing computers and servers. Thus, we have to use only some parts of the simulation for the MLE methods, that is, data representing 2 ns which requires only 800 GB.

For $\Delta t = 2$ fs, as assumed in the AMBER simulations, the MLE method applied for l_1 gives the following confidence intervals:

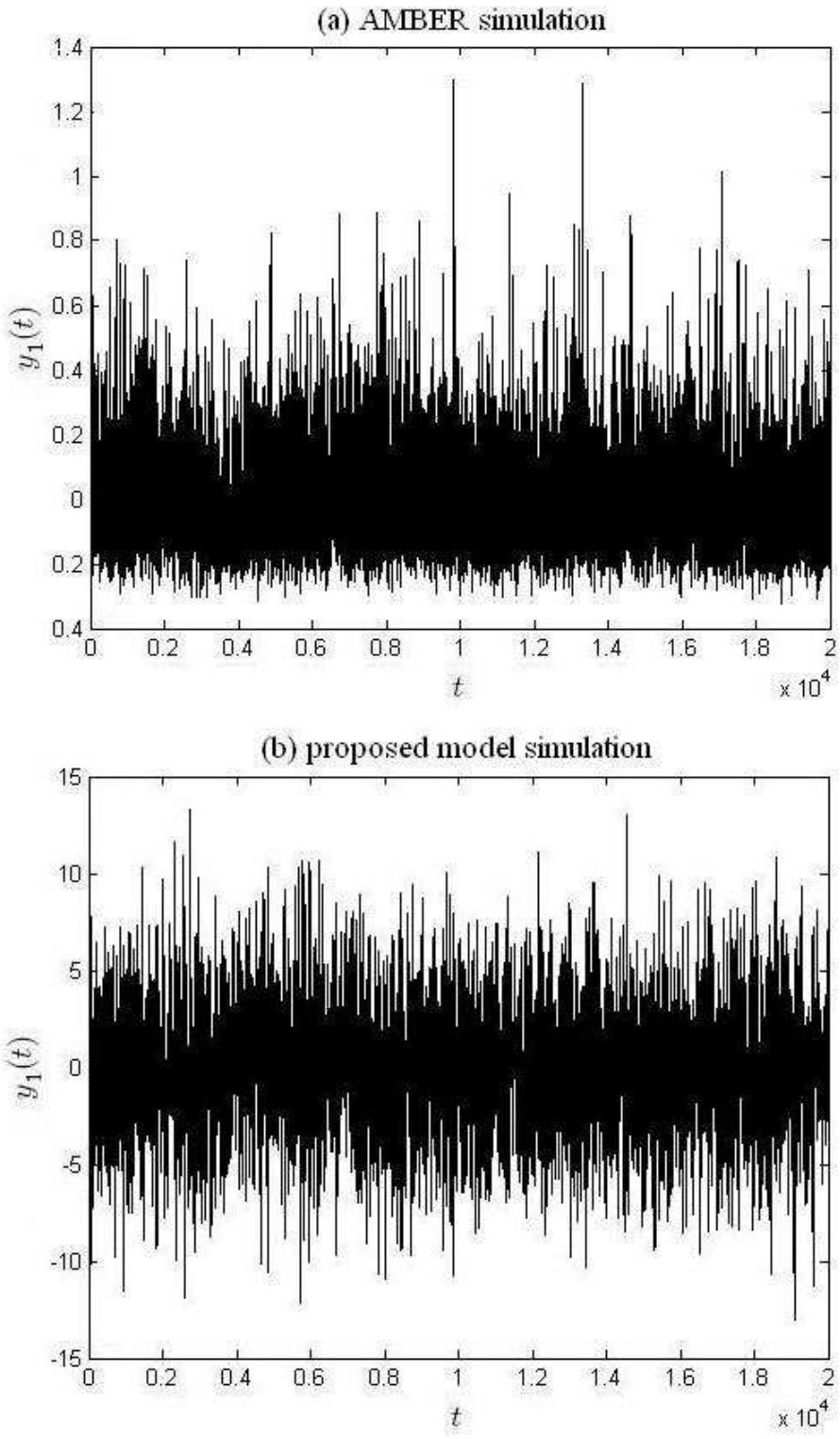

Figure 3.5: Illustration of the variation of the distance (measured in Å) between the bases of the nonbreathing pair, obtained using (a) AMBER and (b) the proposed model, for 20ns for a 30° undertwisted DNA.

- $6.3571 \leq k \leq 8.8856$;
- $1.7848 \leq \widehat{k} \leq 2.0731$;
- $121.4107 \leq \gamma \leq 124.5864$;
- $3.3897 \leq \epsilon \leq 3.3991$.

3.3.2 MLE method for $E_0(y_0)$

We can apply the MLE method for the breathing pair to obtain a more accurate estimation of $E_0(y_0)$. Note that the system we considered above had the same noise coefficient for all base-pairs, but our computations show that y_0 requires a larger noise amplitude than for the others ($y_{\pm n}$, with $n > 0$). Hence, we introduce new parameters are ϵ_0 and η_0, where the damping coefficient value for the breathing pair is also determined by the fluctuation-dissipation relation, namely $\eta_0 = \epsilon_0^2/2k_BT$.

Taking into account our need of about 15 interpolation points to estimate E_0, the search interval belongs to $\mathbb{R}^{17}$ (taking into account $\widehat{k}$ and ϵ_0). Hence, considering $\widehat{k}$ to be fixed by using its value obtained from the l_1 maximization reduces the search interval on $\mathbb{R}^{16}$ and we can obtain a more accurate expression for $E_0(y_0)$. In what follows we consider $\widehat{k} = 1.9289$.

From (3.2.14), we have that $v_0^{i+1} \approx N(\mu_{i+1}, \sigma^2)$, with

$$\mu_i = v_0^{i-1} - (\eta_0 v_0^{i-1} + \frac{dE_0}{dy}(y_0^{i-1}) - \widehat{k}(y_1^{i-1} - 2y_0^{i-1} + y_{-1}^{i-1}))\Delta t \tag{3.3.12}$$

and $\sigma^2 = \epsilon_0^2 \Delta t$. This implies that the log-likelihood is

$$\begin{aligned} l_0(E_0, \epsilon_0) &= \log(L_0(E_0, \epsilon_0)) \\ &= -\frac{n}{2}\log(\sigma^2) - \frac{1}{2\sigma^2}\sum_{i=1}^{n}(v_0^i - \mu_i)^2 \\ &= -\frac{n}{2}\log(\epsilon_0^2\Delta t) - \frac{1}{2\epsilon_0^2\Delta t}\sum_{i=1}^{n}[v_0^i - v_0^{i-1} + (\eta_0 v_0^{i-1} \\ &\quad + \frac{dE_0}{dy}(y_0^{i-1}) - \widehat{k}(y_1^{i-1} - 2y_0^{i-1} + y_{-1}^{i-1}))\Delta t]^2. \end{aligned} \tag{3.3.13}$$

Note that E_0 is represented as a vector of pairs (x_i, y_i), with x (an increasing array) representing the bins and y the value of the free energy for each bin. The final expression of $E_0(y_0)$ is obtained using a cubic spline approximation. During the maximization only the values of y will be modified. Due to this, we can only compute numerically the confidence intervals, since the partial derivatives with respect to the E_0 components (needed for the Hessian matrix) cannot be computed analytically.

Applying MLE method for l_0 we obtain $\epsilon_0 = 5.5160$, while E_0 is represented in Figure 3.6. As we observe, the expression of $E_0(y_0)$ after applying MLE method is surprising. Figure 3.2, in which we have a representation of $PMF(y_0)$ obtained from AMBER data using a bins count, suggests that the equilibrium state is around 0Å, while the expression of $E_0(y_0)$, fitted to AMBER data, suggests that the closed state is a lower energy configuration than the open state.

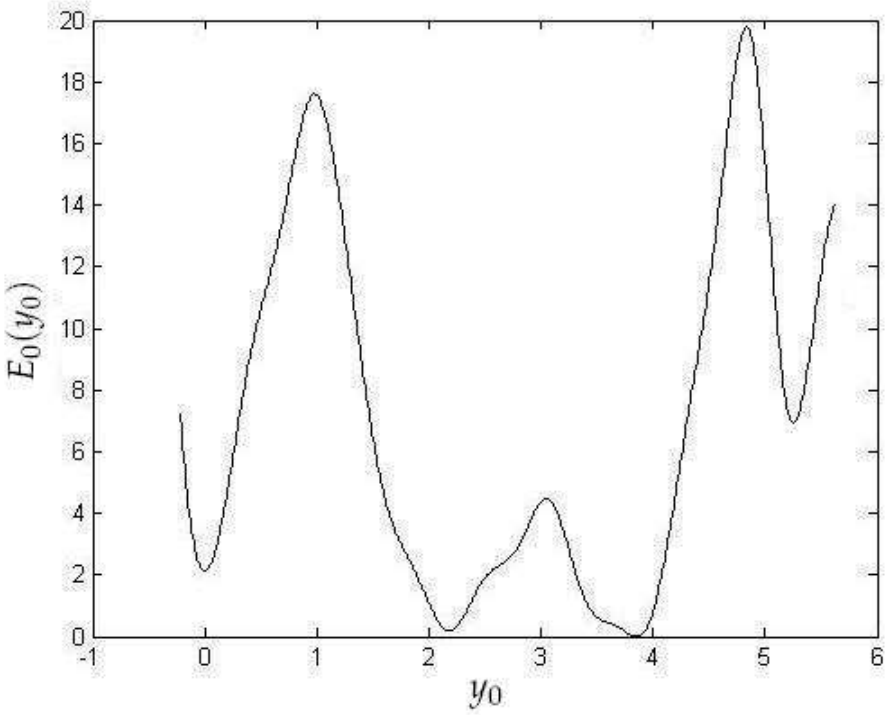

Figure 3.6: Illustration of E_0 function (Å^2 ps^{-2}), obtained using the MLE method for a 30° undertwisted DNA.

In the SDE system, the deterministic force acting on the breathing pair has two components:

1. the along-chain force: $\widehat{k}(y_1 - 2y_0 + y_{-1})$;
2. the inter-chain force: $-\dfrac{dE_0}{dy}(y_0)$.

Using bin counts of the AMBER data, we compute the so called "potential of mean force" $PMF(y_0)$, which includes all the deterministic forces in our system, while the MLE method considers $E_0(y_0)$ to be the energy specific to the inter-chain interactions. From (3.1.14) we have the total potential energy corresponding to the breathing pair being $E_0(y_0) + \frac{1}{4}\widehat{k}((y_1 - y_0)^2 + (y_0 - y_{-1})^2)$. If we take into account the fact that the neighbouring pairs do not breath and have only small deviations from equilibrium, we have that $\langle y_1 \rangle = \langle y_{-1} \rangle = 0$, $\langle y_{-1}^2 \rangle \ll \langle y_0^2 \rangle$ and $\langle y_1^2 \rangle \ll \langle y_0^2 \rangle$, which implies that the total potential energy is approximately $E_0(y_0) + \frac{1}{2}\widehat{k}y_0^2$.

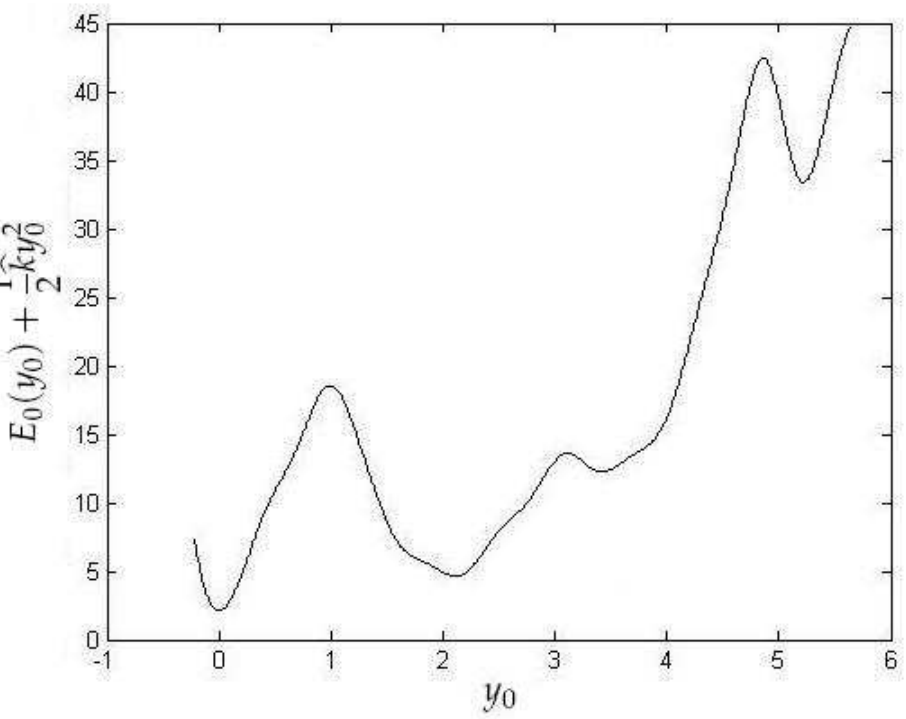

Figure 3.7: Illustration of potential energy function (Å^2 ps^{-2}) of the breathing pair, specific to the SDE system, for a 30° undertwisted DNA.

Figure 3.7 shows that a graph of the total potential energy of our SDE system is close to the potential of mean force displayed in Figure 3.2. In fact, the two representations differ only by a constant. This means that we can approximate

the potential of mean force of our SDE system using

$$PMF(y_0) = E_0(y_0) + \frac{1}{2}\widehat{k}y_0^2 \tag{3.3.14}$$

Even with results more accurate than in the previous case (Figure 3.5), the simulation of the system using the proposed model proved that the parameters have not been fitted correctly and no breathing or only very rare and short breathing events were obtained. As can be observed, the breathing barrier ΔB from Figure 3.7 is very high and this could be one of the reasons for which the breathing events are so rare.

3.3.3 Improving $E_0(y_0)$ estimation

The method needs to be improved so that the central base-pair crosses the breathing barrier more often. One of the possible methods is to approximate E_0 using smooth splines instead of simple splines.

As already mentioned, E_0 is represented as an array of points (x_i, y_i). The spline approximation of E_0 is obtained by computing

$$\min_{(E_0)_i} \left\{ \sum_i (E_{0i} - y_i)^2 + \lambda \int_{y_{min}}^{y_{max}} \left[\frac{d^2 E_0}{dy^2}(x) \right]^2 dx \right\}, \tag{3.3.15}$$

where $\lambda \geq 0$ and $[y_{min}, y_{max}]$ is the domain of definition of E_0. After obtaining $(E_0)_i$ we apply the spline approximation to compute E_0 for the new pairs of points (x_i, E_{0i}).

Note that for large values of λ we obtain a straight line, given by the least squares approximation to the data, while for $\lambda = 0$ the minimum obtained is the standard cubic spline approximation through the points (x_i, y_i). For $0 < \lambda < \infty$ a curve somewhere between these two extremes is obtained. This means that choosing the correct value of λ is an important task for this method.

In our case, we have $y_{min} = -0.5998$ and $y_{max} = 5.1437$. For $\lambda = 0.01$ we obtain $\epsilon_0 = 5.5131$ and E_0 is represented in Figure 3.8. The circles represent the actual points through which the cubic spline approximation normally pases, while

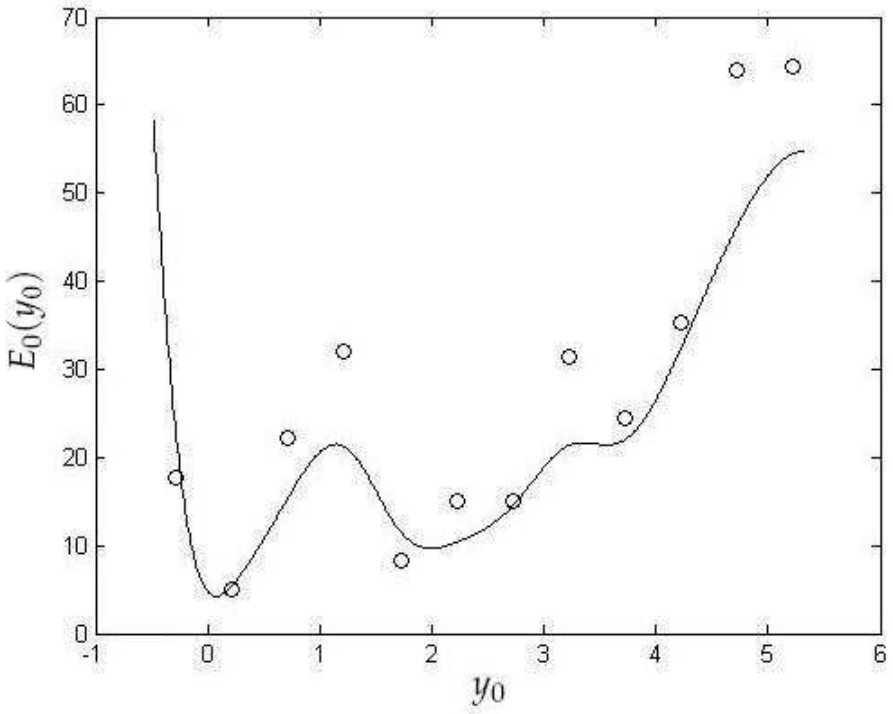

Figure 3.8: Illustration of E_0 function (Å^2 ps^{-2}), obtained using the MLE method and smooth splines (3.3.15) for a 30° undertwisted DNA, using $\lambda = 0.01$. The small circles describe the values for the centres of the bins, used to compute the smooth splines approximation.

the line shows what the smooth approximation is. Again only rare and short breathing events could be observed, since the breathing barrier is too high.

The method uses y_i for MLE application and then constructs a spline which does not necessary pass through (x_i, y_i) points, only close to them. The points through which the spline actually passes are (x_i, E_{0i}).

Another way to improve the MLE results is adding a penalty term to the l_0 expression. More precisely to use

$$
\begin{aligned}
(3.3.16)\quad l_0^p(E_0, \epsilon_0) &= l_0(E_0, \epsilon_0) - P \\
&= -\frac{n}{2}\log(\epsilon_0^2 \Delta t) - \frac{1}{2\epsilon_0^2 \Delta t}\sum_{i=1}^{n}[v_0^i - v_0^{i-1} \\
&\quad +(\eta v_0^{i-1} + \frac{dE_0}{dy}(y_0^{i-1}) - \widehat{k}(y_1^{i-1} - 2y_0^{i-1} + y_{-1}^{i-1}))\Delta t]^2 \\
&\quad -\alpha \int_{y_{min}}^{y_{max}} \left[\frac{d^2E_0}{dy^2}(x)\right]^2 dx,
\end{aligned}
$$

where the penalty term is

$$P = \alpha \int_{y_{min}}^{y_{max}} \left[\frac{d^2 E_0}{dy^2}(x) \right]^2 dx, \tag{3.3.17}$$

with $\alpha \geq 0$ and $[y_{min}, y_{max}]$ is the range of values that y_0 can take. Dealing with the same function as above, we use $y_{min} = -0.5998$ and $y_{max} = 5.1437$.

The MLE method with a penalty term is different from the previous one. The penalty is inside the MLE function, which produces points $(x_i, \bar{y}_i)$, hence the penalty term influences directly the MLE result. Moreover, we use a cubic spline approximation, which passes through the points $(x_i, \bar{y}_i)$, to determine the expression of $E_0(y_0)$. In the previous case we determine first the points (x_i, y_i) and then we find $(E_0)_i$, which minimizes (3.3.15).

The penalty term helps reduce the range of y parameter values explored and depends on the value of α. We use for α a value equal to

$$\alpha_0 = \left| \frac{l_0(E_0, \epsilon_0)}{\int_{min}^{max} [E_0''(x)]^2 dx} \right|, \tag{3.3.18}$$

while with lower values we get the final result closer to the initial guess of E_0, obtained from AMBER data using a bins count. For $\alpha \gg \alpha_0$ the approximation is close to a straight line. Using $\alpha = 0.2413$ we have $\epsilon_0 = 5.5402$ and E_0 is represented in Figure 3.9. Again, a simulation of the resulting SDE system shows that the breathing events are not as frequent as in the original AMBER simulation.

Note that it is also possible to combine the MLE with penalty term method and approximate $E_0(y_0)$ using smooth spline, which helps to further decrease the breathing barrier ΔB. But lowering the barrier did not generate longer and more frequent breathing events, hence we conclude that one of the other parameters ϵ_0, η_0, ϵ, η, k, $\widehat{k}$ and γ is the cause of the differences between AMBER simulation and the proposed model. The quantities ϵ, η, k, $\widehat{k}$ and γ were fitted independently of the breathing pair, thus they can not generate the error.

The random movement of the breathing base-pair is generated by ϵ_0. No matter which method we used its value was around 5.5, which suggests that this parameter was fitted correctly. In conclusion the damping coefficient η_0 is the one

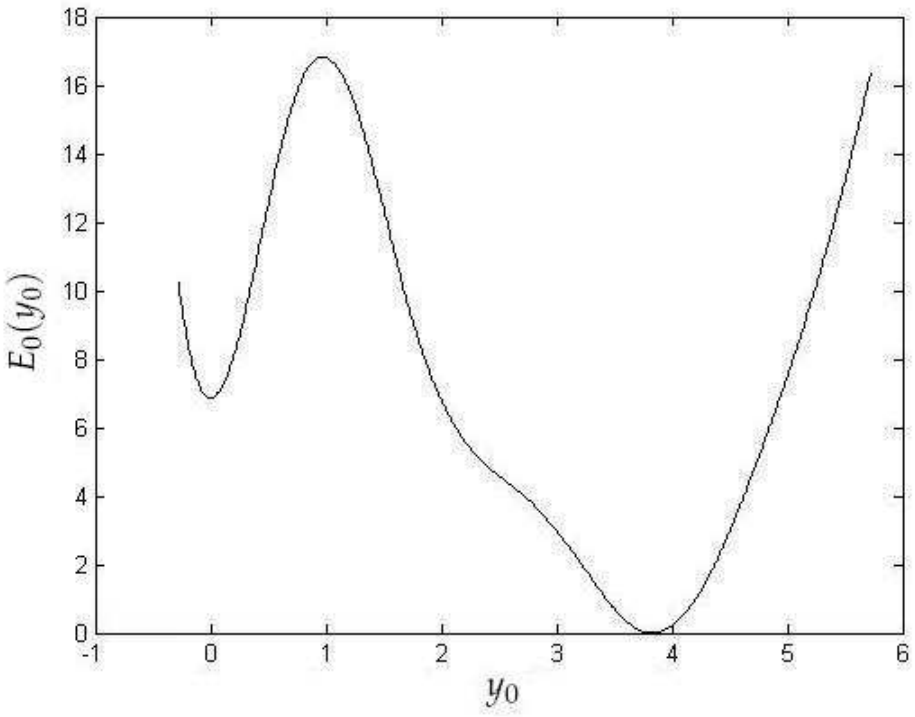

Figure 3.9: Illustration of E_0 function (Å^2 ps^{-2}), obtained using the MLE method with a penalty term, with $\alpha = 0.2413$, for a 30° under-twisted DNA.

for which a wrong value is used. This parameter was computed based on the fluctuation-dissipation relation using the fitted value of ϵ_0, which means that this relation, defined in general for a particle subject to deterministic forcing, cannot be applied in our case. We will revisit this in Section 3.4.

3.3.4 An improved potential of mean force

Figure 3.7 represents what we consider to be an approximation of the "potential of mean force" of our SDE system, as described in Section 3.3.2. But, the inter-chain force and the along-chain force are not the only deterministic forces in our system. The damping term also contributes as a deterministic force to the system, since the coefficient is constant (not stochastic), being related only to the noise amplitude and not to the noise term itself.

Consider the simple case of a moving particle subject to both, deterministic and

nondeterministic forces, with the equation of motion given by

$$\frac{d^2x}{dt^2} = -kx - \eta\frac{dx}{dt} + \epsilon\xi(t). \tag{3.3.19}$$

Then the associated energy is $E(x) = K(x) + U(x)$, where $K(x) = \frac{1}{2}(dx/dt)^2$ is the kinetic energy and $U(x)$ is the potential energy. Using the fact that $\frac{d^2x}{dt^2} = -\frac{\partial U}{\partial x}$, we obtain that

$$U(x) = \frac{1}{2}kx^2 + \eta\int\frac{dx}{dt}dx. \tag{3.3.20}$$

If we take $\eta \ll 1$ and $\epsilon \ll 1$ we can consider that $E(x) = E_1$ fixed and then $x(t)$ is periodic, with $(dx/dt)^2 = 2E_1 - kx^2$. Using this value for E and integrating we obtain

$$\begin{aligned}\int\frac{dx}{dt}dx &= \pm\int\sqrt{2E_1 - kx^2}dx \\ &= \pm\int\left[\sin^{-1}\left(x\sqrt{\frac{k}{2E_1}}\right) + x\sqrt{\frac{k}{2E_1}\left(1 - \frac{kx^2}{2E_1}\right)}\right]\frac{E_1}{\sqrt{k}}dx \\ &\approx x\sqrt{2E_1}\left(1 - \frac{kx^2}{12E(x)}\right),\end{aligned} \tag{3.3.21}$$

the approximation being for small x. Since, during AMBER simulations, the energy is preserved at $E(x) \approx \frac{1}{2}k_BT$, we obtain the leading order result $U(x) = \frac{1}{2}kx^2 + \eta x\sqrt{k_BT}$.

Hence, an approximation of the potential of mean force in our SDE system is actually given by

$$PMF(y_0) = E_0(y_0) + \frac{1}{2}\widehat{k}y_0^2 + \sqrt{k_BT}\eta_0 y_0. \tag{3.3.22}$$

Figure 3.10 represents an approximation of the actual potential of mean force of our SDE system (containing the damping contribution), obtained using $E_0(y_0)$ expression from Figure 3.6. This representation clarifies the reasons for which no breathing events were obtained when simulating the SDE system: the damping term is large enough to overcome the noise term and to keep the system in its minimum energy state at any moment.

Hence, we need either to reconstruct the system from scratch or to reconsider the fluctuation-dissipation relation, as already suggested. Our analysis shows that redefining the relation between the noise and damping coefficient solves the inconsistencies between our mesoscopic model and AMBER.

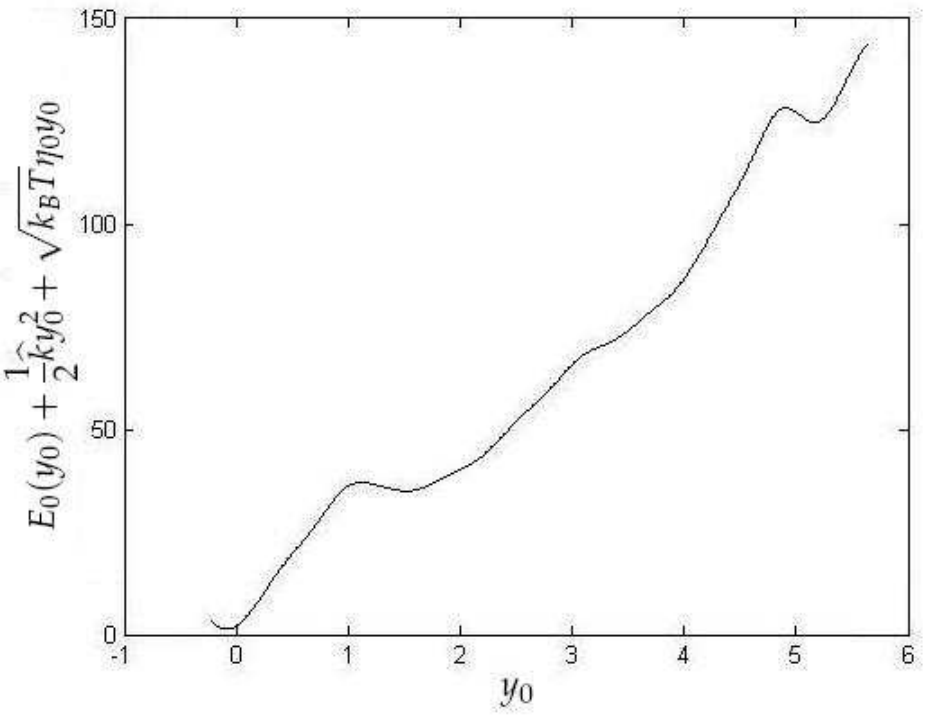

Figure 3.10: Illustration of potential energy function (Å^2 ps^{-2}) of the breathing pair, including the damping contribution, specific to the SDE system, for a 30° undertwisted DNA.

3.4 Fluctuation-dissipation relation

Note that each equation of the SDE system is obtained using a change of variables from two other equations (in u_n and v_n, respectively) and that we have added the noise and damping to the deterministic equation for y_n. However, the distance y_n represents the distance between two bases moving independently one of each other and each subject to random fluctuations. The intra- and inter-chain potential energies describe the interactions between the bases of the system, but the random forcing from our system should be described separately for each base. For this reason, the noise and damping should be added to the equation of motion of each base, that is, the system in u_n and v_n, and only after that the change of variables to (x_n, y_n) can be made.

Taking into consideration the above observations, we add noise and damping terms to (3.1.2)–(3.1.3) to obtain

$$
\begin{aligned}
(3.4.1)\ m_n\frac{d^2u_n}{dt^2} &= k^{(u)}_{n+\frac{1}{2}}(u_{n+1}-u_n) - k^{(u)}_{n-\frac{1}{2}}(u_n-u_{n-1}) - \frac{1}{2}F_n(u_n-v_n) \\
&\quad -\widetilde{\eta_n}\frac{du_n}{dt} + \widetilde{\epsilon_n}\xi^u_n(t), \\
(3.4.2)\ m_n\frac{d^2v_n}{dt^2} &= k^{(v)}_{n+\frac{1}{2}}(v_{n+1}-v_n) - k^{(v)}_{n-\frac{1}{2}}(v_n-v_{n-1}) + \frac{1}{2}F_n(u_n-v_n) \\
&\quad -\widetilde{\eta_n}\frac{dv_n}{dt} + \widetilde{\epsilon_n}\xi^v_n(t),
\end{aligned}
$$

Considering $x_n = u_n + v_n$ and $y_n = u_n - v_n$ and $k^{(u)}_n = k^{(v)}_n = k_n$ for all n, the system becomes

$$
\begin{aligned}
(3.4.3) m_n\frac{d^2x_n}{dt^2} &= k_{n+\frac{1}{2}}(x_{n+1}-x_n) - k_{n-\frac{1}{2}}(x_n-x_{n-1}) - \widetilde{\eta_n}\frac{dx_n}{dt} \\
&\quad +\widetilde{\epsilon_n}(\xi^u_n(t)+\xi^v_n(t)), \\
(3.4.4) m_n\frac{d^2y_n}{dt^2} &= k_{n+\frac{1}{2}}(y_{n+1}-y_n) - k_{n-\frac{1}{2}}(y_n-y_{n-1}) - F_n(y_n) - \widetilde{\eta_n}\frac{dy_n}{dt} \\
&\quad +\widetilde{\epsilon_n}(\xi^u_n(t)-\xi^v_n(t)).
\end{aligned}
$$

Let $N(\mu,\sigma^2)$ be a Gaussian random variable, with mean μ and standard deviation σ. Since for a random variable X with normal distribution $N(0,1)$ we have $f_X(x) = \frac{e^{-x^2/2}}{\sqrt{2\pi}}$ and $P(z \le Z) = \int\int f_{XY}(x,y)$, then when we add two random variables ($Z = X + Y$), both with normal distribution $N(0,1)$, we have

$$
\begin{aligned}
(3.4.5)\ f_Z(z) &= \int_{-\infty}^{\infty} f_X(x)f_Y(z-x)dx = \int_{-\infty}^{\infty}\frac{1}{2\pi}e^{-x^2/2-(z^2+x^2-2zx)/2}dx \\
&= \frac{e^{-z^2/2}}{2\pi}\int_{-\infty}^{\infty}e^{-x^2+zx}dx = \frac{e^{-z^2/4}}{2\pi}\int_{-\infty}^{\infty}e^{-x^2+zx-z^2/4}dx \\
&= \frac{e^{-z^2/4}}{2\pi}\int_{-\infty}^{\infty}e^{-(x-z/2)^2}dx = \frac{e^{-z^2/4}}{2\pi}\sqrt{2\pi} \\
&= \frac{e^{-\frac{1}{2}(z/\sqrt{2})^2}}{\sqrt{2\pi}}.
\end{aligned}
$$

The random variable obtained $Z = X + Y$ is normal distributed $Z \approx N(0,2) = \sqrt{2}N(0,1)$.

Since for all n each of $\xi^u_n(t)$ and $\xi^v_n(t)$ represent an independent Wiener processes that can be written in the discrete case as $\sqrt{\Delta t}N(0,1)$, we obtain that

$\xi_n^u(t) \pm \xi_n^v(t) = \sqrt{2}\xi_n(t)$. Comparing (3.4.3)-(3.4.4) with (3.4.1)-(3.4.2) we note that the damping coefficients are identical ($\widetilde{\eta_n}$), but the noise coefficients are larger in (3.4.3)-(3.4.4) than in (3.4.1)-(3.4.2). Since the fluctuation-dissipation relation involves the noise and damping coefficients and (3.4.3)-(3.4.4) has different noise amplitude than (u, v) system, the (x, y) system satisfies an alternative fluctuation-dissipation relation, which will be determined later.

Taking $m_n = m, \forall n$, $k_{n+\frac{1}{2}} = mk$, for all n, except for $k_{\frac{1}{2}} = k_{-\frac{1}{2}} = m\widehat{k}$, $V_n(y) = \frac{1}{2}m\gamma y^2$, for $n \neq 0$, and $V_0(y) = mE_0(y)$, where E_0 is the energy function for the breathing base-pair, which will be discussed later, and also considering $\widetilde{\eta_n} = m\eta_n$, with $\eta_n = \eta$ for $n \neq 0$, and $\widetilde{\epsilon_n} = m\bar{\epsilon_n}$, with $\bar{\epsilon_n} = \bar{\epsilon}$ for $n \neq 0$, our system in y_n becomes

$$\frac{d^2y_n}{dt^2} = k(y_{n+1} - 2y_n + y_{n-1}) - \gamma y_n - \eta\frac{dy_n}{dt} + \bar{\epsilon}\sqrt{2}\xi_n(t), \qquad |n| > 1, \tag{3.4.6}$$

$$\frac{d^2y_{-1}}{dt^2} = \widehat{k}(y_0 - y_{-1}) - k(y_{-1} - y_{-2}) - \gamma y_{-1} - \eta\frac{dy_{-1}}{dt} + \bar{\epsilon}\sqrt{2}\xi_{-1}(t), \tag{3.4.7}$$

$$\frac{d^2y_0}{dt^2} = \widehat{k}(y_1 - 2y_0 + y_{-1}) - \frac{dE_0}{dy}(y_0) - \eta_0\frac{dy_0}{dt} + \bar{\epsilon_0}\sqrt{2}\xi_0(t), \tag{3.4.8}$$

$$\frac{d^2y_1}{dt^2} = k(y_2 - y_1) - \widehat{k}(y_1 - y_0) - \gamma y_1 - \eta\frac{dy_1}{dt} + \bar{\epsilon}\sqrt{2}\xi_1(t). \tag{3.4.9}$$

Observe that the fluctuation-dissipation relation for system (3.4.1)-(3.4.2), that is, $\widetilde{\eta} = \widetilde{\epsilon}^2/2k_B\widetilde{T}$ implies

$$\eta = \frac{\bar{\epsilon}^2}{2k_BT}, \tag{3.4.10}$$

and $\eta_0 = \bar{\epsilon_0}^2/2k_BT$. The noise coefficients in this case are $\epsilon = \sqrt{2}\bar{\epsilon}$ and $\epsilon_0 = \sqrt{2}\bar{\epsilon_0}$, and based on (3.4.10) we obtain that the alternative fluctuation-dissipation relation is

$$\eta = \frac{\epsilon^2}{4k_BT}, \tag{3.4.11}$$

and $\eta_0 = \epsilon_0^2/4k_BT$. We observe that the fluctuation-dissipation relation (3.4.11) for our $x - y$ system (3.4.3)-(3.4.4) has an increased noise to damping ratio of 2 over that from (3.4.10) for $u - v$ system (3.4.1)-(3.4.2). The reason for this is that (3.4.1)-(3.4.2) is a coupled system of $2N$ differential equations, whilst each of (3.4.3) and (3.4.4) is a closed system of just N differential equations. Yet each of (3.4.3) and (3.4.4) contains the effects of all $2N$ noise terms from (3.4.1)-(3.4.2).

Next, we have to take into account that AMBER computes at each time the coordinates of each atom of a base-pair. The new coordinates of an atom are influenced by the neighboring atoms. Our initial mesoscopic model considers a base as a single particle, while AMBER considers the bases as a group of molecules linked together by several bonds. Since we try to fit our parameters using AMBER data, it might be possible that the new expression for the fluctuation-dissipation relation is still wrong.

Analysing the four bases of our DNA duplex, we observe that adenine (A) as well as thymine (T) contain 32 atoms, guanine (G) contains 33 atoms, while cytosine (C) contains only 30 atoms. Hence we can say that on average each base contains 32 atoms and the equation of motion of each base is actually obtained from the equations of motion of the 32 atoms composing the base. Our system parameters are fitted to data obtained using AMBER, which simulates all atoms in a 12 base-pair DNA sequence solvated in water. For this reason, we consider the generalised fluctuation-dissipation relation

$$\eta = \frac{\epsilon^2}{C k_B T}, \tag{3.4.12}$$

where C is a parameter to be determined. The four bases contain different combinations of *Hydrogen*, *Carbon*, *Nitrogen* or *Oxygen* atoms. Whilst the mass of *Carbon*, *Nitrogen* and *Oxygen* are similar, that of *Hydrogen* is negligible. Since *Hydrogen* represents about half of the atoms of each base, we may expect $C = 64$. On the other hand, we consider the distance between the bases of a pair to be the distance between the atoms from the extremities of these bases, which are linked to one or two atoms only. In addition, the interactions between the DNA atoms and the solvent surrounding it also influence the value of the constant C. Indeed, MD analysis of a DNA sequence solvated in a water box show that the box slows the DNA atoms and hence, influences the value of the damping coefficient. Thus, we have $2 < C < 64$ and a precise value of C will be determined later.

Note that the parameter C determines the ratio of noise to damping and represents an important quantity in our system. Too much damping means no breathing events occur, while not enough damping allows too many breathing events to take place. One may think this value will be the same for all DNA

twist angles. From structural point of view, the DNA sequence does not modify with the twist angles, however, interactions between atoms within a base and interactions between the base and its surrounding water box may depend on twist angle. This allows a breathing base to explore different volumes of space. We model this effect by varying the parameter C with twist angle. Our simulations show that $4.8 \leq C \leq 8$, depending on the twist angle.

3.5 Summary

In this chapter we have introduced a new stochastic differential equation model for a DNA duplex useful for simulating short timescale breathing events at a defect. After presenting the nonlinear deterministic model, we derived the stochastic version of our system, which incorporates noise and damping terms.

Next, we show how the system parameters can be fitted to data from the MD-simulation package AMBER simulations using the Maximum Likelihood Estimation (MLE) method. We also present an improved MLE method containing a penalty term, as well as the smooth spline approximation of a discrete function, both useful to determine a more accurate value of the inter-chains potential energy function.

We also emphasize the difference between the potential of mean force and the various potential energies in our system, by determining an approximation for the potential of mean force expression.

Finally, we also discuss the need of an alternative fluctuation-dissipation relation in reduced mesoscopic models. We show how the noise coefficient changes in derivation of reduced models, influencing the fluctuation-dissipation relation. We also explain the importance of the damping term in preserving the system energy and show its contribution to the total energy of the system as a deterministic force.

CHAPTER 4

Analysis of Parameter Values

In this chapter, we show how data from AMBER influences the values of parameters in our SDE system, as well as an analysis of how each parameter influences the length and the frequency of the breathing events, by considering the expression of the potential of mean force, obtained using (3.3.22).

Note that the MLE method described in Section 3.3.1 is sensitive to input data. If the data used for parameters fitting is not representative of the behaviour of a DNA sequence, then the parameter values obtained may not be the appropriate ones. In addition, considering a wider confidence interval increases the probability of including the right result.

4.1 An example calculation

For each twist angle, several computations are needed to obtain an input data sample for our MLE method, which is representative of DNA breathing behaviour. Also several steps have to be covered to obtain the parameter values, as follows:

1. We first simulate using AMBER 20 ns of data and we keep this information about each 1 ps. It is impossible to store 20 ns of data every 2 fs – the timestep used for AMBER simulations – given that such simulations would require more than 10 weeks and about 8000 GB of storage capacity.

At this point, we are only interested in the distances between the bases of the A-F pair. Applying the methodology described in Section 2.3 for a simulation of a 30° undertwisted DNA molecule, we obtain the distances from Figure 4.1.

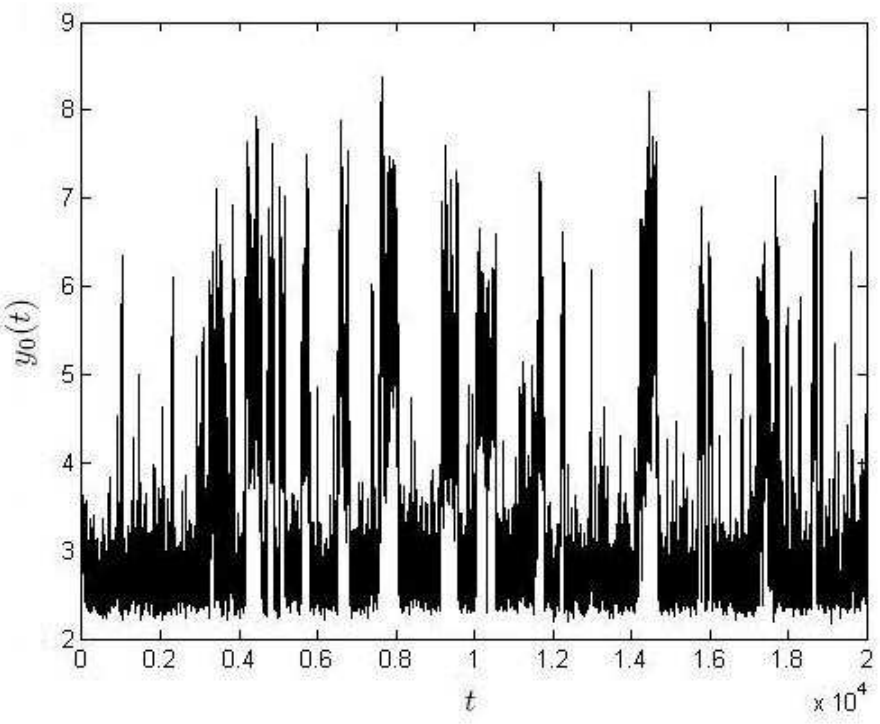

Figure 4.1: Graph of the distance $y_0(t)$ in Å between bases of the breathing pair, plotted against time measured in ps, obtained from a 20 ns AMBER simulation for a 30° undertwisted DNA.

Considering that any value above 3.6 Å represents an open state of the A-F base-pair, we can compute the percentage of time spent breathing. Note that our results suggest to ignore the first 5 or 6 ns of each simulation, since the data shows an unrepresentative initial transient. We finally obtain an average of 26.0890% of time spent breathing.

2. Next, we perform a shorter AMBER simulation, of 2 ns, for example, and we store information about the position and velocity of each atom every 2 fs. After computing the distances specific to A-F pair, we select a subset of this data, which agrees in time spent breathing at the defect site with the 20 ns AMBER simulations. For the same 30° undertwisted DNA sequence, we obtain the distances from Figure 4.2. In this particular case, the

representative subset starts at the first data point and ends after 1.8×10^6 fs, for which 25.8900% of time is spent breathing.

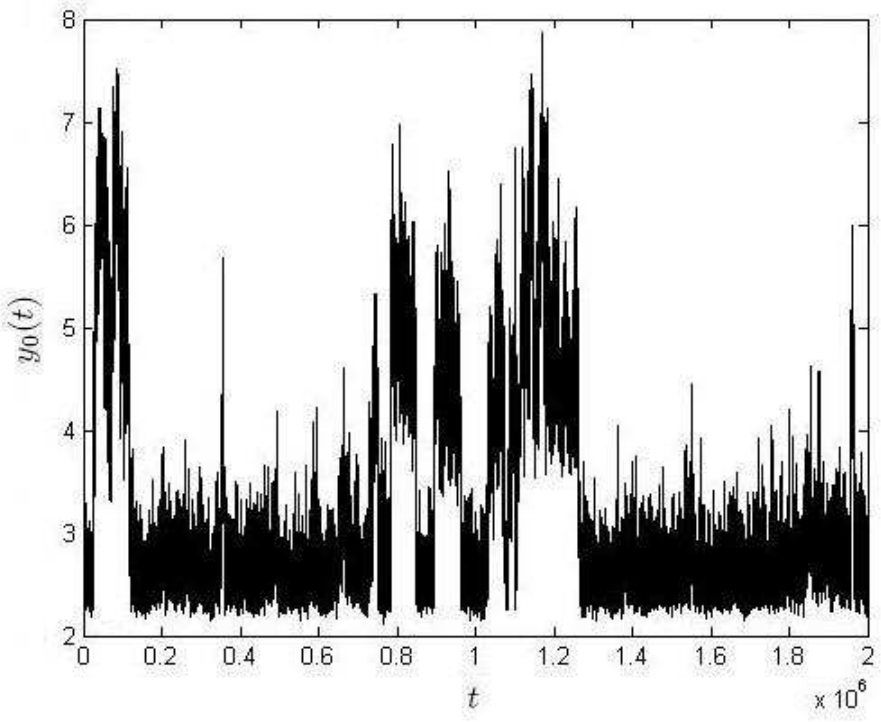

Figure 4.2: Graph of the distance $y_0(t)$ in Å between bases of the breathing pair, used for fitting system parameters, plotted against time measured in fs, obtained from a 2 ns AMBER simulation for a 30° undertwisted DNA.

Having the representative subset data, we compute the displacements from equilibrium y_{-1}, y_0, y_1 and y_2 by subtracting from the base-pairs distances their mean value. We also compute the velocities v_0 and v_1, all this information being required by the MLE method.

3. Using (3.3.2) we apply MLE on y_0, y_1, v_1 and y_2 to determine ϵ, k, $\widehat{k}$ and γ. We also determine η based on the fluctuation-dissipation relation (3.4.12).

4. Using (3.3.13) and the previously determined value of $\widehat{k}$, we perform MLE on y_{-1}, y_0, v_0 and y_1 to determine ϵ_0 and $E_0(y_0)$, as well as η_0 using the fluctuation-dissipation relation.

4.2 Influence of data samples on parameter values

For each twist angle we have analysed the effect of discarding 6, 7, 8, 9 and 10 ns of simulation to determine the breathing time variation. The results displayed in Table 4.1 show that the time spent breathing is different for different portions of data analysed, but the way in which it varies with the twist angle is preserved, no matter which sample interval is used. Moreover, the variation in each row of Table 4.1 is no more than 5%, which is quite small compared to the 45% variation from the entire table.

Twist angle	$14ns$	$13ns$	$12ns$	$11ns$	$10ns$
30°	26.9500%	26.8769%	25.1583%	26.7000%	24.7600%
32°	45.1143%	48.5000%	48.4000%	50.7909%	46.3300%
33°	23.8786%	25.4385%	27.3750%	29.5273%	27.5400%
34°	21.7571%	23.1692%	24.9583%	21.0636%	23.0500%
35°	26.2500%	24.3538%	25.6500%	27.5545%	28.2600%
36°	40.1357%	38.0538%	37.9833%	35.7455%	37.9500%
38°	64.1143%	65.1462%	69.0917%	70.2636%	70.9000%
40°	57.0429%	55.9000%	58.0250%	63.2364%	66.9900%

Table 4.1: Time spent breathing for each angle analysed, using different numbers of data points. The underlined values represent the smallest and largest percentages for a given twist angle.

For each angle, we compute the parameter values from simulation data corresponding to the smallest and the largest proportion of time spent breathing (see the underlined values from Table 4.1). In this way we obtain two confidence intervals for each parameter, which we combine to give the final intervals for our parameters.

4.2.1 Confidence intervals

In Section 2.3, we mentioned that we need to subtract the mean value from the distance between the two bases of a pair to obtain how this distance actually

evolves in time. For a 30° undertwisted DNA sequence, this implies that $\langle y_1 \rangle = 2.0178$ and $\langle y_2 \rangle = 1.9718$. Analysing Figure 4.2, we see that for y_0 it is not possible to obtain a correct value using the mean of the vector data and only data between 1.3×10^6 fs and 1.5×10^6 fs, for example, should be used. Hence, we obtain $\langle y_0 \rangle = 2.5778$.

Subtracting the mean values for y_1 and y_2 from the AMBER data and applying MLE method for l_1 – see (3.3.2) for its definition – and using $C = 6.5$, we obtain the confidence intervals quoted in Table 4.2.

k	$\widehat{k}$	γ	ϵ
$[5.8837, 8.9828]$	$[1.3050, 1.5597]$	$[126.0255, 130.8682]$	$[3.3871, 3.4094]$

Table 4.2: Parameter values from MLE, obtained for a 30° undertwisted DNA, using for each base-pair the mean value of displacements.

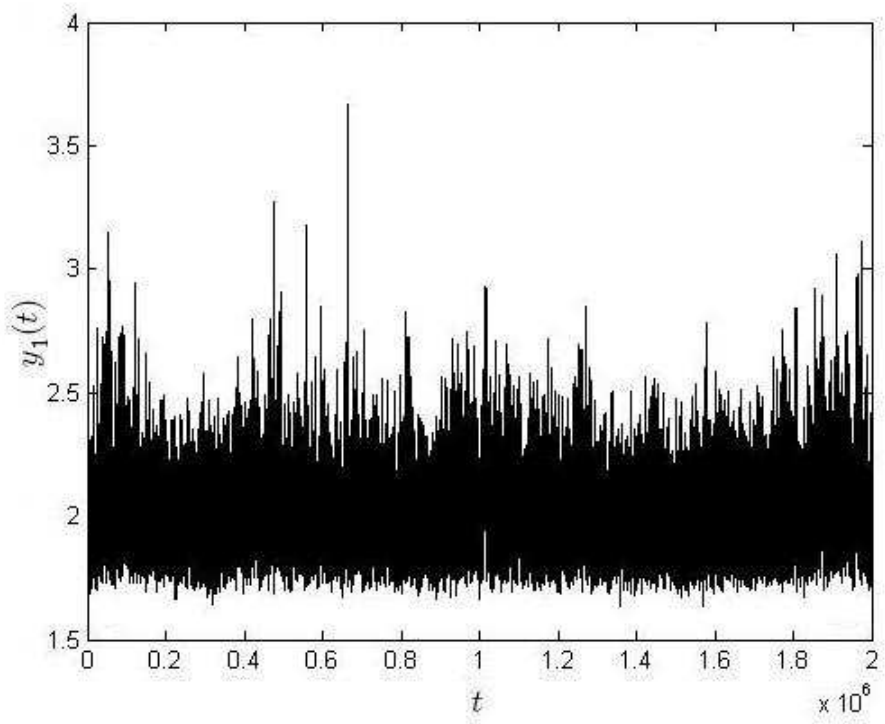

Figure 4.3: Graph of the distance $y_1(t)$ in Å between bases of an A-T pair, used for fitting system parameters, plotted against time measured in fs, obtained from a 2 ns AMBER simulation for a 30° undertwisted DNA.

Figure 4.3 shows that the equilibrium value for y_1 is not its mean either. One may think that small differences in the equilibrium values would not affect the parameters estimates obtained from MLE. However, using 1.9750Å for the equilibrium value of y_1 and 1.9596Å for y_2, we have

k	$\widehat{k}$	γ	ϵ
$[9.2146, 12.0713]$	$[3.4965, 3.8737]$	$[116.0208, 121.7915]$	$[3.3908, 3.4125]$

Table 4.3: Parameter values from MLE, obtained for a 30° undertwisted DNA, using equilibrium base-pairs values that are smaller than the mean displacements.

The later values for equilibrium are obtained by considering only a part of the AMBER data, for which the range of displacements is 0.5 Å, for example. Comparing Tables 4.2 and 4.3 we observe that $\widehat{k}$ suffers the most dramatic change of value.

As will be discussed in the next section, some values of our parameters might not be consistent with the values obtained for other twist angles. This is due to the expectation that if 20 tests were performed at a 5% significance level, one would expect one error. Extending the confidence intervals can solve this problem. This can be achieved by using the Bonferroni correction, which repleces a confidence interval of $100(1-\alpha)\%$ with a $100(1-\alpha/n)\%$ confidence interval, where n is the number of data sets tested and α is the significance level. In our case $n = 8$, since we analyse eight different twist angles, and $\alpha = 0.05$, hence the 99.375% confidence interval for $\widehat{\theta_i}$ become

$$\left[\widehat{\theta_i} - 2.5\sqrt{(I_{obs}^{-1}(\widehat{\theta}))_{ii}},\ \widehat{\theta_i} + 2.5\sqrt{(I_{obs}^{-1}(\widehat{\theta}))_{ii}}\right], \tag{4.2.1}$$

where $\widehat{\theta} = (\widehat{\theta_i})$ is the estimate of the vector of parameters θ and I_{obs} is the observed information, as defined in Section 3.3.1.

Applying the Bonferroni correction, we replace the confidence intervals from Table 4.3 with those displayed in Table 4.4. Comparing with the confidence intervals from Table 4.2, we observe that the noise amplitude and the inter-strands spring constant γ suffer only minor changes in value, but the values of the along-chain interaction parameters k and $\widehat{k}$ are strongly affected.

k	$\widehat{k}$	γ	ϵ
$[8.8589, 12.4481]$	$[3.4613, 3.9086]$	$[115.5826, 122.2566]$	$[3.3894, 3.4139]$

Table 4.4: Confidence intervals for parameter values from MLE, for a 30° undertwisted DNA, obtained using Bonferroni correction.

This situation can be easily explained, given that the noise term is not dependent on displacements, the inter-chain force only depends on y_1, while the forcing terms related to k and $\widehat{k}$ depend on y_1 and y_2, and y_0 and y_1, respectively. Hence, the last two terms are more affected by the change in the values subtracted from the base-pairs distances, obtained from AMBER, than the other two forcing terms involved in l_1 maximization.

4.3 The fluctuation-dissipation relation

As already mentioned, one of the most important parameters in our system is $C = \epsilon^2/\eta k_B T$ from the fluctuation-dissipation relation (3.4.12), which varies with twist angle. The values used for this parameter are presented in Table 4.5. The variety of values of C is due to the water box that slows the atoms of the DNA sequence. This event is generated by the interactions between the solvent and DNA atoms, which are angle dependent.

Twist angle	30°	32°	33°	34°	35°	36°	38°	40°
C	6.5	6	5.8	5.6	4.8	7	7.25	8

Table 4.5: Parameter C values.

One might think that the parameter C value should be fitted to the AMBER data as the other parameters were, using MLE method. Recall how the displacements and velocities were obtained in Section 2.3: we measured the distances between the extremities atoms of the bases of each pair, while the velocities were obtained via using (2.3.2). When C was treated as a parameter in the MLE process, we obtained similar values for all parameters, while C was predicted

to have a value less than 1, even though our computations from the previous chapter show that $C \geq 2$.

Moreover, AMBER uses rescaling to keep the system at a fixed temperature. This means that some measurable quantities, for example, the velocities, are rescaled in order to keep the average kinetic energy constant. Also, there is no noise explicitly involved in AMBER simulations, which suggests that the MLE method attempts to compensate the noise contribution through the damping term.

4.4 Parameter values

After fitting the parameters via MLE, for all the twist angles, we select, inside the confidence intervals obtained for each parameter, the values for which the SDE simulations are close to MD results. We obtain the values listed in Table 4.6, for which the best results are obtained when simulating the SDE system.

Twist angle	k	$\widehat{k}$	γ	ϵ	ϵ_0
30°	10.6536	3.6851	120.0904	3.4074	5.6285
32°	9.5585	3.2132	131.0919	3.3585	5.9770
33°	9.5374	2.8261	135.5951	3.3429	5.3214
34°	9.2678	2.4625	145.6987	3.3225	5.4843
35°	8.1819	1.8256	149.5683	3.3471	5.6744
36°	7.6577	1.4307	165.4327	3.3499	5.9238
38°	8.1438	2.1462	139.0797	3.3511	6.8702
40°	19.5297	2.6341	132.0731	3.3550	6.1750

Table 4.6: Parameter values obtained using MLE on l_1 and l_0 – see (3.3.2) and (3.3.13) for definitions.

As can be seen, the along-chain bonds k and $\widehat{k}$ become weaker as the twist angle is increased from 30°. On the other hand, the interchain bond γ becomes stronger with twist angle, but once the DNA becomes overtwisted (twist angle greater than 36°) the along-chain bonds become stronger and the interchain

bonds decrease. From 36° upwards all these trends are reversed and we see a 20.19% decrease in γ and 90% increase in $\widehat{k}$. Moreover, the noise coefficient ϵ is almost constant, varying by only 0.2%, whilst for the A-F pair we observe small oscillations of 15.98% in the noise coefficient ϵ_0.

4.4.1 Noise and damping coefficients

Figure 4.4 shows that, in the case of the breathing pair, we need more noise (higher value for the noise coefficient ϵ_0) for the twist angles for which the DNA sequence spends more time breathing – see Table 4.1. The ϵ values suggest that the extreme twist cases (30° and 40°) are slightly noisy than those closer to the normally twisted DNA.

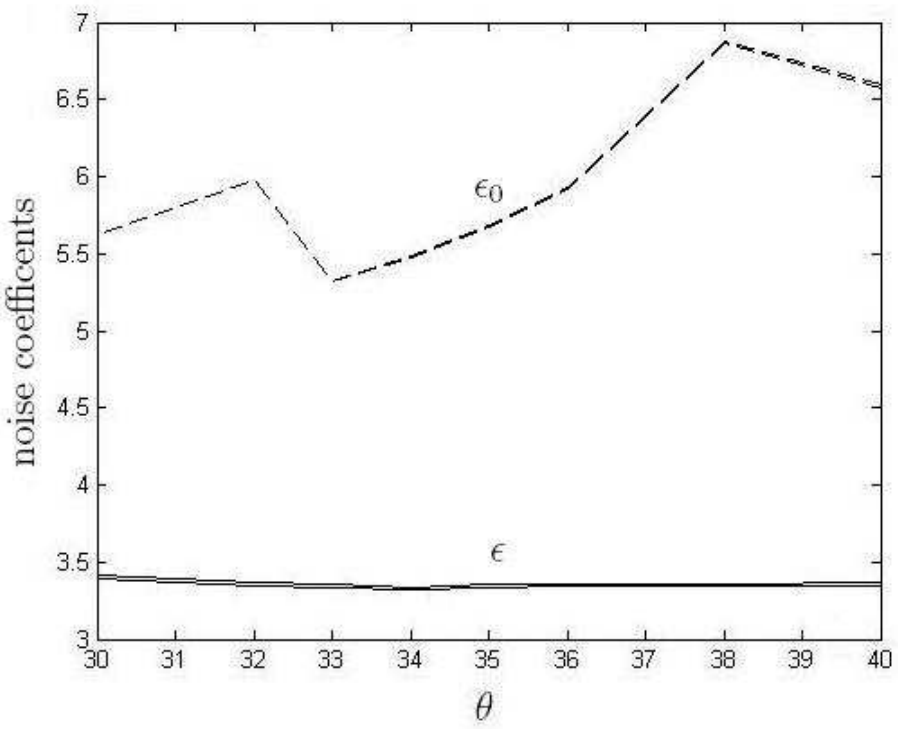

Figure 4.4: Illustration of the confidence intervals of system parameters ϵ (continuous line) and ϵ_0 (dash line), both measured in Å $ps^{-3/2}$, plotted against the twist angle θ.

Using the fluctuation-dissipation relation (3.4.12) we can determine the values of the damping coefficients η and η_0. Table 4.7 presents the averaged values of

damping amplitude, obtained using for ϵ and ϵ_0 the values from the middle of the confidence intervals.

Twist angle	η	η_0
30°	2.1910	5.9986
32°	2.3086	7.3282
33°	2.3797	6.0089
34°	2.4329	6.6104
35°	2.8725	8.2561
36°	1.9735	6.1699
38°	1.9084	8.0127
40°	1.7317	5.8676

Table 4.7: Values of damping coefficients η and η_0, both measured in ps^{-1}.

The damping rate of the breathing pair η_0 has a variation with twist angle of 40.7%, while for the rest of the base-pairs we observe a variation of 65.87% in η. One might expect the two parameters to follow similar variations as the noise coefficients (η,η_0), but this does not hold, since parameter C has different values for each twist angle and this strongly influences the final values of the damping coefficients.

As mentioned in Sections 3.3.3 and 3.3.4, the damping term contributes to the deterministic potential of mean force and a larger damping coefficient means less time spent breathing, which shows that damping influences breathing duration. In addition, as will be discussed later in this chapter, the damping term also influences the breathing frequency.

4.4.2 Along-chain interactions

In Figure 4.5 we show the variation of the along-chain interactions parameters k and $\widehat{k}$. For the 40° overtwisted DNA, the values of k are higher than the values for the other twist angles, but the variation is very high. One explanation is that at this extreme twist angle the bases situated on the same strand might be strongly connected by covalent bonds.

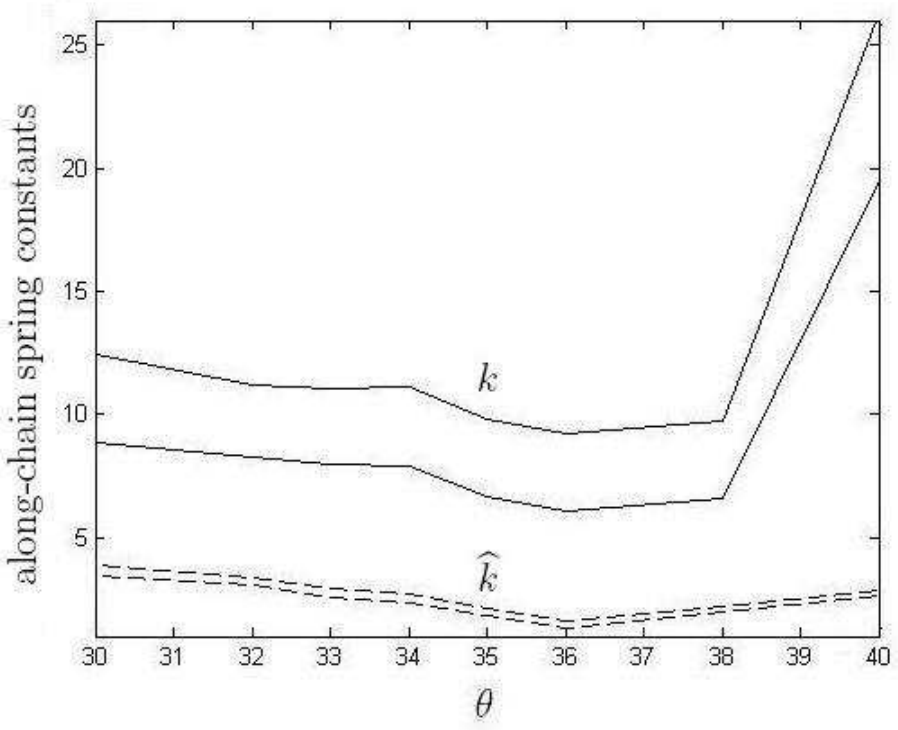

Figure 4.5: Illustration of the confidence intervals of system parameters k (continuous line) and $\widehat{k}$ (dash line), both measured in ps^{-2}, plotted against the twist angle θ.

In addition, both, k and $\widehat{k}$, follow the same path, decreasing in value from the extreme angles (30° and 40°) to the normal twist angle (36°). Hence, the defect only affects the bond's strength, but their variation with twist angle is preserved.

4.4.3 Inter-chain interactions

Figure 4.6 presents the confidence intervals for the inter-chain interactions parameter γ and suggests that the most stable system is obtained for the normally twisted DNA sequence, that is, for 36° of twist. A higher value of γ means stronger interactions between the bases of a pair. This result, combined with the opposite behaviour of the intra-strand coefficient k (Figure 4.5), show that in an undertwisted, as well as, in an overtwisted DNA sequence the along-chain interactions are stronger, while the inter-chain interactions are weaker than in the case of the 36° twisted DNA strand.

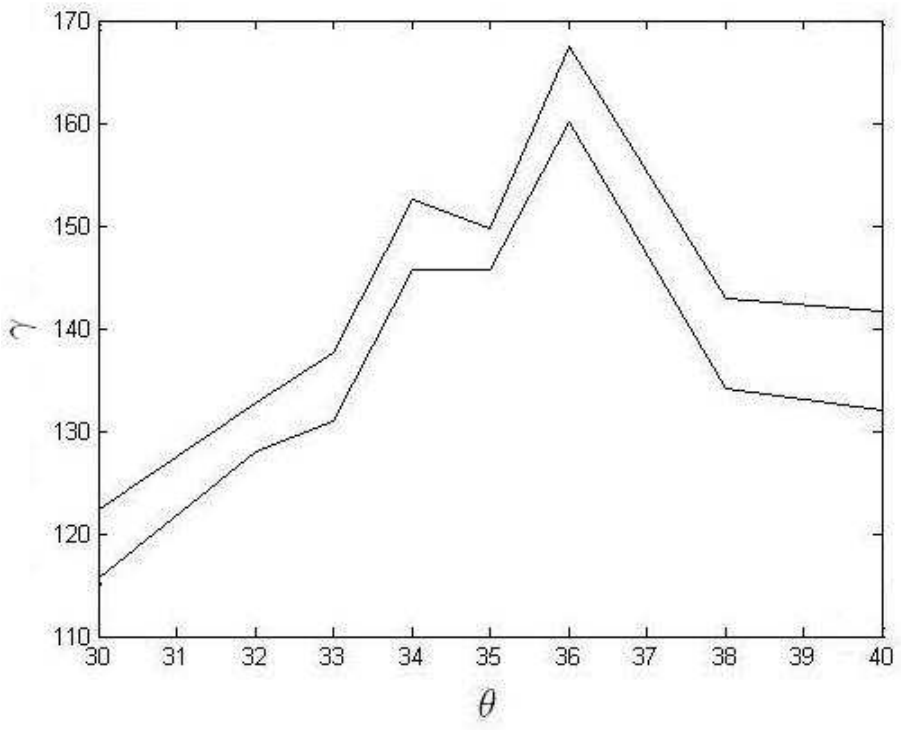

Figure 4.6: Illustration of the confidence intervals of system parameter γ, measured in ps^{-2}, plotted against the twist angle θ.

Note that the parameter γ is fitted to AMBER data for an A-T base-pair (n=1). The bases of such a pair are linked by two hydrogen bonds and the bases of a C-G pair are linked by three hydrogen bonds, but our model does not take into account which type of base-pairs our DNA sequence contains. Hence, using an average value between γ_{AT} and γ_{CG} solves this problem, but this means obtaining information about the velocity and coordinates for at least another base-pair. Alternatively, supposing that each hydrogen bond has equal contribution to the interactions between the bases of a pair, we use for our simulations $\gamma = 5\gamma_{AT}/4$.

Analysing Figures 4.5 and 4.6 we observe that in the case of a 34° undertwisted DNA sequence, the confidence intervals for parameters k and γ are not consistent with the behaviour of the other twist angles, which proves that using the Bonferroni correction is helpful to determine correct values for our parameters. In other words, the wider range of values obtained by using the Bonferroni correction increases the chances to have the correct parameter value inside the confidence interval.

4.4.4 The A-F inter-base potential $E_0(y_0)$

Applying MLE for l_0 (see Section 3.3.2 for definition), for a 30° undertwisted DNA, using $\widehat{k} = 3.6851$ and taking into account all the improvements specified above, we obtain $\epsilon_0 = 5.6285$ and an expression for $E_0(y_0)$ which is displayed in Figure 4.7. This expression is even more surprising than the one from Figure 3.6. In Figure 3.2, we show $PMF(y_0)$ obtained from AMBER data using bin counts, which suggests that the equilibrium state is around 0 Å. Local minima at 2 Å and 4 Å, indicate breathing states. The $E_0(y_0)$ expression, graphed in Figure 4.7, suggests that the most stable state of our system is an open state at 5 Å, since it has a lower energy than the closed state at 0 Å.

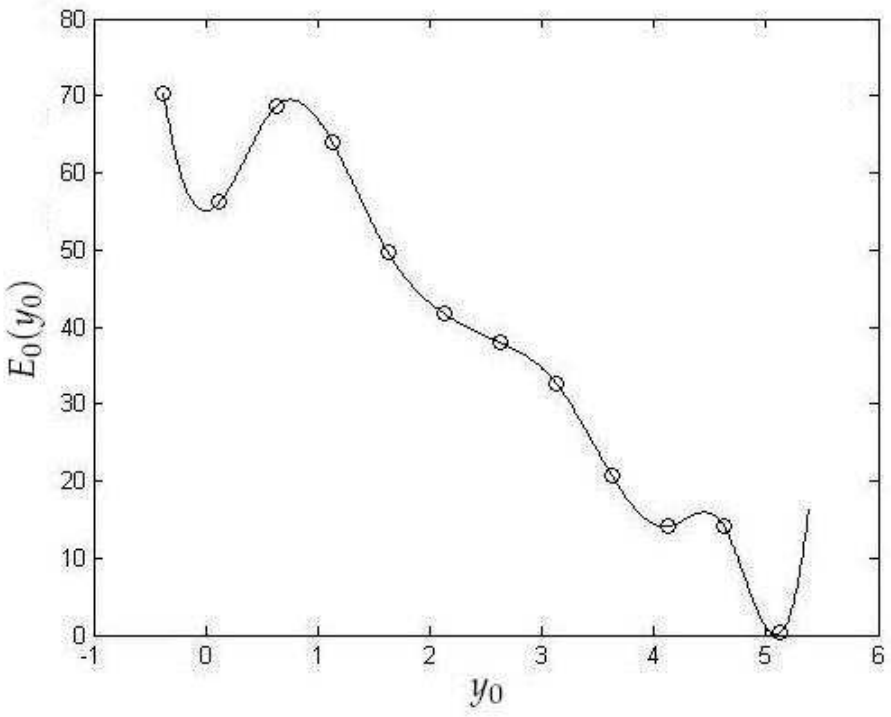

Figure 4.7: Illustration of E_0 function (Å^2 ps^{-2}), obtained using the MLE method for a 30° undertwisted DNA. The small circles describe $E_0(y_0)$ values for the centres of the bins.

Considering the inter-chain contribution to the potential of mean force, we obtain the expression shown in Figure 4.8, which suggests that the damping term also contributes to the potential of mean force.

Indeed, Figure 4.9 shows that the breathing states at $y_0 = 2$ Å and $y_0 = 4$ Å

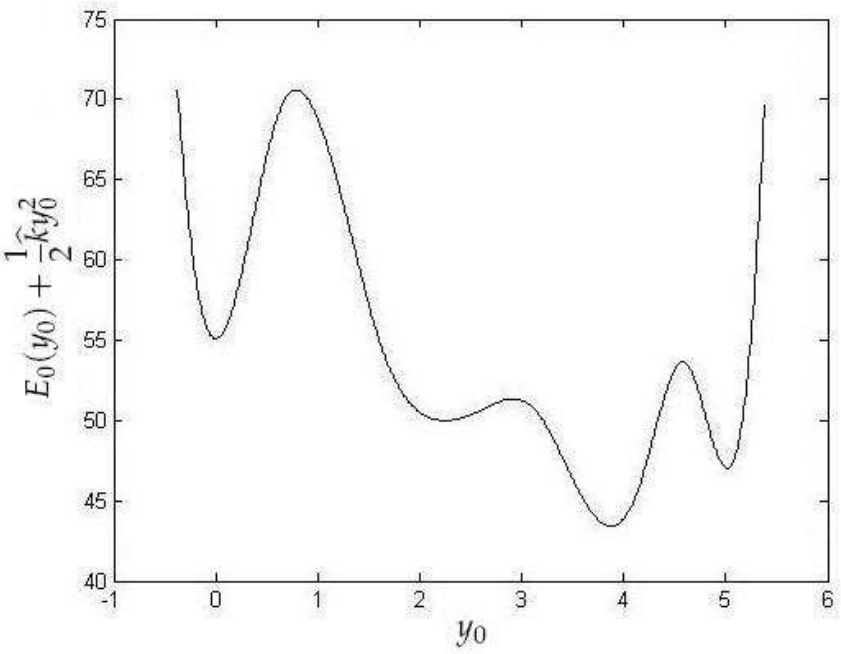

Figure 4.8: Illustration of potential energy function $E_0(y_0) + \frac{1}{2}\widehat{k}y_0^2$ (Å^2 ps^{-2}) of the breathing pair, specific to the SDE system, for a 30° undertwisted DNA.

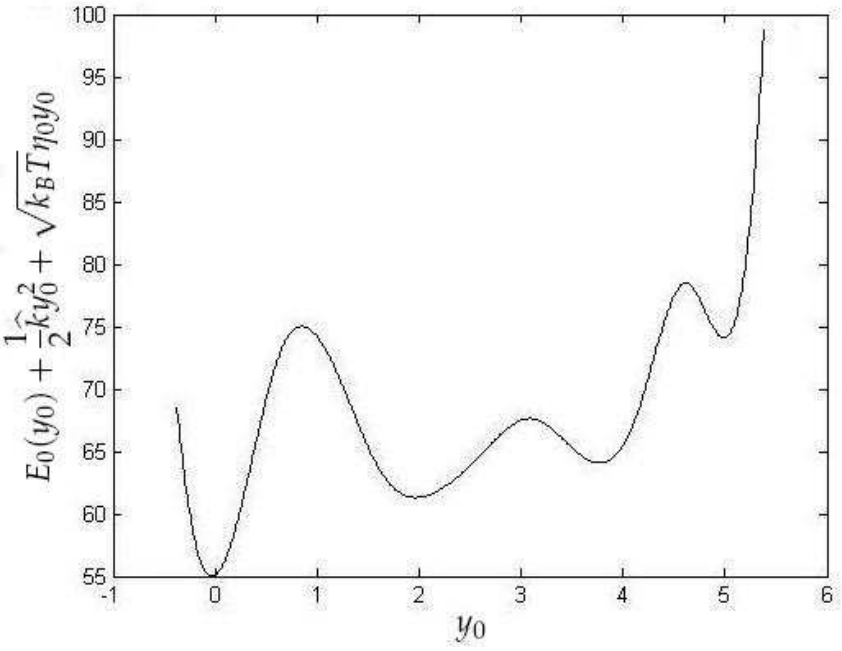

Figure 4.9: Illustration of potential energy function $E_0(y_0) + \frac{1}{2}\widehat{k}y_0^2 + \sqrt{k_B T}\eta_0 y_0$ (Å^2 ps^{-2}) of the breathing pair, including the damping contribution, specific to the SDE system, for a 30° undertwisted DNA.

both have higher energy than the closed state $y_0 = 0$ Å, and thus Figure 4.9 and equation (3.3.22) are close to the classic "potential of mean" force of Figure 3.2. However, as we observe, the two expressions differ by a constant.

Firstly, Figure 3.2 is obtained from a straightforward bin count of the number of timepoints at which the displacement falls within each interval. A fairly crude division of the interval into widths of s =0.5Å is used, and as noted in Figure 3.4 the height of the breathing barrier is dependent on the bin width, s. As s is reduced, the accuracy will improve, and Figure 3.4 shows that the breathing barrier height increases. Figure 4.9 shows the results of a maximum likelihood estimate of the parameters followed by a calculation of the potential of mean force. We observe a significantly higher potential barrier (than in Figure 3.2), since the method of calculation takes account of the order of data points in the sample data. The calculation can distinguish between a few long breathing events and many short breathing events, which is impossible when using the simpler bin-counting algorithm for estimating the PMF.

For the other angles, we obtain different expressions for the energy function $E_0(y_0)$. Figures 4.10(a)-4.16(a) represent $E_0(y_0)$ the for the other seven twist angles for which the DNA sequence is analysed. Some of the differences between the expressions for $E_0(y_0)$ are presented in Table 4.8.

Twist angle	ΔB	ΔE
30°	13.9853	–11.5855
32°	8.3315	–11.0732
33°	12.4900	–5.3957
34°	12.8309	–4.0070
35°	7.6100	–1.8403
36°	19.2640	0.6502
38°	13.6796	–7.1785
40°	14.8387	–9.3841

Table 4.8: Values of ΔB and ΔE (both measured in Å^2 ps^{-2}) corresponding to $E_0(y_0)$. See Figure 3.2 for their definition.

Here ΔB is the height of the barrier from the closed state and ΔE is the energy

difference between the breathing (open) and normal (closed) stated. Hence, the energy barrier from open to closed state is ΔB-ΔE (see Figure 3.2 for an illustration). The energy differences ΔB and ΔE control the frequency and the length of breathing events, respectively, and both vary with twist in the range 30°-40°.

For an undertwisted DNA sequence ΔE is negative, as seen in Figures 4.10(a)-4.13(a) and Table 4.8; for the typical twist of 36° its value is close to zero (see Figure 4.14(a)), while for an overtwisted DNA sequence it decreases again – Figures 4.15(a) and 4.16(a).

However, note that the proportion of time spent breathing is determined by ΔE specific to the total system energy $E_0(y_0) + \frac{1}{2}\widehat{k}y_0^2 + \sqrt{k_B T}\eta_0 y_0$ – see Table 4.9 for more details. In Figures 4.10(b)-4.16(b), we have an approximation of these potentials for the twist angles analysed, which shows that ΔE, the damping coefficient η, and the along-chain spring constant $\widehat{k}$, all determine the percentage of time that the A-F pair spends breathing.

Twist angle	ΔB	ΔE
30°	19.9193	6.2191
32°	16.6841	7.3191
33°	18.8165	8.3515
34°	19.1949	9.2904
35°	15.7846	12.40193
36°	25.8559	14.2498
38°	21.4337	8.9198
40°	20.9347	4.5738

Table 4.9: Values of ΔB and ΔE (both measured in Å^2 ps^{-2}) corresponding to the potential of mean force $PMF(y_0)$.

On the other hand, ΔB controls the frequency at which the barrier between open and closed states is crossed and has a different behaviour. The lower this barrier is, the larger the number of breathing events that occur. Given that ΔB is measured around $y_0 = 1$ Å, the difference between the potential of mean force $PMF(y_0)$ and the energy $E_0(y_0)$ consists of two linear terms. Observe

that ΔB varies from Table 4.8 to Table 4.9 by 6 to 8 units, which shows that the along-chain interactions and damping contribution to $PMF(y_0)$ influence in a small proportion, compared to the inter-strand interactions, the variation of the breathing frequency with twist angle.

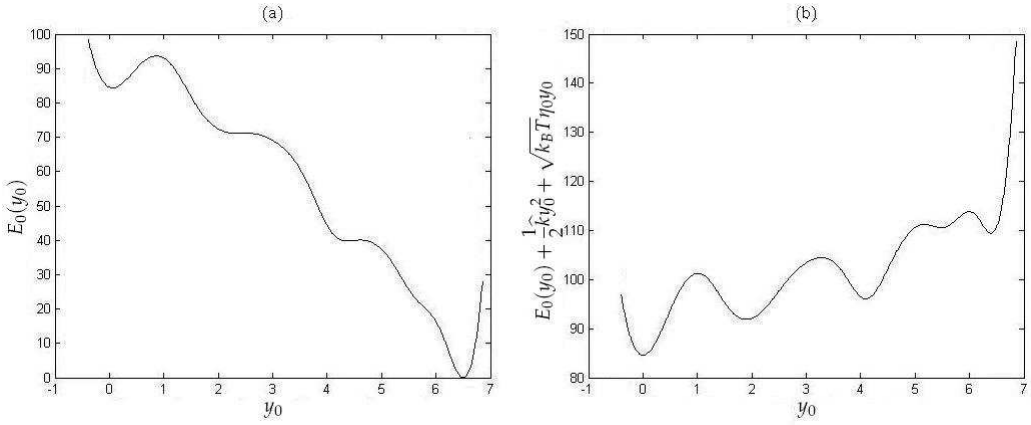

Figure 4.10: Illustration of (a) inter-chain potential ($E_0(y_0)$ in Å^2 ps^{-2}) and (b) potential of mean force ($E_0(y_0) + \frac{1}{2}\widehat{k}y_0^2 + \sqrt{k_B T}\eta_0 y_0$ in Å^2 ps^{-2}) plotted against A-F bond length (y_0 in Å), obtained after fitting parameters for a 32° undertwisted DNA.

Table 4.8 shows that ΔE has similar values for $E_0(y_0)$ for the cases of 30° and 32° of twist, whilst $\widehat{k}$ is 12.8% higher in the first case (see Table 4.6); this is counterbalanced by the differing values of ϵ_0 and C, which imply $\eta_0^{30^\circ} = 5.9985$ and $\eta_0^{32^\circ} = 7.3281$. This means that the total potential energy in the two cases gives rise to similar values for ΔE, although the 32° undertwisted DNA breathes for about 20% more of the simulation time – see Table 4.1. Hence, ΔB is the parameter which is responsible for this difference by allowing more breathing events for a lower value, as it will be discussed in the next chapter.

Further analysis of Table 4.1 shows that the 30° and 33° undertwisted DNA sequences spend similar amounts of time breathing. Moreover, Table 4.8 suggests that, for $E_0(y_0)$, ΔB also has similar values in the two cases, but ΔE is 50% increased for the 33° twist angle. This shows that the potential of mean force is strongly influenced by the along-chain interactions and damping contribution to potential energy. Hence, breathing can be viewed as competition between the along-chain elastic energy and the inter-chain binding energy.

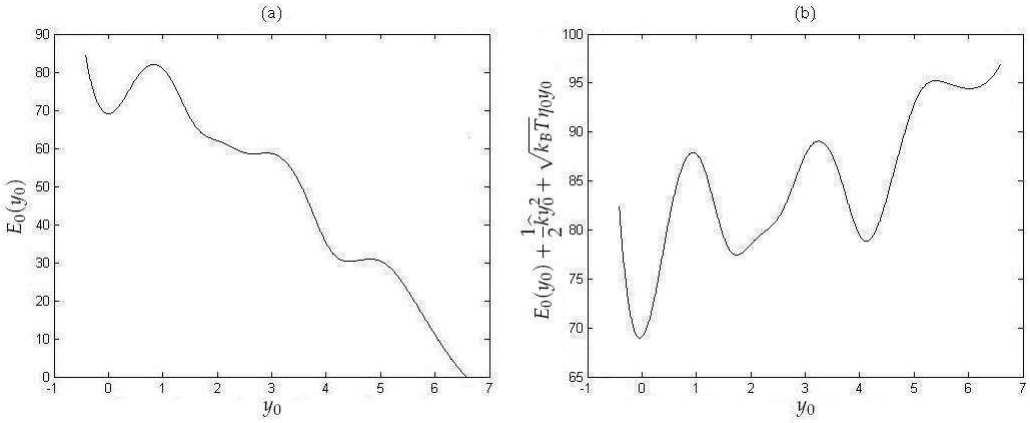

Figure 4.11: Illustration of (a) inter-chain potential ($E_0(y_0)$ in Å^2 ps^{-2}) and (b) potential of mean force ($E_0(y_0) + \frac{1}{2}\widehat{k}y_0^2 + \sqrt{k_B T}\eta_0 y_0$ in Å^2 ps^{-2}) plotted against A-F bond length (y_0 in Å), obtained after fitting parameters for a 33° undertwisted DNA.

Comparing parameters obtained for a 34° twist with those for 35°, we observe that even small differences in breathing time, can be due to different barrier heights of $E_0(y_0)$ (see Table 4.8). A higher ΔE value, as in the case of 35° undertwisted DNA, suggests less time breathing, but decreasing the breathing barrier ΔB might compensate for the ΔE value, as in this case, resulting in more frequent, but shorter, breathing events.

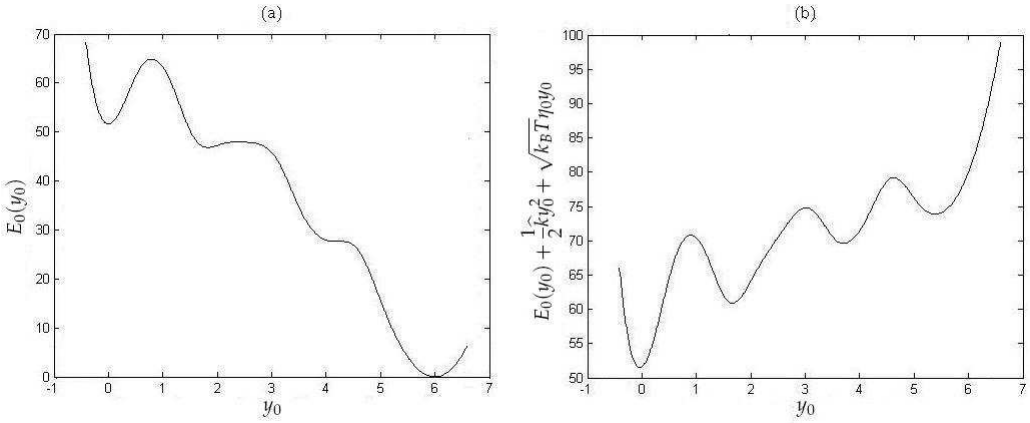

Figure 4.12: Illustration of (a) inter-chain potential ($E_0(y_0)$ in Å^2 ps^{-2}) and (b) potential of mean force ($E_0(y_0) + \frac{1}{2}\widehat{k}y_0^2 + \sqrt{k_B T}\eta_0 y_0$ in Å^2 ps^{-2}) plotted against A-F bond length (y_0 in Å), obtained after fitting parameters for a 34° undertwisted DNA.

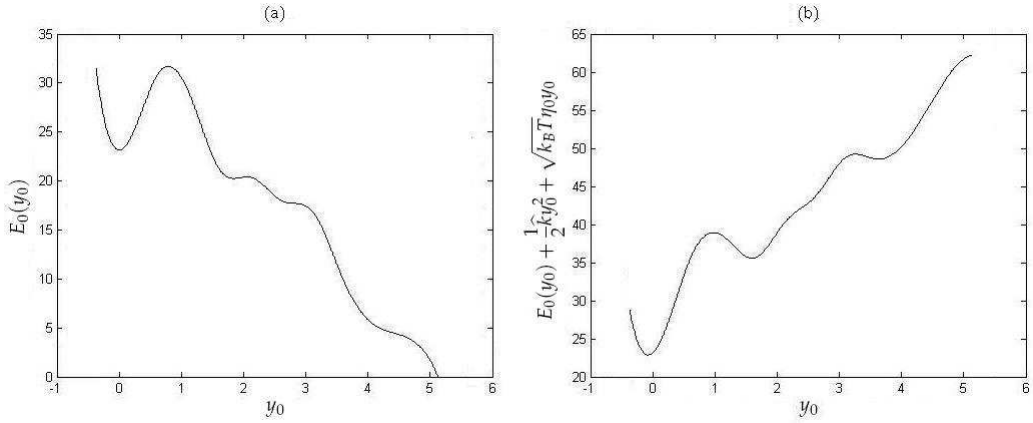

Figure 4.13: Illustration of (a) inter-chain potential ($E_0(y_0)$ in Å^2 ps^{-2}) and (b) potential of mean force ($E_0(y_0) + \frac{1}{2}\widehat{k}y_0^2 + \sqrt{k_BT}\eta_0 y_0$ in Å^2 ps^{-2}) plotted against A-F bond length (y_0 in Å), obtained after fitting parameters for a 35° undertwisted DNA.

Taking into account that, for example, $\eta_0 = 6.0089$ for the 33° twist angle, while for the 36° case we have $\eta_0 = 6.1699$, the damping contribution to the potential of mean force is broadly similar. The differences in ΔE between the two cases (compare Figure 4.11(a) and Figure 4.14(a)) suggest that the stacking interaction parameter $\widehat{k}$ plays an important role for the length of the breathing events.

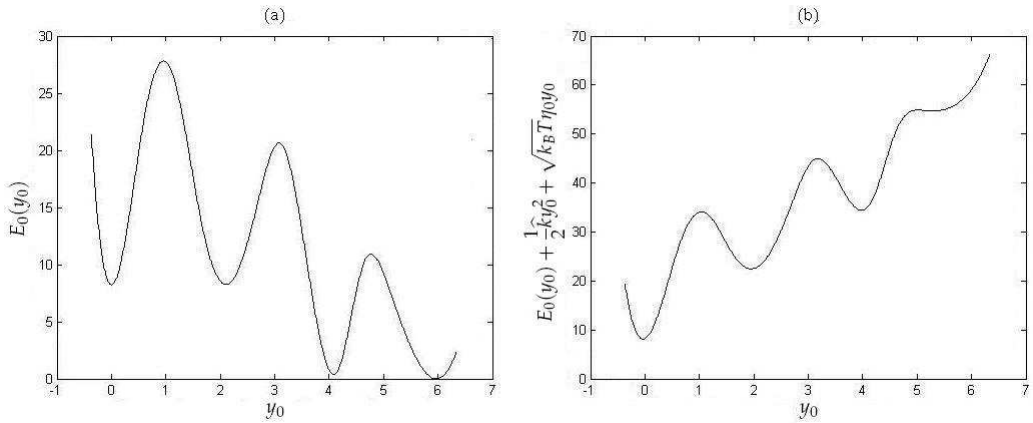

Figure 4.14: Illustration of (a) inter-chain potential ($E_0(y_0)$ in Å^2 ps^{-2}) and (b) potential of mean force ($E_0(y_0) + \frac{1}{2}\widehat{k}y_0^2 + \sqrt{k_BT}\eta_0 y_0$ in Å^2 ps^{-2}) plotted against A-F bond length (y_0 in Å), obtained after fitting parameters for a 36° twisted DNA.

Indeed, the value of $\widehat{k}$ has the most dramatic variation: it decreases with twist

angle until the typical twist angle (36°) is reached and increases with overtwist. A higher value of $\widehat{k}$ means higher energy in the open state and less time spent breathing.

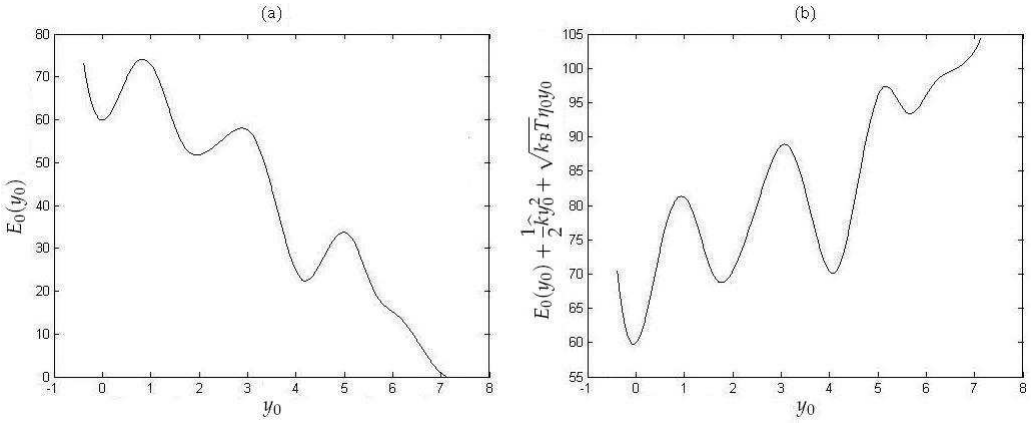

Figure 4.15: Illustration of (a) inter-chain potential ($E_0(y_0)$ in Å^2 ps^{-2}) and (b) potential of mean force ($E_0(y_0) + \frac{1}{2}\widehat{k}y_0^2 + \sqrt{k_B T}\eta_0 y_0$ in Å^2 ps^{-2}) plotted against A-F bond length (y_0 in Å), obtained after fitting parameters for a 38° overtwisted DNA.

Finally, the approximations of the potential of mean force, presented in Figures 4.10(b)-4.16(b), show that the damping and the harmonic inter-chain contribution to the total system energy define the displacements for closed and open states of the A-F pair. For most twist angles, these values are between -0.3 Å and 5 Å.

Note that the overtwisted DNA sequences analysed (38° and 40° twist angles) spend more than 50% of the simulation time breathing, which sugests a lower energy in the open state then in the closed state. But, the potential of mean force expressions (Figures 4.15 and 4.16) suggest that we have the lowest energy in the system when the A-F base-pair is in closed state.

Table 4.9 suggests the same, given that ΔE has positive values for all twist angles. However, observe that the total energy expressions have two local minima at 1.9Å and 3.8Å, hence two breathing states. The time spent breathing is the sum of the time spent in each breathing state, but the A-F pair spends less time in each open state than in the closed state, which explains the form of the potential of mean force.

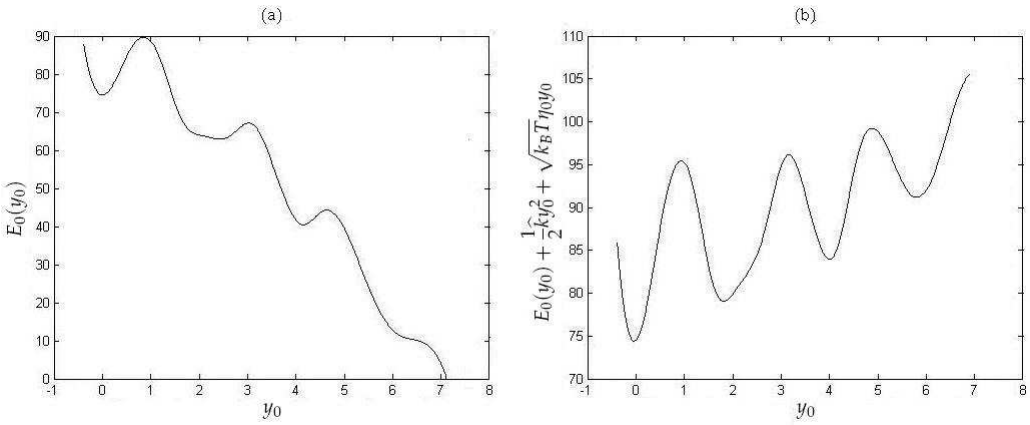

Figure 4.16: Illustration of (a) inter-chain potential ($E_0(y_0)$ in Å^2 ps^{-2}) and (b) potential of mean force ($E_0(y_0) + \frac{1}{2}\widehat{k}y_0^2 + \sqrt{k_B T}\eta_0 y_0$ in Å^2 ps^{-2}) plotted against A-F bond length (y_0 in Å), obtained after fitting parameters for a 40° overtwisted DNA.

4.5 Summary

We start this chapter by showing how to simulate and interpret the data obtained using AMBER in order to avoid obtaining inconsistent parameter values. We also discus the need of selecting a representative data sample and of using the Bonferoni correction to obtain confidence intervals having a larger probability of containing the right parameter values.

We end the chapter by presenting the values of fluctuation-dissipation constant C, as well as the parameter values corresponding to noise and damping terms, along-chain and inter-strands interactions, respectively, and discuss the importance of the opening-closing barrier for breathing events.

CHAPTER 5

System Solutions

To analyse the accuracy of the SDE simulations we compare the breathing frequency and length with the MD simulations obtained using AMBER. The comparison covers a variety of twist angles, involving a DNA sequence with 12 base-pairs that contains a defect, as defined in Chapter 2.

5.1 Undertwisted DNA

Figure 5.1 presents the way in which the distance between the bases of the breathing pair varies over time, for a 30° twisted strand of DNA. As suggested in the previous chapter, by approximating the potential of mean force, we observe three different values around which this distance oscillates:

- 0 Å, which represents the equilibrium (closed or non breathing) state;
- 1.9 Å, which represents the first breathing state;
- 3.8 Å, which represents the second breathing state.

As far as we are aware, it has not yet been determined whether the two breathing states have similar or different causes, i.e. it might be that one base flips to one of the two preferred angles, or it flips out to an angle in one direction and to a different angle in the opposite directions, or even more, the smaller amplitude state may be due to one base flipping out and the larger amplitude

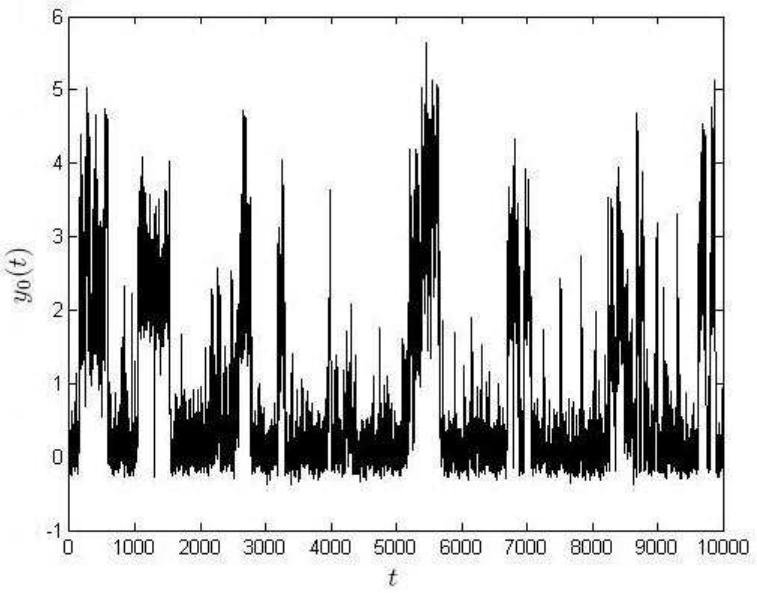

Figure 5.1: Graph of the displacement between bases of the breathing pair (y_0 in Å), plotted against time measured in ps, obtained from an AMBER simulation of 10 ns, for a 30° undertwisted DNA sequence.

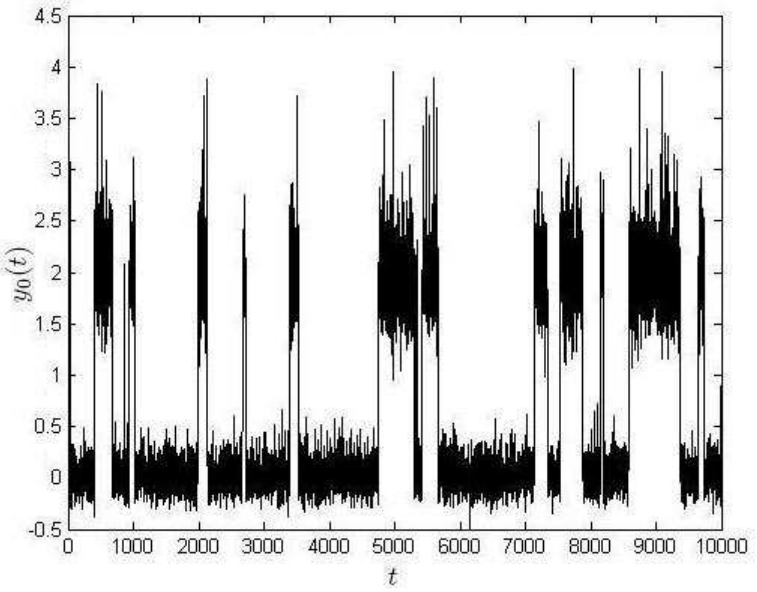

Figure 5.2: Illustration of the displacement (y_0 in Å) between the bases of the breathing pair over 10 ns, obtained using the SDE model for a 30° undertwisted DNA sequence. The parameter values are $C = 6.5$, $\epsilon = 3.4074$, $\epsilon_0 = 5.6285$, $k = 10.6536$, $\widehat{k} = 3.6851$, $\gamma = 120.0904$, while for E_0 the expression from Figure 4.7 was used.

event due to both bases being displaced from their equilibrium. If their nature is similar, then our model is close to reality. Otherwise, each event should be treated separately and a new model which incorporates both events should be developed, for example a model which allows motion in more than one direction. One possible explanation is that in the first breathing state only one base of a pair is breathing, while in the second state both bases are breathing.

We simulated the system using the proposed SDE model (Figure 5.2) and obtained results similar to that from AMBER (Figure 5.1). Some differences may be observed: the AMBER data suggests that the oscillation interval is between -0.3 and 5.7 Å (a 6 Å range), while in our case we have oscillations between -0.3 and 4.1 Å (a 4.4 Å range). One explanation for this reduction is the parameter used in our SDE system, which was eliminated from our equations by redefining the parameters. We have considered that the entire base moves, while in reality just a part of it moves, while the rest remains more or less in the initial position. Moreover, our system contains only one degree of freedom for each base-pair, while AMBER uses on average 90 degrees of freedom per base-pair. The water box also influences the DNA dynamics during a simulation.

In addition, within the equilibrium state $y_0 \approx 0$ Å and the breathing state $y_0 \approx 2 - 4$ Å we note a higher clustering of displacement values in SDE system than in AMBER. This is also due to the reduced number of degrees of freedom in SDE system over AMBER.

Figure 5.3 contains a comparison between the AMBER and SDE systems in terms of the binned frequency data over the 10ns simulations. In the closed state (at $y_0 = 0$ Å), the residence time is similar, however, we observe a reduction in the number of data points at the breathing barrier $\Delta B \approx 1$ Å (see Fig. 3.2 for definition) and an increased number of points for the bins corresponding to the breathing state at $y_0 = 2$ Å. This is counterbalanced by the residence time at $y_0 = 4$ Å, which is reduced in the SDE simulation compared to AMBER, and hence the total time spent breathing is similar, that is, 28.71% of the simulation time. Note that graphs such as Figure 5.3 depend on the width of bins chosen, using wider bins would increase the accuracy of the results on the vertical axis but result in a lower resolution of the detail of the closed and open states, that is a lower resolution on the horizontal axis. Similarly, it was noted in Figure 3.4

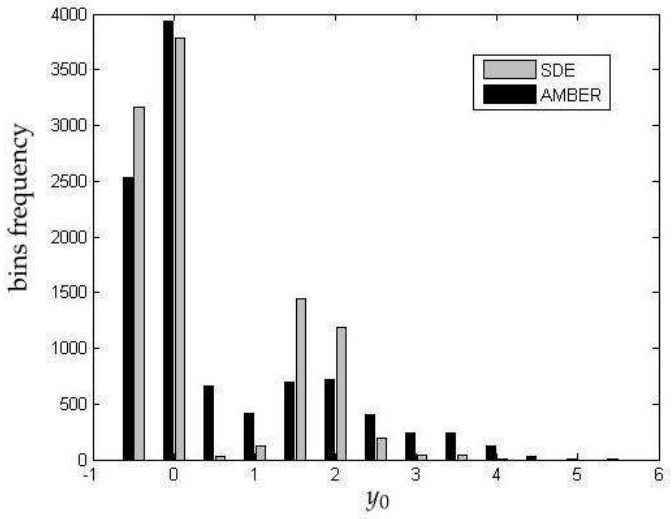

Figure 5.3: Illustration of the occupation of different y_0 positions (Å) for the breathing pair, obtained from the SDE and AMBER simulations using a bin size of $s = 0.5$, for a 30° undertwisted DNA sequence.

that the height of the breather barrier is dependent upon the width of the bins used, since the small time spent near the barrier means that there is a relatively low number of counts there and the relative errors are larger.

Comparing AMBER simulation from Figure 5.4, with the SDE simulation presented in Figure 5.5, both specific to a 32° undertwisted DNA sequence, we observe that although the length and frequency of breathing events is similar, there is an important difference between the two simulations. In AMBER simulation, the A-F pair spends a significant percentage of time in the second breathing state, while in the SDE system the time spent in this state is insignificant. This difference can be explained by the reduced number of degrees of freedom, which implicitly reduces the volume of space explored. SDE system must pass through the lower amplitude state to get to the higher amplitude state, whereas the extra degrees of freedom in AMBER mean that it may access the higher breathing state without even venturing into the lower amplitude state.

Analysing the DNA sequence for the 33° twist angle, we observe in Figure 5.6 the same three states explored by the breathing pair. Whilst the time spent breathing is almost the same as in the 30° twist angle case, the behaviour of the DNA sequence is different: the breathing events are longer and less frequent.

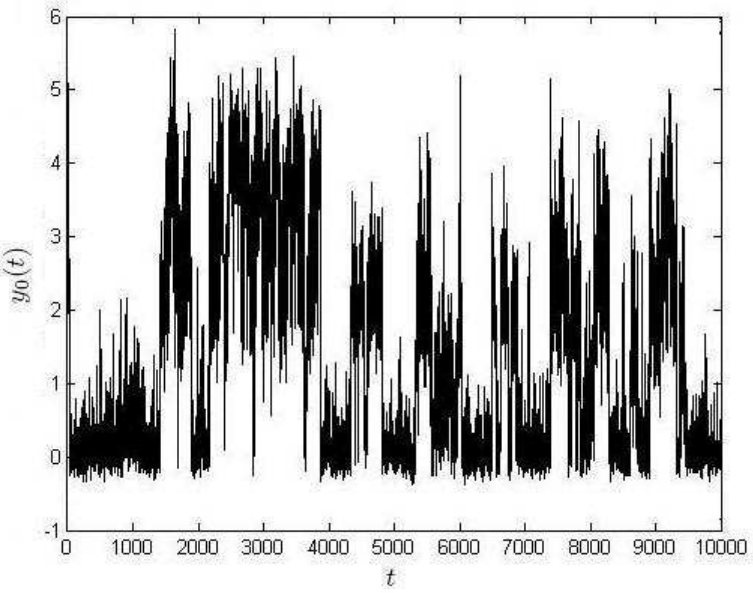

Figure 5.4: Graph of the displacement between bases of the breathing pair (y_0 in Å), plotted against time measured in ps, obtained from an AMBER simulation of 10 ns, for a 32° undertwisted DNA sequence.

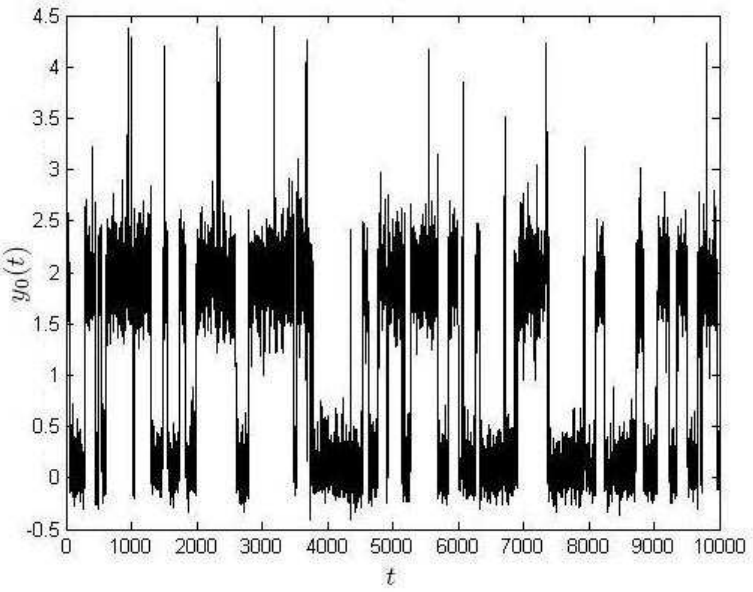

Figure 5.5: Illustration of the displacement (y_0 in Å) between the bases of the breathing pair over 10 ns, obtained using the SDE model for a 32° undertwisted DNA sequence. The parameter values are $C = 6$, $\epsilon = 3.3585$, $\epsilon_0 = 5.9770$, $k = 9.5585$, $\widehat{k} = 3.2132$, $\gamma = 131.0919$, while for E_0 the expression from Figure 4.10(a) was used.

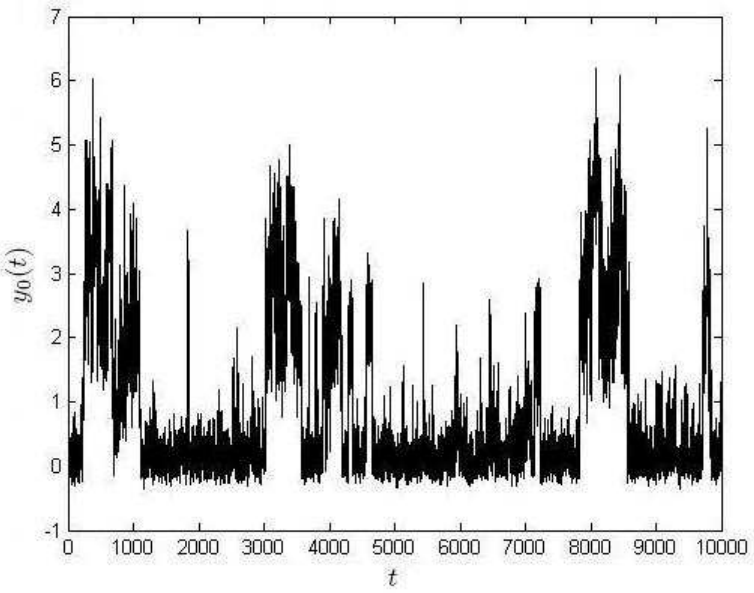

Figure 5.6: Graph of the displacement between bases of the breathing pair (y_0 in Å), plotted against time measured in ps, obtained from an AMBER simulation of 10 ns, for a 33° undertwisted DNA sequence.

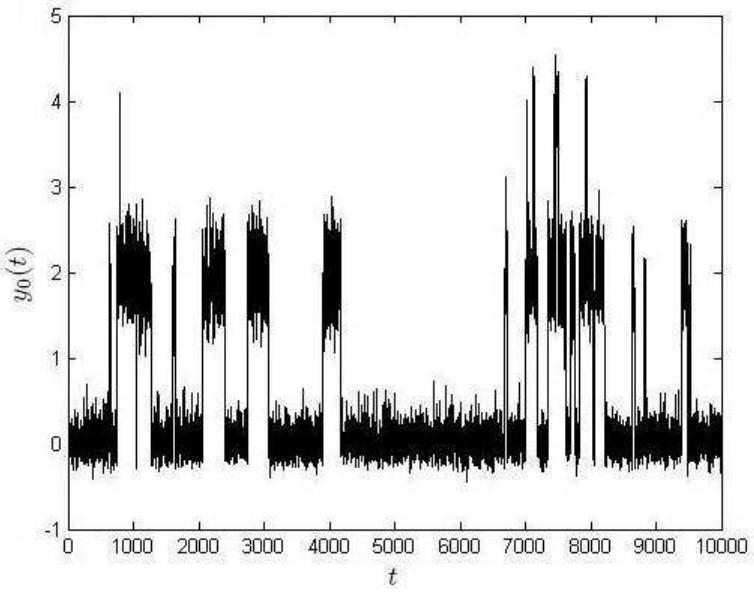

Figure 5.7: Illustration of the displacement (y_0 in Å) between the bases of the breathing pair over 10 ns, obtained using the SDE model for a 33° undertwisted DNA sequence. The parameter values are $C = 5.8$, $\epsilon = 3.3429$, $\epsilon_0 = 5.3214$, $k = 9.5374$, $\widehat{k} = 2.8261$, $\gamma = 135.5951$, while for E_0 the expression from Figure 4.11(a) was used.

The SDE simulation, presented in Figure 5.7, emphasizes that the results obtained using our SDE model agree with the MD simulations in length and frequency of breathing events. In both, closed and open state, the fluctuations are slightly smaller in the SDE model than in the full MD-AMBER simulation. This can be again attributed to the reduction in the number of degrees of freedom as one moves from an all-atom simulation to a mesoscopic model.

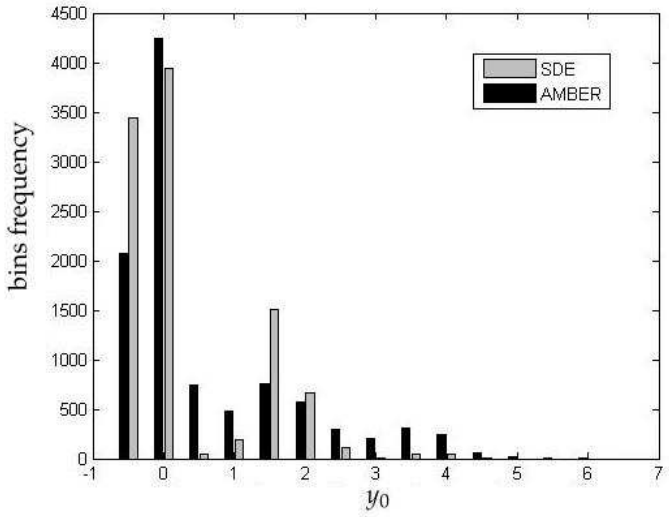

Figure 5.8: Illustration of the occupation of different y_0 positions (Å) for the breathing pair, obtained from the SDE and AMBER simulations using a bin size of $s = 0.5$, for a 33° undertwisted DNA sequence.

Figure 5.8 shows that the residence time in both open and closed states is larger in the SDE simulation than in AMBER, but the number of barrier crossings is higher in the case of AMBER simulation. This is due to the SDE simulation not exhibiting some of the very short breathing events observed in the AMBER simulation. However, overall the time spent breathing during the SDE simulation (25.74%) agrees well with the data obtained using AMBER – see Table 4.1.

The 34° and 35° undertwisted DNA sequences spend the least time in breathing states. In both cases, there are notable differences between the AMBER simulation, presented in Figure 5.9 and Figure 5.11, respectively, and the SDE simulation, from Figure 5.10 and Figure 5.12, respectively.

For the 34° twist angle, the two sets of data disagree in the time spent in the second breathing state (compare Figures 5.9 and 5.10), but agree in the range of values of the displacements from equilibrium – between -0.3 and 5.7 Å in

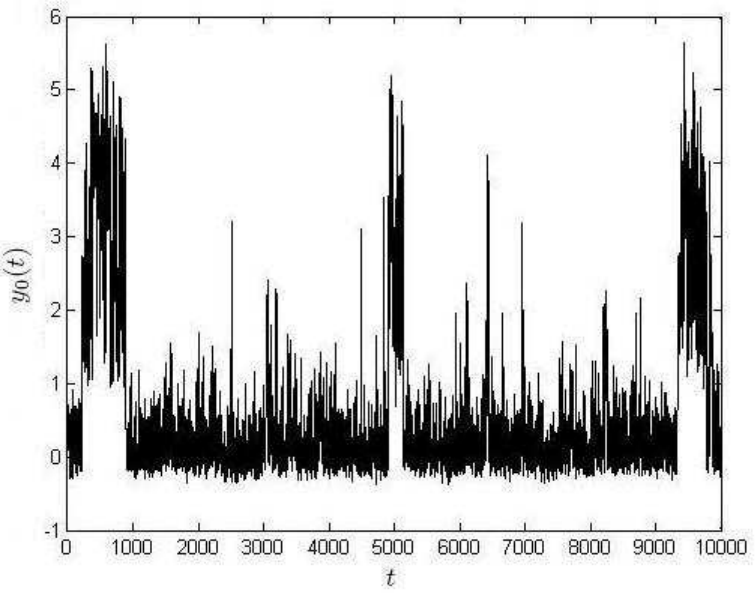

Figure 5.9: Graph of the displacement between bases of the breathing pair (y_0 in Å), plotted against time measured in ps, obtained from an AMBER simulation of 10 ns, for a 34° undertwisted DNA sequence.

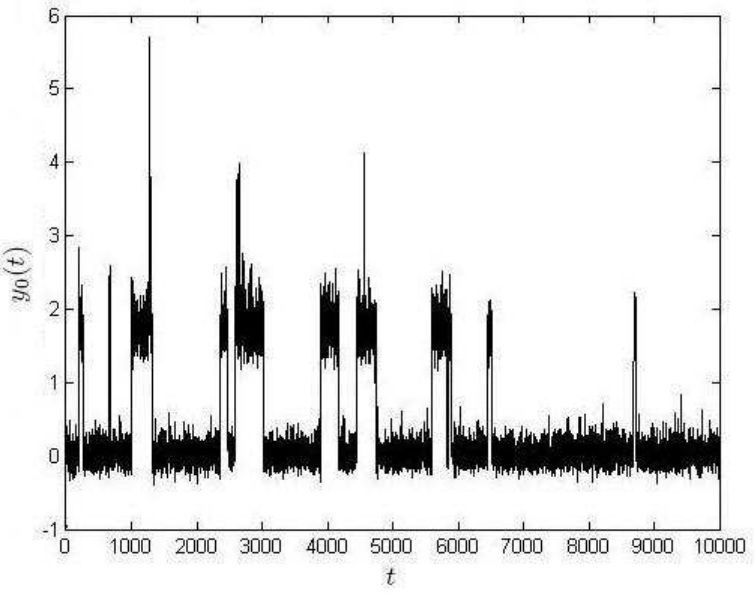

Figure 5.10: Illustration of the displacement (y_0 in Å) between the bases of the breathing pair over 10 ns, obtained using the SDE model for a 34° undertwisted DNA sequence. The parameter values are $C = 5.6$, $\epsilon = 3.3225$, $\epsilon_0 = 5.4843$, $k = 9.2678$, $\widehat{k} = 2.4625$, $\gamma = 145.6987$, while for E_0 the expression from Figure 4.12(a) was used.

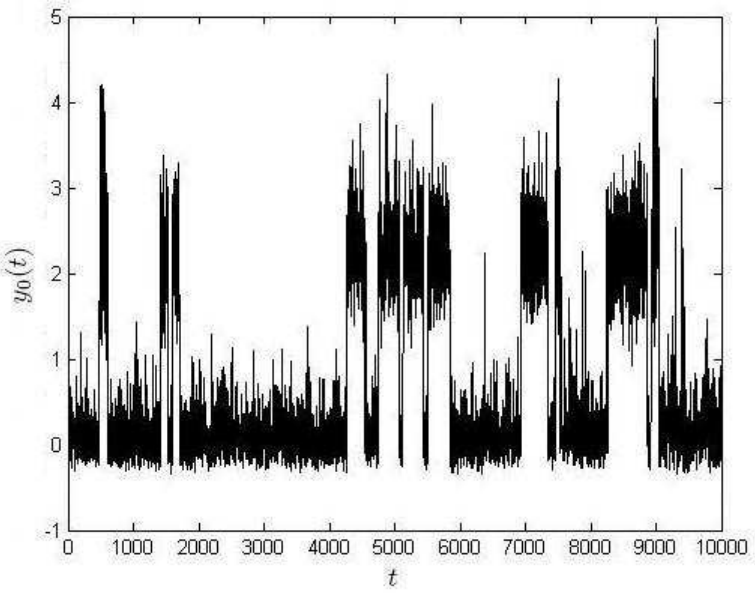

Figure 5.11: Graph of the displacement between bases of the breathing pair (y_0 in Å), plotted against time measured in ps, obtained from an AMBER simulation of 10 ns, for a 35° undertwisted DNA sequence.

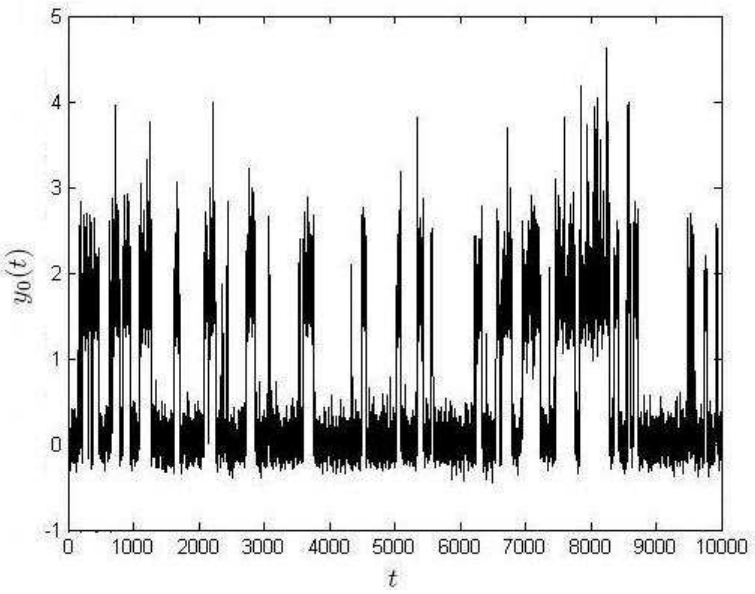

Figure 5.12: Illustration of the displacement (y_0 in Å) between the bases of the breathing pair over 10 ns, obtained using the SDE model for a 35° undertwisted DNA sequence. The parameter values are $C = 4.8$, $\epsilon = 3.3471$, $\epsilon_0 = 5.6744$, $k = 8.1819$, $\widehat{k} = 1.8256$, $\gamma = 149.5683$, while for E_0 the expression from Figure 4.13(a) was used.

both cases. Moreover, the AMBER simulation of 34° (Figure 5.9) contains three breathing events lasting about 800, 200 and 400 ps, respectively, as well as several very short breathing events. The SDE simulation (Figure 5.10) also contains several breathing events that are very short and five breathing events lasting on average 200 ps. This suggests that in the SDE simulation we have a higher breathing frequency, which is due to the breathing barrier ΔB being slightly lower.

Recall that the data was fitted to a short simulation which is representative of our AMBER simulation in terms of the proportion of time spent breathing. In order to obtain accurate results, the short simulations also have to agree in frequency of breathing and time spent in each of the breathing state. Indeed, for the 35° twist angle the two sets of data (shown in Figures 5.11 and 5.12) also agree in the range of values of the displacements and even in the time spent in the second breathing state, although they disagree in breathing frequency. The AMBER simulation (Figure 5.11) contains eleven breathing events including a few very short events, whilst the SDE simulation (Figure 5.12) contains many more short and very short breathing events.

As presented in Table 4.8, $\Delta B = 7.6100$, which is small compared to the breathing barrier values specific to other twist angles. This also requires more damping in the system to reduce the number of barrier crossings, which implies a decrease in C, as observed in Table 4.5. These two observations show once again how sensitive the parameters are to the details of the dataset.

5.2 Normally twisted DNA

Figure 5.13 shows an AMBER simulation of a DNA sequence specific to the typical twist of 36°. Note that the second breathing state is not reached as often as in the undertwisted case and most of the time spent breathing is in the first state (smaller values of y_0). In addition, we observe that the displacement from equilibrium takes values above 6 Å, which suggests that there might exist a third breathing state.

Analysing the SDE simulation presented in Figure 5.14, we again observe a

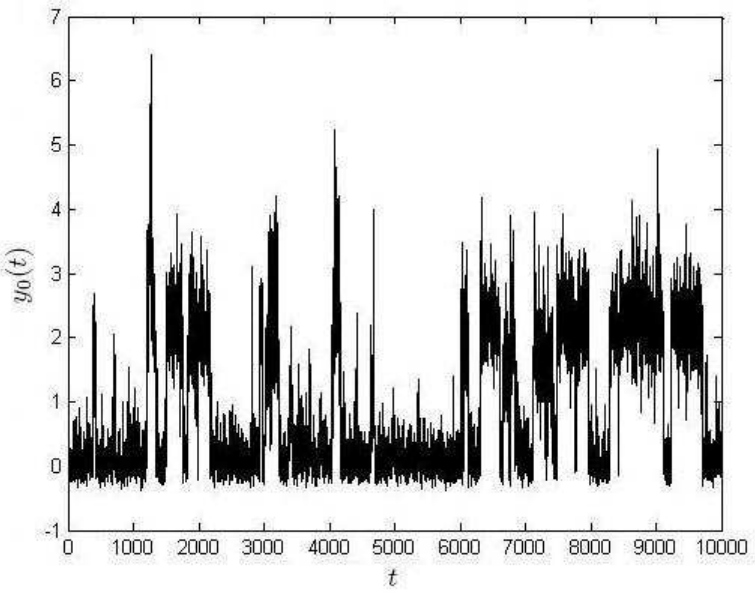

Figure 5.13: Graph of the displacement between bases of the breathing pair (y_0 in Å), plotted against time measured in ps, obtained from an AMBER simulation of 10 ns, for a 36° twisted DNA sequence.

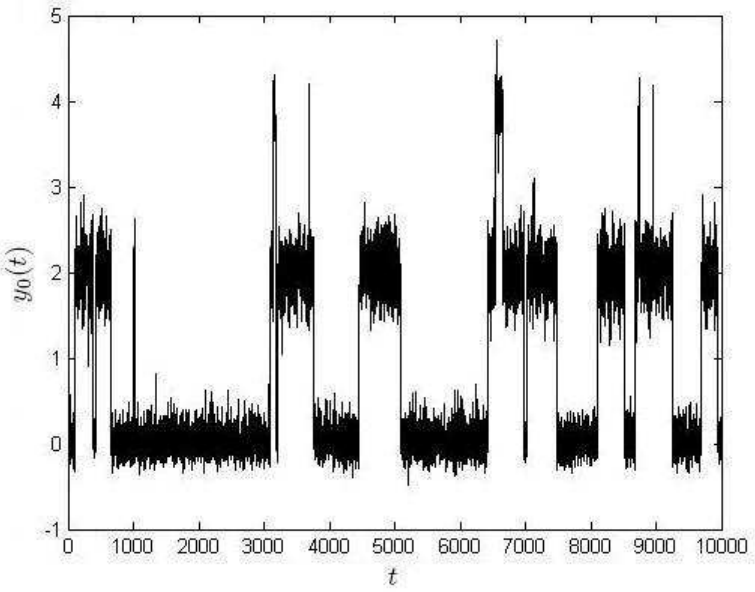

Figure 5.14: Illustration of the displacement (y_0 in Å) between the bases of the breathing pair over 10 ns, obtained using the SDE model for a 36° twisted DNA sequence. The parameter values are $C = 7$, $\epsilon = 3.3499$, $\epsilon_0 = 5.9238$, $k = 7.6577$, $\widehat{k} = 1.4307$, $\gamma = 165.4327$, while for E_0 the expression from Figure 4.14(a) was used.

slight reduction in the range of values from the AMBER simulation from Figure 5.13. Moreover, the SDE simulation is more regular, the three states being well defined, while in the AMBER simulation the degree of randomness seems to be larger. On the other hand, the breathing length and frequency is approximatively the same in both SDE and AMBER simulations.

Figure 5.14 also suggests the existence of a third open state. The breathing event taking place between the 6^{th} and 7^{th} nanosecond explores both open states, but the A-F base-pair also explores, for a very short period of time, a volume of space outside the three states already defined (one closed and two open states). Hence, we can redefine the possible states of the A-F base-pair, as follows:

- open state: between -0.3 and 1 Å
- first breathing state: between 1 and 3 Å
- second breathing state: between 3 and 5 Å
- third breathing state: between 5 and 7 Å

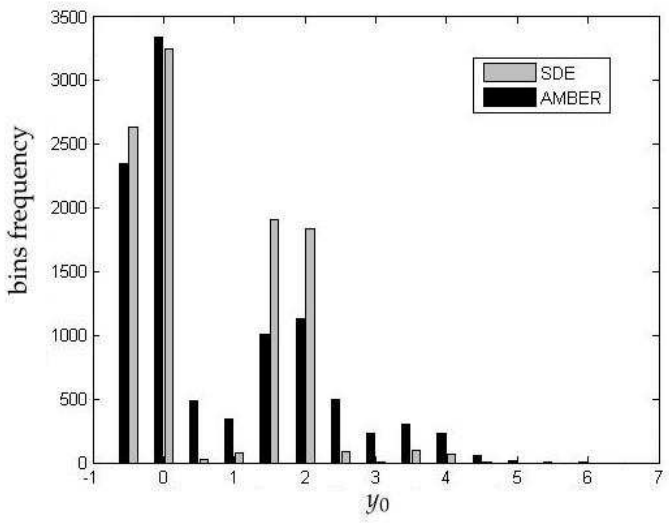

Figure 5.15: Illustration of the occupation of different y_0 positions (Å) for the breathing pair, obtained from the SDE and AMBER simulations using a bin size of $s = 0.5$, for a 36° twisted DNA sequence.

Comparing the results presented in Figure 5.15 with the undertwisted case (Figure 5.3), we observe an increased number of data points in the second breathing

state at $y_0 = 4$ Å. This increase occurs in both the AMBER and the SDE systems, though in all twist angles, there AMBER shows more time in the second breathing state than the SDE system. Even though the SDE simulation has a larger amount of data around the first breathing state, $y_0 = 2$ Å, the percentage of time spent in a breathing state is the same in both AMBER and SDE simulations, namely 40.95%.

5.3 Overtwisted DNA

For a 38° overtwisted DNA sequence, the time spent breathing represents more than 65% of the total time of a simulation, as shown in Figure 5.16, representing the AMBER simulation specific for this angle. An important proportion of this time is spent in the second breathing state, in contrast with the undertwisted and normally twisted DNA sequences, for which the breathing events were much shorter, although they were as frequent as in this case. More than that, this simulation shows that we indeed have a new open state around 6 Å and that long breathing events of 2 or 3 ns allow the breathing pair to explore the third breathing state.

The SDE simulation, presented in Figure 5.17, does not explore this third breathing state, but it confirms the regularity of the SDE simulations. More than that, it emphasizes that our system also allows to explore the second open state, when the data used for parameters fitting is representative for an AMBER simulation from all points of view.

For 38° of twist, Fig. 5.18 shows that more time is spent in the two breathing states at $y_0 = 2$ Å and $y_0 = 4$ Åin the SDE simulation than in the AMBER data (Fig. 5.16). Less data points are observed near the breathing barriers at $y_0 = 1$ Å and $y_0 = 3$ Å. Even though this implies a small reduction in breathing frequency, that is, 9 breathing events in SDE simulation instead of 12 as in AMBER, the general DNA behaviour is preserved. Compared to the undertwisted and normally twisted DNA sequence, in both AMBER and SDE systems we have a high residence time in the second breathing state ($y_0 = 4$ Å).

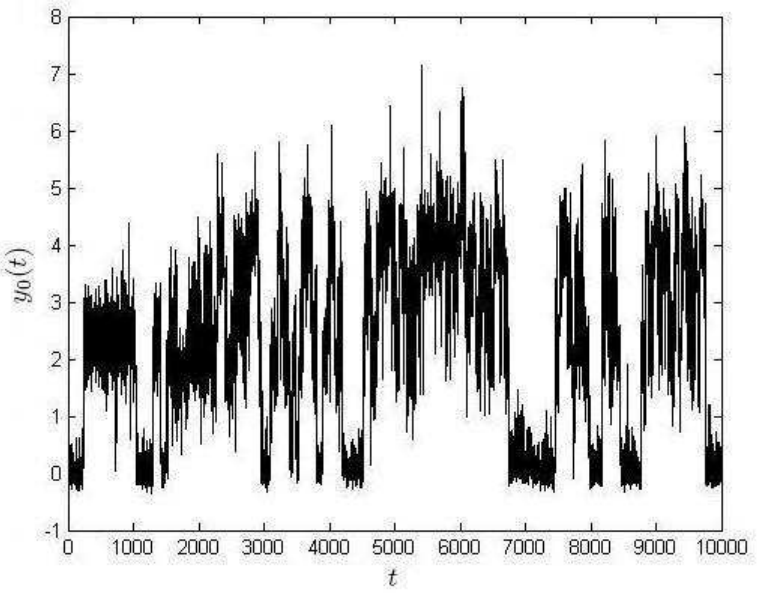

Figure 5.16: Graph of the displacement between bases of the breathing pair (y_0 in Å), plotted against time measured in ps, obtained from an AMBER simulation of 10 ns, for a 38° overtwisted DNA sequence.

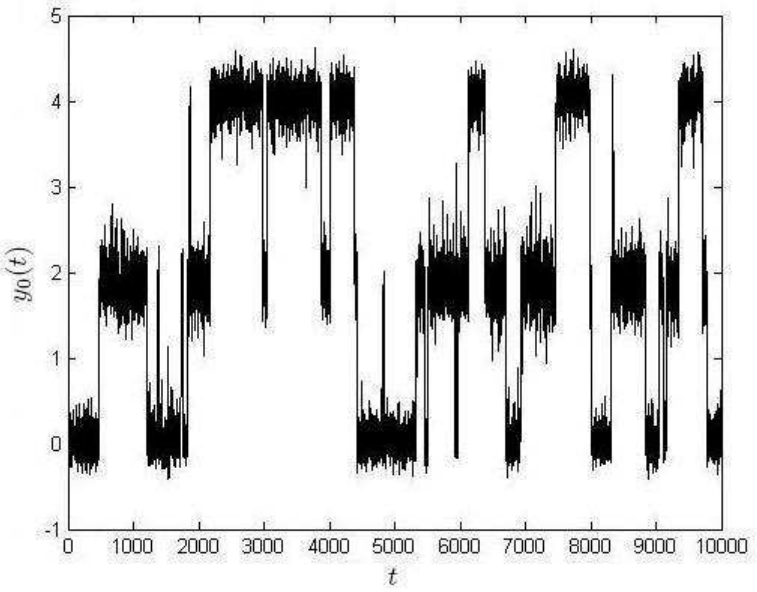

Figure 5.17: Illustration of the displacement (y_0 in Å) between the bases of the breathing pair over 10 ns, obtained using the SDE model for a 38° overtwisted DNA sequence. The parameter values are $C = 7.25$, $\epsilon = 3.3511$, $\epsilon_0 = 6.8702$, $k = 8.1438$, $\widehat{k} = 2.1462$, $\gamma = 139.0797$, while for E_0 the expression from Figure 4.15(a) was used.

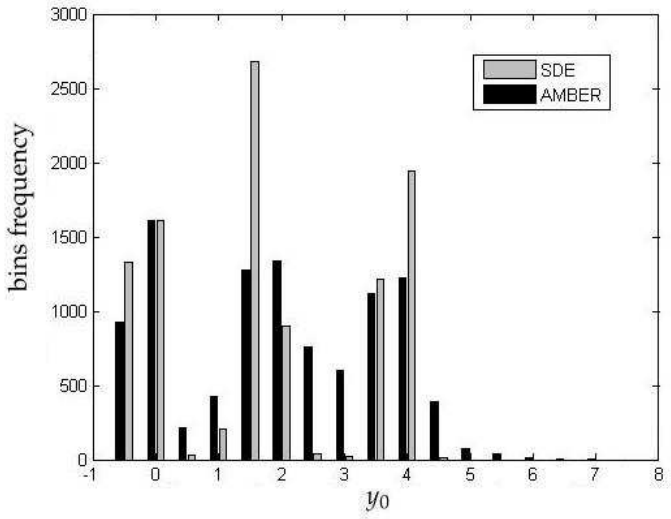

Figure 5.18: Illustration of the occupation of different y_0 positions (Å) for the breathing pair, obtained from the SDE and AMBER simulations using a bin size of $s = 0.5$, for a 38° overtwisted DNA sequence.

Finally, the 40° overtwisted DNA AMBER simulation illustrated in Figure 5.19 emphasizes that long breathing events are specific to overtwisted DNA sequences, while the short events occur in undertwisted DNA sequences. In addition, these simulations also show that spending more time breathing allows the A-F pair to explore larger volumes of space. The pair thus spends more time in the second open state. Compared to an undertwisted angle, overtwisted sequences are also able to explore a third open state for short intervals of time.

Being an extreme twist angle, one might expect a 40° overtwisted sequence to spend more time breathing than the other overtwisted angle analysed, but Figure 4.5 shows that the along-chain interactions (k,$\widehat{k}$) are *stronger* in this case (40°) than for 38°. Hence, the range of motion (for a given energy) of the bases of the breathing pair is reduced, due to the stronger covalent bonds.

Figure 5.20 shows the SDE simulation for a 40° overtwisted DNA sequence. In contrast with the 38° twist angle, it shows that our model is also capable of exploring a third open state (5 to 7 Å). It also confirms that the SDE simulations are more regular than AMBER simulations.

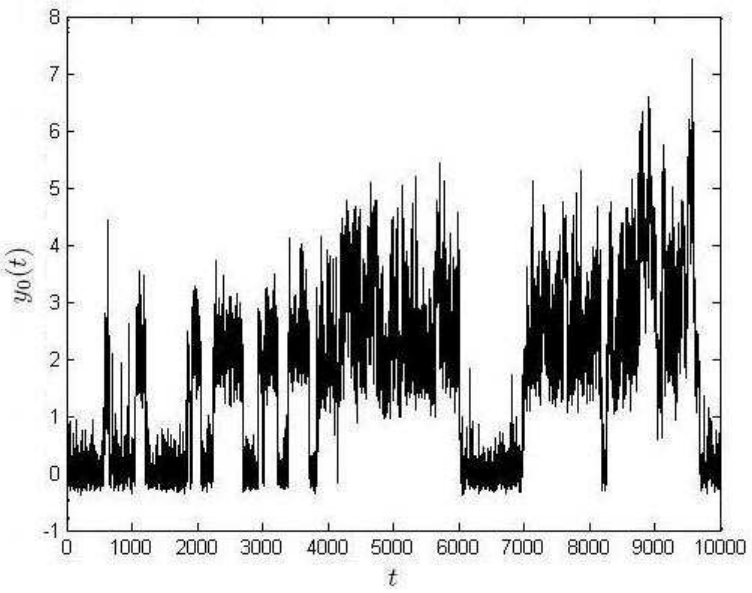

Figure 5.19: Graph of the displacement between bases of the breathing pair (y_0 in Å), plotted against time measured in ps, obtained from an AMBER simulation of 10 ns, for a 40° overtwisted DNA sequence.

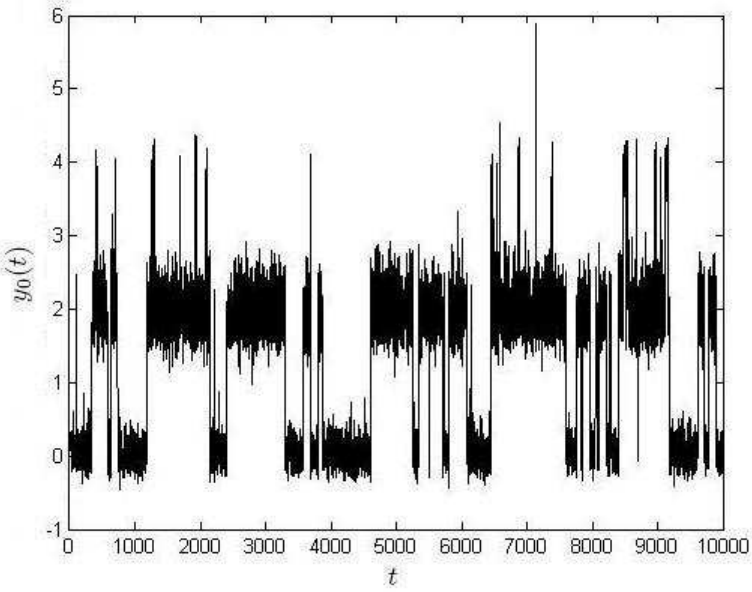

Figure 5.20: Illustration of the displacement (y_0 in Å) between the bases of the breathing pair over 10 ns, obtained using the SDE model for a 40° overtwisted DNA sequence. The parameter values are $C = 8$, $\epsilon = 3.3550$, $\epsilon_0 = 6.1750$, $k = 19.5297$, $\widehat{k} = 2.6341$, $\gamma = 132.0731$, while for E_0 the expression from Figure 4.16(a) was used.

5.4 AMBER-SDE comparison

A standard technique of checking the degree of randomness in a system is to compare the expected value with the standard deviation of a measurable quantity. In our case, the frequency of breathing events can offer such a measure to test the randomness degree of both AMBER ans SDE simulations.

Let T_i be the time measured between the end of breathing event i and the beginning of breathing event $i+1$ and let $T = \{T_i\}_{i=1,n}$ be the set of such measurements. Denote by $\mathbb{E}[T] = \frac{1}{n}\sum_{i=n}^{n} T_i$ the expected value of T (also known as mean value) and by $\sigma(T) = \sqrt{\frac{1}{n}\sum_{i=n}^{n}(T_i - \mathbb{E}[T])^2}$ the standard deviation of T. Then,

- $\mathbb{E}[T] \approx \sigma(T)$ implies a random process
- $\mathbb{E}[T] > \sigma(T)$ implies a regular process
- $\mathbb{E}[T] < \sigma(T)$ implies a clustered process

Figure 5.21 explains how breathing events are distributed in each of the three cases.

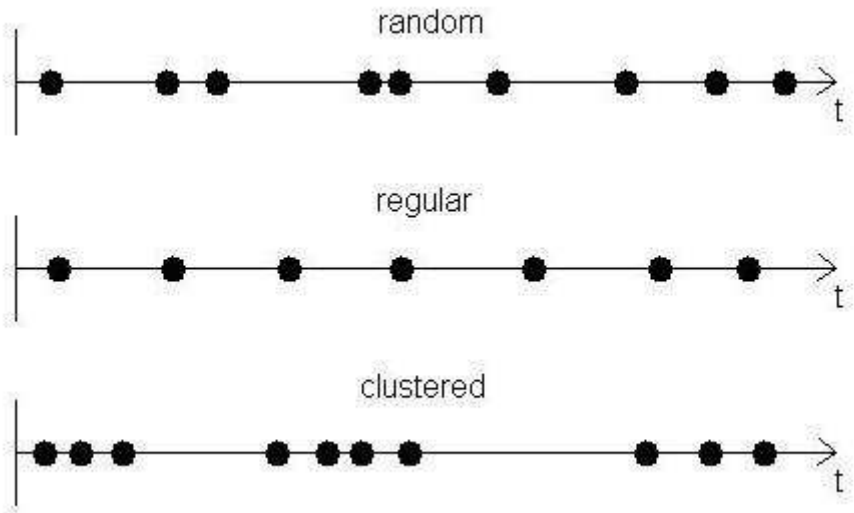

Figure 5.21: Illustration of breathing events distribution for the three examples of processes.

Observe that for this analysis it is very important how we define a breathing event. If we consider each barrier crossing to be such an event, then in the case of AMBER simulations that are more noisy we risk obtaining the wrong answer. Passing the top of the breathing barrier is not enough to consider a breathing event takes place. We consider that breathing means reaching the local minima of the open state. In other words, the A-F pair displacements from equilibrium have to pass the threshold value of 1.9 Å, for which the total energy is minimum during a breathing event. Moreover, breathing occurs on the nanosecond timescale, which means that very short events, of up to 10 ps, are ignored, being determined by the noise from our system. Hence, short breathing events can be considered having causes other then the biological ones and can be ignored during the randomness analysis.

Next, we have the opposite situation, when the breathing barrier is crossed backwards, from an open state to the closed state. We consider a breathing event does not end, unless the minimum energy point of 0 Å displacements is reached. Also, if the time spent between two breathing events is less than 10 ps, we consider them as being just one breathing event, based on the same considerations as in the case when short breathing is ignored.

Figure 5.19 clarifies the definition of breathing. The important events last in order 135, 143, 203, 445, 309, 327, 2197, 1183 and 1426 ps, respectively. Between the first two breathing events, we observe several barrier crossings, but only in two of the cases is the value of 1.9 Å reached, these events lasting 7 and 13 ps, respectively. According to the definition of breathing, we ignore the first one, but take into consideration the latter. Analogously, the longest breathing event of 2197 ps also explores the closed state, but just for 1 ps, hence is considered to be a single event. On the other hand, the closed state is also visited for a short period between the last two breathing events. This visit lasts more than 10 ps and we consider it separating the two breathing events involved.

Computing the required expected values and standard deviations of AMBER and SDE simulations, for each twist angle previously analysed, we obtain the values given in Table 5.1. Surprising at it might seem, not all AMBER simulations are random. The angles for which the overall time spent breathing is lower and which have a lower breathing frequency (see Tables 5.2 and 5.3 for

details) are regular, for example, the cases 33° and 34° cases, or clustered, as in the 35° twist angle. The SDE simulations are all regular, except for the 33° twist angle, which is clustered, as observed in Figure 5.7.

Twist angle	$\mathbb{E}[T_{AMBER}]$	$\sigma(T_{AMBER})$	$\mathbb{E}[T_{SDE}]$	$\sigma(T_{SDE})$
30°	347.1500	345.2123	579.1000	496.6771
32°	222.8636	218.9180	233.4000	199.1919
33°	968.4286	922.9846	909.0000	922.8703
34°	899.2222	745.8227	731.4444	626.4760
35°	566.2000	795.1886	262.3333	225.0561
36°	400.9286	402.4417	807.5614	670.3514
38°	202.7273	202.5429	331.4286	313.0313
40°	298.6364	292.5520	231.8571	203.7003

Table 5.1: Expected values and standard deviations of time elapsed between two breathing events, both measured in ps.

However, note that $\mathbb{E}[T]$ and $\sigma(T)$ have the same order of magnitude for both AMBER and SDE simulations. We can use Pearson's chi-square test, for example, to test the fit of a distribution. In our case, this requires computing the value of

$$\chi^2 = \frac{(\sigma(T) - \mathbb{E}[T])^2}{\sigma(T)^2} = (1 - \mathbb{E}[T]/\sigma(T))^2. \tag{5.4.1}$$

Given that the range of values of $\mathbb{E}[T]/\sigma(T)$ is between 0.71 and 1.20 for AMBER simulations, while for the SDE simulations the range is 0.98 and 1.20, we obtain that $\chi^2 \leq 0.1$ in both cases. This suggests that all AMBER and SDE simulation are random. The small differences between the analysed simulations might be due to the number of breathing events, which is rather small, as can bee seen in Table 5.3.

Table 5.2 contains the average values of lengths of breathing events. We observe that there are significant differences between the AMBER and SDE simulations, which can be explained by the low number of breathing events sampled in the two models – see Table 5.3.

Twist angle	$\mathbb{E}[l_{AMBER}]$	$\mathbb{E}[l_{SDE}]$
30°	161.7500	275.0909
32°	237.0500	266.4500
33°	308.2222	226.0000
34°	249.8333	191.4000
35°	208.0000	114.2222
36°	217.6471	523.8750
38°	608.0833	872.8750
40°	638.1000	407.2667

Table 5.2: Average length of breathing events, measured in ps.

Note that there are also significant differences in the number of breathing events between the two simulation methods (see Table 5.3), since the parametrisation described earlier aimed to match the proportion of time spent breathing. Furthermore, note that, there might also be differences between different AMBER simulations of the same twist angle.

Twist angle	AMBER	SDE
30°	20	11
32°	19	21
33°	10	13
34°	10	10
35°	12	25
36°	15	8
38°	12	9
40°	10	15

Table 5.3: Number of breathing events specific to each sequence analysed.

For some twist angles (32°, 33°, 34° and 38°) the number of breathing events is similar for the two models, while for the 32° undertwisted DNA sequence we have the same number of breathing events, similarly distributed in time (see Table 5.1), but having different average length of breathing events. During the AMBER simulation of Figure 5.4 we have a 1716 ps long breathing event, while

during the SDE simulation from Figure 5.5 the maximum length is 966 ps.

In addition, if we take into account just the first 5 ns of the 32° AMBER simulation, we have six breathing events, with lengths of 30, 24, 483, 1716, 195 and 241 ps, and an average of 448.1667 ps, while for the last 5 ns we have 14 breathing events with an average length of 146.5714 ps. This not only explains the large difference between the two models in average breathing length, for a 32° overtwisted DNA sequence (see Table 5.2), but it also means that breathing length and frequency analysis is not a criteria for our SDE model strength.

As already mentioned, in order to have most of the models' features similar, we aim to fit data that agrees (in time spent in each state and breathing frequency) with longer AMBER simulations. In our case the representative data selected only respects the percentage of time spent breathing. Indeed, comparing the time spent breathing in the sequences simulated above by the two models (AMBER and SDE), for each twist angle, we observe small differences, but none larger than 6% – see Table 5.4 for details. This suggests that the SDE simulations are close to all-atom MD simulations, from breathing time point of view. Note that the values for the AMBER simulation are not the ones from the last column of Table 4.1, since for consistency, here we present 10 ns simulation intervals that do not start or end during a breathing event.

Twist angle	AMBER	SDE
30°	28.71%	30.08%
32°	46.36%	52.01%
33°	29.37%	25.74%
34°	17.61%	19.39%
35°	29.57%	32.73%
36°	37.75%	40.95%
38°	72.56%	70.22%
40°	62.71%	68.69%

Table 5.4: Percentage of time spent breathing in the analysed AMBER and SDE simulations.

In conclusion, the analysis of parameter values from Chapter 4 and the compar-

ison between AMBER and SDE simulations show how important it is to select the data which best reflects the DNA properties for each twist angle. All-atom MD simulations based on thousands of degrees of freedom are more accurate then reduced mesoscopic models, but the latter models allow consistent analysis of different measurable quantities, when their parameter values are correctly determined. Moreover, mesoscopic models, such as our SDE system, reduce the time needed to simulate a DNA system and thus, are able to predict its behaviour for longer time periods.

5.5 Long-time SDE simulation

Given the capacity of our SDE system to simulate with accuracy breathing in a DNA sequence, we have decided to also study the long-time dynamics in our reduced DNA model.

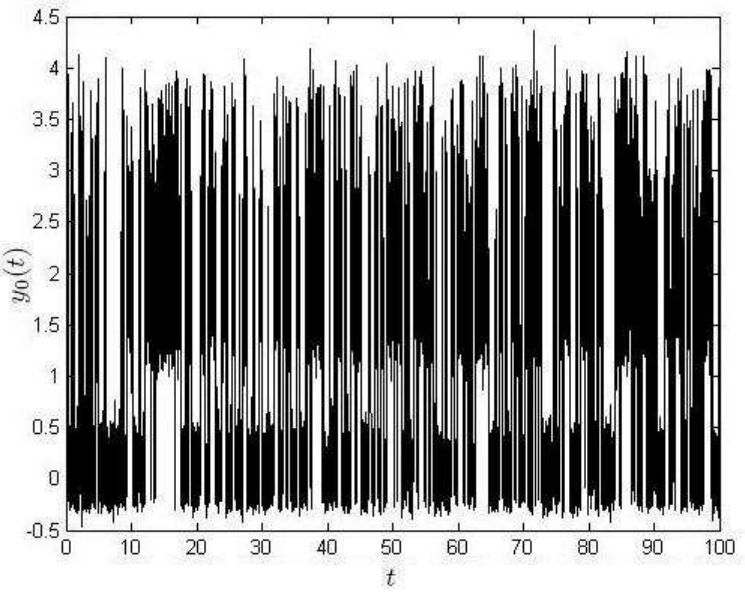

Figure 5.22: Illustration of the displacement (y_0 in Å) between the bases of the breathing pair over 100 ns, obtained using the SDE model for a 30° undertwisted DNA sequence. The parameter values are $C = 6.5$, $\epsilon = 3.4074$, $\epsilon_0 = 5.6285$, $k = 10.6536$, $\widehat{k} = 3.6851$, $\gamma = 120.0904$, while for E_0 the expression from Figure 4.7 was used.

One might expect to obtain from this more details about the time needed to

obtain a bubble in our DNA sequence or about the time needed to emerge the DNA melting point. We have continued the SDE simulation of a 30° under-twisted DNA from Figure 5.2 with another 100 ns, but this simulation could not answer to any of these question. However, as can be seen in Figure 5.22 the time spent in breathing state increased considerably compared to the previous 10 ns. More precisely, for the first 10 ns of this new simulation 31.84% of the time is spent breathing, wich represents an increase of only 1.76%. However, after 50 ns this percentage increases to 48.03%, while for the full simulation this value becomes 51.01% (almost twice bigger than at the beginning of the simulation). This might suggest that for longer SDE simulations we could observe a full separation of the A and F bases. Another important observation concerns the length of the breathing events: in Figure 5.2 we observe breathing events of at most 1 ns, while in Figure 5.22 longer breathing events of 2 ns can be observed.

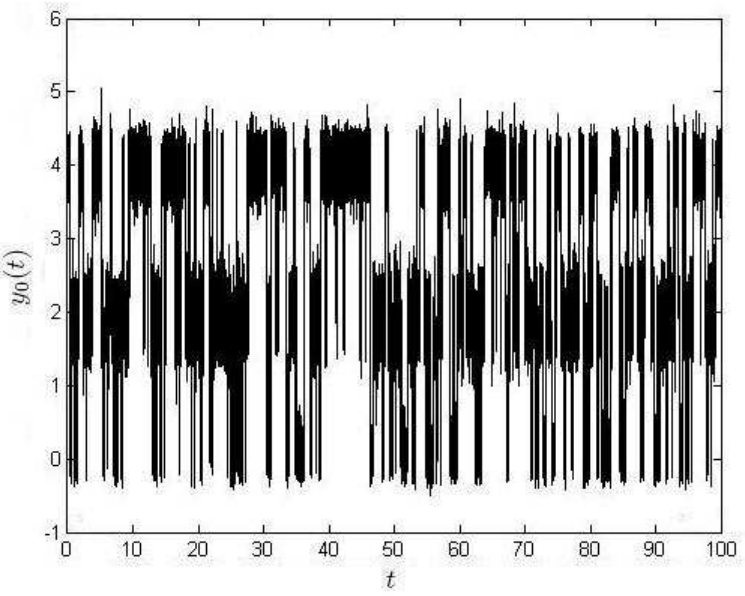

Figure 5.23: Illustration of the displacement (y_0 in Å) between the bases of the breathing pair over 100 ns, obtained using the SDE model for a 38° overtwisted DNA sequence. The parameter values are $C = 7.25$, $\epsilon = 3.3511$, $\epsilon_0 = 6.8702$, $k = 8.1438$, $\widehat{k} = 2.1462$, $\gamma = 139.0797$, while for E_0 the expression from Figure 4.15(a) was used.

Given that the 38° overtwisted DNA sequence spends more time breathing than any of the other DNA sequences analysed, we have continued the simulation

presented in Figure 5.17 for another 100 ns as well. The same behaviour as in the undertwisted case was observed (see Figure 5.23), that is, the total period spent breathing increased with time from 79.72% during the first 10 ns to an average of 81.52% for the full 100 ns simulation, which represents an increase of about 10% compared to the initial simulation from Figure 5.17. Moreover, the longest breathing event in Figure 5.23 is of about 9 ns compared to the 3 ns breathing event observed in Figure 5.17. This result also indicates that much longer simulations could offer more information about DNA properties. The increase in breathing length sustain the idea of bubble generation: the longer a breathing event is, the higher the chances to obtain a bubble are.

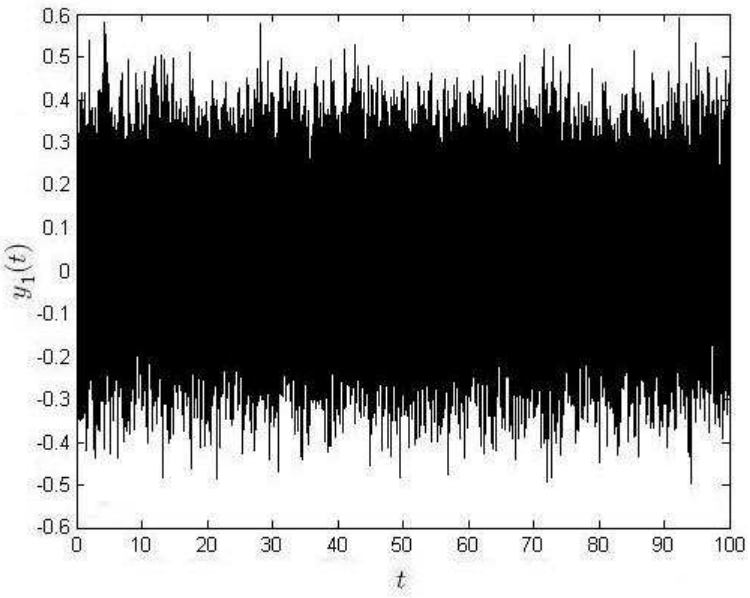

Figure 5.24: Illustration of the displacement (y_1 in Å) between the bases of a nonbreathing pair over 100 ns, obtained using the SDE model for a 38° overtwisted DNA sequence. The parameter values are $C = 7.25$, $\epsilon = 3.3511$, $\epsilon_0 = 6.8702$, $k = 8.1438$, $\widehat{k} = 2.1462$, $\gamma = 139.0797$, while for E_0 the expression from Figure 4.15(a) was used.

On the other hand, the small difference of only 1.80% between the time spent breathing in the first part of the simulation presented in Figure 5.23 and the total breathing time of this simulation might suggest that due to the non-defective bases from our DNA sequence, bubble generation, for example, might be inaccessible to our system, since these bases might never open. In fact, the evolution

in time of $y_1(t)$ represented in Figure 5.24 confirms this assumption and suggests that the answer to such questions might be offered by further analysis of our SDE system.

5.6 Summary

Comparing the AMBER and the SDE simulations, we observe a reduction in the range of values of the displacements from equilibrium specific for the A-F base-pair. The difference is due to the reduced number of degrees of freedom in our SDE model. Also, analysing expected value of the time spent between two breathing events, we reach the conclusion that the SDE simulations are more regular when compared to AMBER results, but in both case we can classify the simulations as being random. Finally, we present the DNA dynamics in two SDE simulations of 100 ns.

Part II

System Analysis

CHAPTER 6

Methods for Analysing Hamiltonian Systems

The dynamics of Hamiltonian systems still represent a challenge for scientists. Experiments or all-atom molecular dynamics (MD) simulations give most of the information needed to analyse such systems, but, as discussed in Chapter 2, the time required to generate a representative set of data is of the order of weeks or months. Reducing the system complexity, by considering, for example, a reduced mesoscopic model, is useful in many cases. Such simple models can be close to MD simulations or experiments, but lose some of the system features. For example, our SDE mesoscopic model preserves the general DNA behaviour, but cannot offer any information about the trajectory or velocity of each atom in the system.

There exist several methods to analyse, on one hand, how close two different models are and, on the other hand, the properties that a system possesses. In this chapter, we focus on some of these methods. We start with principal component analysis (PCA), which is an analysis tool, useful for determining the quantities in a system with high variances. Next, we describe the autocorrelation function, followed by the normal mode representation of Hamiltonian systems. We also show how the normal modes and the specific frequencies can be determined using the Fourier Transform. Finally, we consider a simple system consisting of four particles, for which we apply the analytic methods described herein.

6.1 Principal Component Analysis

Principal Component Analysis (PCA) [126] is a simple way to reduce the dimension of complex datasets, its main goal being to reveal a simplified structure with the same properties as the initial one. This nonparametric method is helpful for extracting relevant information from different types of datasets.

The PCA method, invented in 1901 by Karl Pearson, is used for data analysis in different domains, such as microbiology, for example, as explained by Zacharias in [134]. Yet, PCA is mostly applied in exploratory data analysis or for constructing predictive models. Eriksson et al. [47] give an introduction to PCA, for non-specialists in mathematics and linear algebra. They consider that a system can be characterised through a set of observations. Each observation contains information about some measurable quantities, such as pressures, temperatures or spatial coordinates. The multivariate data table, obtained from these observations, is then represented using PCA as a low-dimensional space, consisting of at least two components.

6.1.1 Data pre-treatment

First of all, the numerical range of the quantities analysed may differ. A quantity with a large range has a large variance, while the ones with small ranges have small variances. PCA tries to determine the directions with maximum variance, hence quantities with larger variances are preferentially selected by PCA over the others.

Let X be the set of observations, N the number of observations and M the number of quantities analysed in each observation. Hence, X can be seen as an $N \times M$ matrix, that is,

$$X = \begin{pmatrix} X_1(t_1) & X_2(t_1) & ... & X_M(t_1) \\ X_1(t_2) & X_2(t_2) & ... & X_M(t_2) \\ ... & & & \\ X_1(t_N) & X_2(t_N) & ... & X_M(t_N) \end{pmatrix}, \tag{6.1.1}$$

where each row is an observation, denoted by $\mathbf{X}(t_n)$, with $1 \leq n \leq N$ and $t_1 < t_2 < ... < t_N$ are equally spaced points in time. In this form, each column of X represents data specific to one of the quantities measured.

Note that we might have different dimensions for each measurable quantity analysed. PCA requires the computation of the covariance matrix C_X (of size $M \times M$), which will be defined later and which involves the dot product between different columns of X. Hence, this dot product involves different quantities with different dimensions. Thus, we need to apply some transformations to our data to obtain a nondimensionalised form that allows the assumptions considered important to be verified. System nondimensionalisation may also give a more unbiased analysis.

Data can be scaled using several transformations, but the division of each column i of X by the standard deviation of $\{X_i(t_n)\}_{n=1}^N$ is the easiest way to solve both requirements: diminishing predominance quantities with large variances and system nondimensionalisation. The standard deviation of X_k is defined to be $\sigma_k = \sqrt{\frac{1}{N}\sum_{i=1}^N (X_k(t_i) - \langle X_k \rangle)^2}$, where $\langle X_k \rangle = \frac{1}{N}\sum_{i=1}^N X_k(t_i)$ is the mean of the column $\{X_i(t_n)\}_{n=1}^N$. Then, the column vector $\frac{1}{\sigma_k}X_k$ has unit variance and applying this scaling for each column of X we obtain unit variance data. More than that, the standard deviation has the same dimension as the quantity for which it is defined, hence unit scaling also ensures the system becomes nondimensional.

Next, we apply data centering, where the mean value of each quantity is subtracted from the matrix of data. This enable us to determine the orthonormal vectors produced by PCA, as well as the data distribution. In addition, the mean value is needed for unit variance scaling: this transformation involves data mean when the standard deviations of the quantities are determined. Note that the order in which the two transformations (data centering and scaling to unit variance) are applied does not influence the final form of the data.

6.1.2 PCA methodology

The goal of PCA is to find the directions, in an M-dimensional space, that approximate the data as closely as possible in the least squares error sense. In

other words, if we analyse noisy data, as in the case of our DNA dynamics trajectory, we are looking for a basis that allows us to rewrite the data in a way that filters the noise without affecting the system properties.

In what follows, we consider the dataset X to be centered. Moreover, each row of X is represented with respect to a canonical basis $\{\mathbf{b_1}, ..., \mathbf{b_M}\}$, where $\mathbf{b_i}$ is a column vector having all elements equal to 0, except the i^{th} entry, which is 1. Let B be the matrix associated to the canonical basis, more precisely, containing on each column one of the basis elements, that is, $B = I_M$, where I_M represents the identity matrix of order M. PCA is equivalent to finding a change of basis from B to another orthogonal basis $\{\mathbf{v_1}, ..., \mathbf{v_M}\}$, with associated matrix V. This change of basis is made assuming that the observed data X is a linear combination of the columns of V. Considering Y to be the data expressed in terms of the basis V, we have

$$\mathbf{X}(t_n) = (X_1(t_n)\ X_2(t_n)\ ...\ X_M(t_n)) = \sum_{i=1}^{M} Y_i(t_n)\mathbf{v_i}^T, \tag{6.1.2}$$

where $\mathbf{v_i}^T$ denotes the transpose of $\mathbf{v_i}$.

Note that $\mathbf{X}(t_n)V = (\langle \mathbf{X}(t_n), \mathbf{v_1}^T \rangle, ..., \langle \mathbf{X}(t_n), \mathbf{v_M}^T \rangle)$, where $\langle \mathbf{a}, \mathbf{b} \rangle$ represents the dot product between $\mathbf{a}$ and $\mathbf{b}$. On the other hand, assuming the columns of V are unit vectors, the projection of $\mathbf{X}(t_n)$ onto $\mathbf{v_i}^T$ is

$$pr_{\mathbf{v_i}^T}\mathbf{X}(t_n) =< \mathbf{X}(t_n), \mathbf{v_i}^T > \mathbf{v_i}^T, \forall i = 1, M \tag{6.1.3}$$

and

$$\mathbf{X}(t_n) = \sum_{i=1}^{M} pr_{\mathbf{v_i}^T}\mathbf{X}(t_n) = \sum_{i=1}^{M} < \mathbf{X}(t_n), \mathbf{v_i}^T > \mathbf{v_i}^T, \tag{6.1.4}$$

which, based on (6.1.2) and the orthogonality of vectors from V, implies

$$XV = Y. \tag{6.1.5}$$

Observe that if a matrix V contains on columns the elements of a basis (not necessarily orthogonal) that spans the same space as the canonical basis and Y is the data representation with respect to V, then we have $\mathbf{X}(t_n) = \sum_{i=1}^{M} \alpha_i \mathbf{v_i}^T$, where $(\alpha_1, ..., \alpha_M) = \mathbf{Y}(t_n)$. Also $X_j(t_n)$ can be written as $X_j(t_n) = \sum_{i=1}^{M} \alpha_i V_{i,j}$.

In other words, we obtain $\mathbf{X}(t_n) = \mathbf{Y}(t_n)V^T$ equivalent to $\mathbf{X}(t_n)(V^T)^{-1} = \mathbf{Y}(t_n)$, and finally, we obtain

$$X(V^T)^{-1} = Y. \tag{6.1.6}$$

But, if V is an orthogonal matrix of unit vectors, then $VV^T = I_M$, which implies $V^{-1} = V^T$. Thus, for any orthonormal basis V (6.1.5) and (6.1.6) are equivalent and imply that the matrix containing on each column the vectors of the new orthogonal basis is the transformation matrix that maps X into Y.

In conclusion, finding the new basis means finding the transformation matrix, based on some properties of the result Y that we want to achieve. We mentioned that, in case of noisy data, PCA separates the noise and the deterministic data. This task is achieved by determining the directions with the highest variance, the rest of data being considered noise. PCA transforms a set of correlated variables into a set of uncorrelated variables, thus we are looking for a basis V for which Y^TY is a diagonal matrix.

Let C_Y be the covariance matrix of Y, that is

$$C_Y = \frac{1}{N}Y^TY. \tag{6.1.7}$$

Using (6.1.5) we obtain

$$\begin{aligned} C_Y &= \frac{1}{N}(XV)^TXV \\ &= \frac{1}{N}V^TX^TXV \\ &= V^T\left(\frac{1}{N}X^TX\right)V \\ &= V^TC_XV, \end{aligned}$$

where C_X is the covariance matrix of X. Using a theorem from linear algebra which states that a symmetric matrix is diagonalized by a matrix of its orthonormal eigenvectors (see [112] for details), we can write $C_X = EDE^T$, where D is the diagonal matrix of eigenvalues of C_X and E is the matrix of the corresponding eigenvectors arranged on columns.

Let $r \leq M$ be the rank of C_X. Since we suppose the data can be reconstructed by the orthogonal directions with maximum variance, all data occupies a subspace

of dimension r. Hence, we can complete the remaining $M - r$ vectors, representing the null eigenvalues, in such a way that the orthogonality is preserved. Having null variances, these $M - r$ directions do not influence our analysis.

Selecting $V = E$, we have that $C_Y = E^T(EDE^T)E$. The orthonormality of E also implies $EE^T = I_M$, hence, $C_Y = D$ is a diagonal matrix. In other words, the basis that we are looking for is the orthonormal basis of eigenvectors of $C_X = \frac{1}{N}X^TX$. In addition, the diagonal matrix D contains, on one hand the eigenvalues of C_X and on the other hand, these same values represent the variances of data in the directions of the corresponding eigenvectors.

Note that when data pre-treatment is required, instead of the covariance matrix we actually need to compute the correlation matrix of X, that is,

$$CR_X = \frac{1}{N}\widehat{X}^T\widehat{X}, \tag{6.1.8}$$

where $\widehat{X} = \frac{X-\langle X\rangle}{\sigma_X}$ represents the data after applying data centering and scaling.

Considering the eigenvalues in descending order, we have to decide which are the principal components and which are the components representing noise. The decision about how many principal components are considered can be taken in several ways, one of them being, for example, the signal-to-noise-ratio (SNR), as discussed in [47], which requires the ratio between the signal variance and noise variance to be very large, that is

$$\frac{\sigma^2_{signal}}{\sigma^2_{noise}} \gg 1, \tag{6.1.9}$$

where σ^2_{signal} and σ^2_{noise} represent the sums of the variances specific to the principal components and to the rest of the orthonormal directions, respectively.

Applying PCA for Hamiltonian systems mostly involves distances and, in some cases, velocities analysis. For such systems, the principal components actually determine the volume of space explored by the system particles. Jackson [65] discusses PCA method in detail and presents several PCA applications, such as simplifications and inferential techniques, missing data recovery, or data quality improvement, for example.

Due to its wide range of applicability, PCA has been continuously developed and several nonlinear versions of PCA have been obtained. Kramer [74] proposes a nonlinear principal component analysis (NLPCA) method based on a

feedforward neural network, which identifies and removes correlations among system variables. Compared to PCA, NLPCA uncovers both linear and nonlinear correlations and does not restrict the character of nonlinearities from the data analysed. Scholkopf et al. [107] use the integral operator kernel functions to describe a nonlinear form of PCA. They determine the principal components in high-dimensional spaces related to an input space by some nonlinear maps. They apply this method in image processing and also discuss other kernel techniques.

6.2 The Mahalanobis distance

Multivariate studies often involve distances, the most common measurement used being the Euclidian Distance (ED). Another distance measure is proposed by Mahalanobis [80]. He computes an expression for the distance between two normal (Gauss-Laplacian) statistical populations, described through the P-dimensional frequency distribution

$$df = ce^{-\frac{1}{2\mu}\left[A_{11}(x_1-\mu_1)^2+...+A_{PP}(x_P-\mu_P)^2++A_{12}(x_1-\mu_1)(x_2-\mu_2)+...\right]}dx_1..dx_P, \tag{6.2.1}$$

where c is a constant and $\mu_1, ..., \mu_P$ are the mean values of the population statistics $\{x_1, ..., x_P\}$. Let $\sigma_1, ..., \sigma_P$ be the population standard deviations. Then μ is the determinant of $C = \left(\mu_{ij}\right)_{1\leq i,j\leq P}$, with $\mu_{ij} = \sigma_i\sigma_j\rho_{ij}$. Note that ρ_{ij} are correlation coefficients for which we have $\rho_{ii} = 1$. Finally, A_{ij} are the corresponding minors of the correlation matrix C. In [80], Mahalanobis names C as "the dispersation matrix".

Mahalanobis first proved in [81] that considering two populations a and b with the same dispersations μ_{ij}, but different mean values μ_i^a and μ_i^b, $i = 1, ..., P$, respectively, the distance between a and b measured by a Δ^2-statistic is

$$\Delta^2 = \frac{1}{P}\sum_{i=1}^{P}\frac{\left(\mu_i^a - \mu_i^b\right)^2}{\mu_{ii}}, \tag{6.2.2}$$

which is generalised in [80] for P correlated variables to

$$\Delta^2 = \frac{1}{P}\sum_{1\leq i,j\leq P}^{P}\mu^{ij}\left(\mu_i^a - \mu_i^b\right)\left(\mu_j^a - \mu_j^b\right), \tag{6.2.3}$$

where $\mu^{ij} = \mu_{ij}/\mu$ and $\mu = \det(C)$, as already defined. The distance defined in (6.2.3) is also known as Mahalanobis distance.

In practice, for a sample X of N data observations and n quantities represented as multivariate vectors $\mathbf{x^i} = (x_1^i, ..., x_n^i)$, with mean $\mu = (\mu_1, ..., \mu_n)$ and covariance matrix C_X, the Mahalanobis distance from μ for each observation i is defined as

$$D_X(\mathbf{x^i}) = \sqrt{(\mathbf{x^i} - \mu)\, C_X^{-1}\, (\mathbf{x^i} - \mu)^T}. \tag{6.2.4}$$

Note that if the n quantities analysed are uncorrelated, C_X is diagonal and (6.2.4) becomes the normalised ED, that is

$$D_X(\mathbf{x^i}) = \sqrt{\sum_{j=1}^{n} \frac{\left(x_j^i - \mu_j\right)^2}{\sigma_j^2}}, \tag{6.2.5}$$

where σ_j is the standard deviation for quantity j. Moreover, if $C_X = I_n$, that is the data has unit variance, the Mahalanobis distance equals the Euclidian distance $ED = \sqrt{\sum_{j=1}^{n} \left(x_j^i - \mu_j\right)^2}$.

Also, when using the Euclidian distance to compute the distance from an observation to the dataset center we assume the observations are spherically distributed around this center. On the other hand, PCA analysis of data usually suggests that data distribution is rather ellipsoidal. Hence, if we want, for example, to test if a point belongs to a data sample, we need to take into consideration both the direction and the distance from the center.

In Figure 6.1 the spherical distribution obtained using the Euclidian distance from the center O suggests that points A and B belong to the set, while C and D are not from the dataset. However, if the dataset has an ellipsoidal distribution, the points belonging to the set are B and C, while A and D are outside the set. The ellipsoid best representing the samples probability distribution is estimated using PCA, based on the covariance matrix. The Mahalanobis distance defined in (6.2.4) divides the ED from data center by the width of the ellipsoid in the direction of the point. This gives an accurate prediction whether an observation does or does not belong to a dataset.

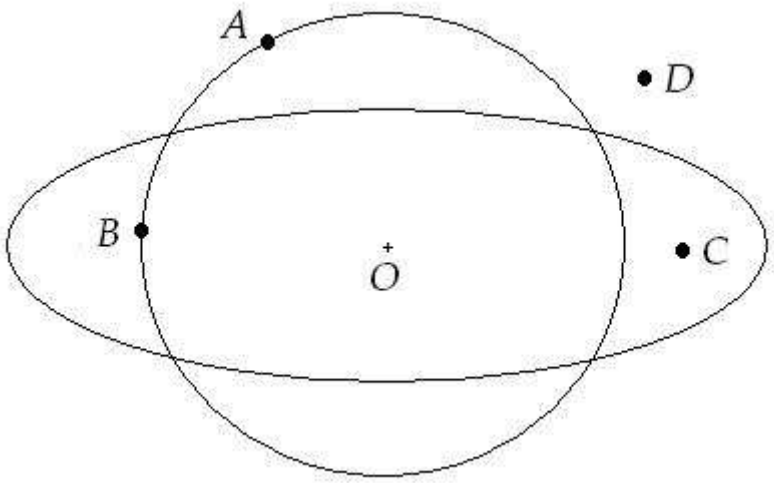

Figure 6.1: Illustration of spherical and ellipsoidal data distribution.

Maesschalck et al. [79] compare the Mahalanobis distance and the Euclidean distance in both the original and principal component (PC) space. They also discuss chemometric methods based on the Mahalanobis distance, such as multivariate calibration, process control and pattern recognition. In fact, the Mahalanobis distance has a wide range of applications in many fields: classification techniques, like cluster analysis, the selection of calibration samples from a large set of measurements, development of linear regression models, by determining outliners (observations that are numerically distant from the analysed data), as well as linear, quadratic and regularised discriminating techniques.

Note that, in the original space, several errors may appear when computing the Mahalanobis distance, the most common one being the covariance matrix singularity due to the so-called data multicollinearity. Analysing the system in the reduced PC space eliminates these errors and we can easily compute the inverse of the covariance matrix, which becomes diagonal.

Finally, the Mahalanobis distance is an example of a Bregman divergence (also known as Bregman distance), which represents a metric not satisfying the triangle inequality nor the symmetry property. Banerjee et al. [9] explain that the Bregman distance generalises the squared Euclidean distance and is strongly connected to exponential families of distributions through a bijection between regular exponential families and regular Bregman divergences.

6.3 Data autocorrelation

The data autocorrelation function represents another analytical technique requiring data centering and unit variance scaling. This function allows one to determine, for example, the presence of a periodic signal that is not visible due to the amplitude of noise. Such goals are achieved by computing the data correlation between values at different time points.

Note that for two arbitrary time steps $t_1 < t_2$ there exists $\tau > 0$ such that $t_2 = t_1 + \tau$ and then we can consider the autocorrelation function to be a lag-function. For a random variable X_t, the autocorrelation function is

$$R_X(\tau) = \frac{\mathbb{E}\left[(X_t - \mu_X)\left(X_{t+\tau} - \mu_X\right)\right]}{\sigma_X^2}, \tag{6.3.1}$$

were μ_X is the mean of X_t and σ_X its standard deviation. It can be easily verified that this function (also called the autocovariance function) is even, that is, $R_X(\tau) = R_X(-\tau)$.

If X is a discrete random variable of length n, we have

$$R_X(k) = \frac{1}{(n-k)\sigma_X} \sum_{i=1}^{n-k} \left(X_i - \mu_X\right)\left(X_{i+k} - \mu_X\right), \forall 0 \leq k < n. \tag{6.3.2}$$

The autocorrelation function has several properties. First of all, if X is a periodic random variable, then R_X is also periodic. Next, if X and Y are two uncorrelated random variables, the autocorrelation function of $X + Y$ is $R_{X+Y} = R_X + R_Y$. Finally, if we rescale X to be the unit variance random vector $\widehat{X} = \frac{X - \mu_X}{\sigma_X}$, then (6.3.1) becomes

$$R_X(\tau) = \mathbb{E}\left[\widehat{X}_t \widehat{X}_{t+\tau}\right], \tag{6.3.3}$$

which, based on Chauchy-Schwarz inequality, implies that $R_X(\tau) \leq R_X(0)$, $\forall \tau \in \mathbb{R}$.

Autocorrelation analysis is important because it reveals how much time we need to simulate a system such that its behaviour is not dependent on the initial conditions. Further analysis of (6.3.3) suggests that $R_X(0) = 1$. As τ increases R_X is expected to decrease uniformly until the data from X becomes independent of the system initial conditions.

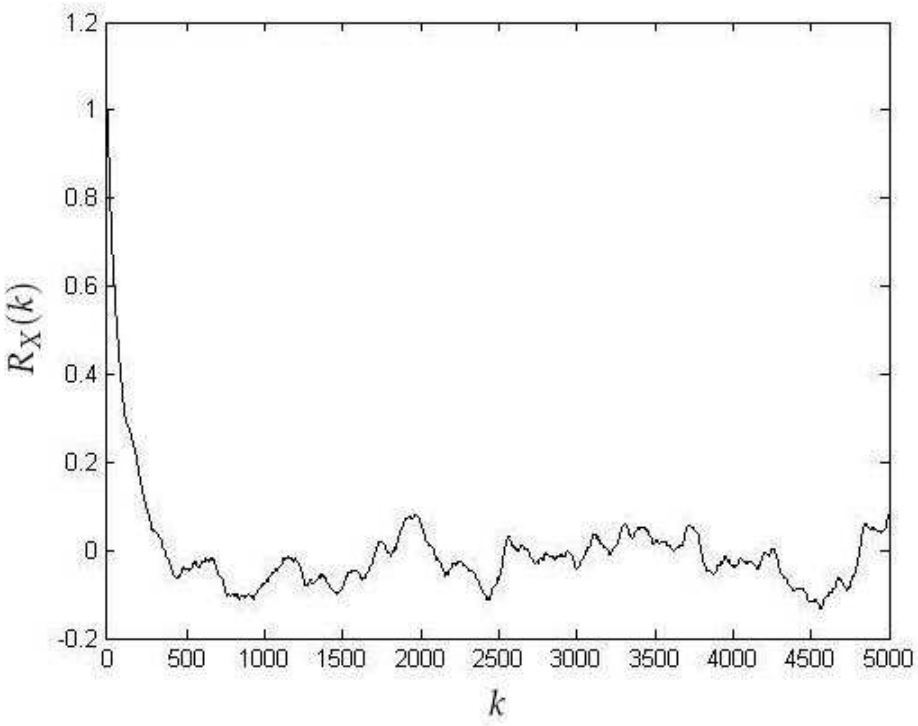

Figure 6.2: Illustration of autocorrelation function plotted against lag value.

Figure 6.2 presents an example of such function, applied for a random vector of size 5000. According to this expression, after 400 iterations, the information about the initial system configuration is lost and the rest of 4600 data observations are independent of the initial system configuration.

6.4 Normal modes

As already discussed, the space explored by a dynamical system can be determined using PCA. Normal modes have similar properties with the ones of the principal components and also allow one to determine the volume of space explored by such systems. Montaldi et al. discuss in [85] the existence of normal modes in symmetric Hamiltonian systems. Due to its dynamical properties, a DNA sequence can be considered a Hamiltonian system and hence it might be possible to describe its dynamics in terms of normal modes – see [124] for more details on the normal mode representation of nonlinear Hamiltonian systems.

We take an example to clarify how Hamiltonian systems can be represented in

terms of normal modes. Consider n particles with masses $m_1, ..., m_n$ that interact through the Hamiltonian

$$H(\mathbf{x}) = \sum_{i=1}^{n} \frac{1}{2} m_i \left(\frac{dx_i}{dt}\right)^2 + \sum_{i=1}^{n} \frac{1}{2} k_i (x_i)^2 + \sum_{1 \le i < j \le n} L_{i,j} x_i x_j, \tag{6.4.1}$$

where $\mathbf{x} = (x_1, ..., x_n)$.

Let $L_{i,j} = L_{j,i}, \forall 1 \le i, j \le n$, $L_{i,i} = 0, \forall i = 1, n$ and consider $L = (L_{i,j})_{1 \le i,j \le n}$ the associated matrix. We also define the diagonal matrices K and M having on the diagonal $(k_1, ..., k_n)$ and $(m_1, ..., m_n)$, respectively. Then, our system becomes

$$M \frac{d^2 \mathbf{x}}{dt^2} = -(K + L)\mathbf{x}. \tag{6.4.2}$$

Considering $\mathbf{u} = (u_1, ..., u_n)^T$, the solution of our system is a sum of terms having the form $\mathbf{x_1}(t) = \mathbf{u} \cos(\omega t)$ or $\mathbf{x_2}(t) = \mathbf{u} \sin(\omega t)$, where $\mathbf{u}$ represents the normal mode and ω its specific frequency. Having n particles in our system, we also have n normal modes and n associated frequencies. Since $\frac{d^2 \mathbf{x_1}}{dt^2}(t) = -\omega^2 \mathbf{u} \cos(\omega t)$ and $\frac{d^2 \mathbf{x_2}}{dt^2}(t) = -\omega^2 \mathbf{u} \sin(\omega t)$, (6.4.2) becomes, for both $\mathbf{x_1}$ and $\mathbf{x_2}$,

$$\omega^2 M \mathbf{u} = (K + L)\mathbf{u}, \tag{6.4.3}$$

which is equivalent to

$$\omega^2 \mathbf{u} = M^{-1}(K + L)\mathbf{u} \tag{6.4.4}$$

and hence, ω^2 is an eigenvalue of $M^{-1}(K + L)$, while $\mathbf{u}$ is the corresponding eigenvector. In this way, we determine the frequencies ω_j and the normal modes $\mathbf{u_j}$, $(j = 1, ..., n)$, as well as the general solution $\mathbf{x}(t)$ of the system given by (6.4.2), which has the form

$$\mathbf{x}(t) = \sum_{j=1}^{n} \left[C_j^1 \mathbf{u_j} \cos(\omega_j t) + C_j^2 \mathbf{u_j} \sin(\omega_j t) \right]. \tag{6.4.5}$$

Here C_j^1 and C_j^2 are the modes amplitudes that can determined by imposing the condition that each mode has the same energy, for example, unity ($H_j^{cos} = H(C_j^1 \mathbf{u_j} \cos(\omega_j t)) = 1$ and $H_j^{sin} = H(C_j^1 \mathbf{u_j} \sin(\omega_j t)) = 1$, $\forall j = 1, ..., n$). Let U be the matrix containing as columns the normal mode vectors $\mathbf{u_j}$, $j = 1, ..., n$.

Considering first $X(t) = C_j^1 \mathbf{u_j} \cos(\omega_j t)$, for some fixed j, we have that

$$H_j^{cos} = \sum_{i=1}^{n} \frac{1}{2} m_i (\omega_j)^2 (C_j^1)^2 (U_{i,j})^2 \sin^2(\omega_j t) + \sum_{i=1}^{n} \frac{1}{2} k_i (C_j^1)^2 (U_{i,j})^2 \cos^2(\omega_j t) + \sum_{1 \le i1 < i2 \le n} L_{i1,i2} (C_j^1)^2 U_{i1,j} U_{i2,j} \cos^2(\omega_j t) \tag{6.4.6}$$

(6.4.7)

equivalent to

$$H_j^{cos} = \frac{(C_j^1)^2}{2} \sin^2(\omega_j t) \sum_{i=1}^{n} m_i (\omega_j)^2 (U_{i,j})^2 + \frac{(C_j^1)^2}{2} \cos^2(\omega_j t) \left[\sum_{i=1}^{n} k_i (U_{i,j})^2 + 2 \sum_{1 \le i1 < i2 \le n} L_{i1,i2} U_{i1,j} U_{i2,j} \right], \tag{6.4.8}$$

which gives

$$H_j^{cos} = \frac{(C_j^1)^2}{2} \sin^2(\omega_j t) \sum_{i=1}^{n} m_i (\omega_j)^2 (U_{i,j})^2 + \frac{(C_j^1)^2}{2} \cos^2(\omega_j t) \left[\sum_{i=1}^{n} U_{i,j} \left[k_i U_{i,j} + \sum_{1 \le i1 \le n, i1 \ne i} L_{i,i1} U_{i1,j} \right] \right], \tag{6.4.9}$$

But (6.4.3) implies

$$k_i U_{i,j} + \sum_{1 \le i1 \le n, i1 \ne i} L_{i,i1} U_{i1,j} = m_i (\omega_j)^2 U_{i,j}, \tag{6.4.10}$$

hence

$$H_j^{cos} = \frac{(C_j^1)^2}{2} \left[\left(\sin^2(\omega_j t) + \cos^2(\omega_j t) \right) \sum_{i=1}^{n} m_i (\omega_j)^2 (U_{i,j})^2 \right] = \frac{(C_j^1)^2}{2} \sum_{i=1}^{n} m_i (\omega_j)^2 (U_{i,j})^2 \tag{6.4.11}$$

and having unit Hamiltonian in each mode implies

$$C_j^1 = \pm \sqrt{\frac{2}{(\omega_j)^2 \sum_{i=1}^{n} m_i (U_{i,j})^2}} \tag{6.4.12}$$

A similar computation shows that $C_j^2 = \pm \left| C_j^1 \right|$, supposing that $H_j^{sin} = 1$. Note that in this case $C_j^1 = \pm C_j^2$, $\forall j = 1, ..., n$.

6.4.1 Normal modes data variances

Let $N > 1$ be an integer and t_k, $1 \leq k \leq N$, an increasing sequence of times. We define the $N \times n$ matrices X and Y (where n is the number of particles) as the data representation in terms of the canonical basis and normal mode vectors, respectively. Then N is actually the number of observations equally spaced between t_1 and t_N and the j^{th} column of Y is

$$Y_j = \begin{pmatrix} C_j^1 \cos(\omega_j t_1) + C_j^2 \sin(\omega_j t_1) \\ \dots \\ C_j^1 \cos(\omega_j t_N) + C_j^2 \sin(\omega_j t_N) \end{pmatrix}. \tag{6.4.13}$$

Note that we have $X = YU^T$ and hence $(U^T)^{-1}$ is the transformation matrix that maps X into Y. In addition, observe that *sine* and *cosine* are uncorrelated functions, hence if the sample of size N is large enough, we have

$$\sum_{k=1}^{N} cos(\omega_j t_k) sin(\omega_j t_k) = 0, \ \forall j = 1, ..., n, \tag{6.4.14}$$

as well as

$$\sum_{k=1}^{N} cos(\omega_j t_k) = \sum_{k=1}^{N} sin(\omega_j t_k) = 0, \ \forall j = 1, ..., n. \tag{6.4.15}$$

Moreover, given $j_1 \neq j_2$ and under the assumption that ω_{j_1} is not a multiple of ω_{j_2} or viceversa, we have

$$\sum_{k=1}^{N} cos(\omega_{j_1} t_k) cos(\omega_{j_2} t_k) = 0, \tag{6.4.16}$$

$$\sum_{k=1}^{N} sin(\omega_{j_1} t_k) sin(\omega_{j_2} t_k) = 0, \tag{6.4.17}$$

and

$$\sum_{k=1}^{N} cos(\omega_{j_1} t_k) sin(\omega_{j_2} t_k) = 0. \tag{6.4.18}$$

Using (6.4.14)-(6.4.18) we obtain the covariance matrix of Y as the diagonal $n \times n$ matrix

$$C_Y = \begin{pmatrix} S_1 & 0 & \dots & 0 \\ 0 & S_2 & \dots & 0 \\ \dots & & & \\ 0 & 0 & \dots & S_n \end{pmatrix}, \tag{6.4.19}$$

where

$$S_j = \frac{1}{N}\sum_{k=1}^{N}\left[\left(C_j^1\right)^2\cos^2(\omega_j t_k) + \left(C_j^2\right)^2\sin^2(\omega_j t_k)\right],\ \forall j = 1, ..., n, \tag{6.4.20}$$

is the data variance in the direction of U_j.

Rewriting S_j we obtain

$$\begin{aligned} S_j &= \frac{1}{N}\sum_{k=1}^{N}\left[\left(C_j^1\right)^2\cos^2(\omega_j t_k) + \left(C_j^2\right)^2(1-\cos^2(\omega_j t_k))\right] \\ &= \left(C_j^2\right)^2 + \frac{\left(C_j^1\right)^2 - \left(C_j^2\right)^2}{N}\sum_{k=1}^{N}\cos^2(\omega_j t_k). \end{aligned} \tag{6.4.21}$$

Since

$$\frac{1}{N}\sum_{k=1}^{N}\cos^2(\omega_j t_k) + \frac{1}{N}\sum_{k=1}^{N}\sin^2(\omega_j t_k) = 1 \tag{6.4.22}$$

and based on (6.4.14) and (6.4.15) we have

$$\frac{1}{N}\sum_{k=1}^{N}\cos^2(\omega_j t_k) - \frac{1}{N}\sum_{k=1}^{N}\sin^2(\omega_j t_k) = \frac{1}{N}\sum_{k=1}^{N}\cos(2\omega_j t_k) = 0, \tag{6.4.23}$$

we obtain

$$\frac{1}{N}\sum_{k=1}^{N}\cos^2(\omega_j t_k) = \frac{1}{N}\sum_{k=1}^{N}\sin^2(\omega_j t_k) = \frac{1}{2}. \tag{6.4.24}$$

Finally, we have

$$S_j = \frac{\left(C_j^1\right)^2 + \left(C_j^2\right)^2}{2}. \tag{6.4.25}$$

Returning to our example, for which C_j^1 and C_j^2 are proportional to $\frac{1}{\omega_j}$ (see (6.4.12)), we have that S_j is proportional to $\frac{1}{\omega_j^2}$. Since $\omega_1 < \omega_2 < ... < \omega_n$, we also have $S_1 > S_2 > ... > S_n$.

Thus, the normal mode data representation has two important properties: first, the covariance matrix is diagonal and, intuitively, we may say the modes having high data variance are specific to low frequencies. In addition, if $\omega_j \gg 1$,

then $S_j \ll 1$ and the contribution of the corresponding normal mode to data can be considered part of the noisy data component.

In conclusion, the normal mode representation is an alternative to PCA, but there is a significant difference between the two methods: PCA determines the directions with the highest variance, normal modes represent a frequency-based analysis, for which the number of non-noise modes is, in general, larger than the number of principal components. Whilst it is tempting to treat the principal components as normal modes, with a large variance component corresponding to a low frequency mode, this is not necessarily the case, given that the principal components are orthogonal, but the normal modes are not.

6.5 Fourier transform

In the previous section, we presented a method to obtain the normal modes given the Hamiltonian of a linear system, but in practice we have to determine the normal modes given a set of data and a priori we do not know if the data is generated by a linear system. Since the principal components are orthogonal, while the normal modes are not necessarily characterised by orthogonal vectors, the method of PCA does not offer a direct algorithm to determine the normal modes.

On the other hand, the Fourier transform [17] provides an algorithm to determine, given a set of data, not only the specific frequencies, but also the corresponding normal modes. However, our purpose is not to present an algorithm to obtain the Fourier Transform for a function or data vector and we only use the Fourier transform to determine the normal modes for a given dataset. Such predefined algorithms have already been developed – see [36], for example – and are available as part of several mathematical software packages, such as MATLAB.

The Fourier transform of a function, also known as the frequency domain representation of the original function, describes the frequencies present in the original function. The input function can be reconstructed using the inverse Fourier transform. In other words, if $f : \mathbb{R} \to \mathbb{C}$ is an integrable function, then

$F : \mathbb{R} \to \mathbb{C}$, defined as

$$F(y) = \int_{-\infty}^{\infty} f(x)e^{-2\pi ixy}dx, \ \forall y \in \mathbb{R}, \tag{6.5.1}$$

is its Fourier Transform. Note that if x represents a time coordinate, then y is a frequency, which suggests that the Fourier transform is useful for determining the normal mode frequencies. Applying the inverse transform, we obtain that

$$f(x) = \frac{1}{2\pi}\int_{-\infty}^{\infty} F(y)e^{2\pi ixy}dy, \ \forall x \in \mathbb{R} \tag{6.5.2}$$

6.5.1 Discrete Fourier transform

A data sample can be viewed as a discrete representation of an event and thus, we need the Discrete Fourier Transform (DFT), which requires a discrete input function. Let $\mathbf{x} = (x_1, ..., x_N)$ be a vector of N complex numbers. Considering $\Omega_k = \frac{2(k-1)\pi}{N}$, $\mathbf{x}$ is transformed into $\mathbf{y} = (y_1, ..., y_N)$ using the DFT formula

$$y_k = \sum_{n=0}^{N-1} x_{n+1}e^{-\Omega_k ni}, \forall k = 1, ..., N. \tag{6.5.3}$$

Note that $e^{-\Omega_k}$ is one of the N^{th} roots of unity. The inverse DFT is given by

$$x_{n+1} = \frac{1}{2N\pi}\sum_{k=1}^{N} y_k e^{\Omega_k ni}, \forall n = 0, ..., N-1. \tag{6.5.4}$$

MATLAB provides the *fft* algorithm to compute the DFT for a data vector. The algorithm is based on the Fast Fourier Transform (FFT) algorithm [18]. There are several FFT algorithms, but the most wide used is Cooley-Tukey algorithm, which reduces the computational complexity by splitting a DFT of a vector of size N into smaller DFTs of sizes N_1, N_2,...,N_k such that $N = N_1 \times N_2 \times ... \times N_k$. This implies that the algorithm gives best results, when N is a composite number. Moreover, the most well-known implementation of Cooley-Tukey FFT algorithm, splits at each step the transform into two DFTs of size $N/2$, which suggests that $N = 2^k$, for some $k \in \mathbb{N}$, is the optimal data sample size, for accurate results.

To complete the spectrum analysis, we note that (6.5.4) determines the DFT contribution for each point in $\mathbf{x}$. For accurate results, it is recommended that we use

an average transform of several DFT, obtained from different data samples, to determine the magnitude for each frequency. It is also recommended to center the data before computing its DFT, by subtracting the mean of the sequence from all elements of the sequence.

6.5.2 Normal modes and the Fourier transform

Let the $N \times n$ matrix X be the centered data representation in the canonical basis for a set of observations specific to a n particles system, for the time interval between t_1 and t_N. Supposing the data can be represented in terms of normal modes, we need to determine the specific frequencies and the associated vectors. Note that we expect $N \ll n$

Each row of X being an observation $(x_1(t_k), ..., x_n(t_k))$, $1 \leq k \leq N$, the sample can be viewed as a function of time t_k, which implies that the Fourier transform can offer information about the desired frequencies. The number of observations N is assumed to be a power of 2, so that we apply the FFT algorithm to each column of X to determine the DFT of the full dataset X.

Normal modes frequencies

Note that if the data is time dependent, that is, $\mathbf{x} = (x_{t_1}, ..., x_{t_N})$, and we write its Fourier transform as $\mathbf{y} = (y_{\omega_1}, ..., y_{\omega_N})$, then (6.5.3) becomes

$$y_{\omega_k} = \sum_{n=1}^{N} x_{t_n} e^{-\omega_k (t_n - t_1) i}, \ \forall k = 1, ..., N, \tag{6.5.5}$$

where $\omega_k = \frac{2(k-1)\pi}{t_N - t_1}$. The inverse DTF becomes

$$x_{t_n} = \frac{1}{N} \sum_{k=1}^{N} y_{\omega_k} e^{\omega_k (t_n - t_1) i}, \ \forall n = 1, ..., N. \tag{6.5.6}$$

Suppose Y_j is the DFT of column X_j of X and let Y be the matrix having as columns these DFTs. We select based on the DFTs n frequencies ($n < N$) in the system, having the highest magnitutude. Note that these n frequencies are not necessarily represented in any column of X. Hence, for accurate results, we sum

for each frequency ω_k the contributions of all columns to get the corresponding magnitude, that is, $\sum_{j=1}^{n} Y_{k,j}$.

Let $\omega_{k_1}, \omega_{k_2}, ..., \omega_{k_n}$ be the n unit roots of order N that we are looking for. Then, the actual frequencies of our system are obtained by the formula

$$\omega_k = \frac{2(k-1)\pi}{t_N - t_1},\ k = k_1, ..., k_n. \tag{6.5.7}$$

Note that the DFT frequencies are equally spaced, while normal modes frequencies from (6.5.7) are not necessarily equally spaced. Since we choose only those n (with the largest y_{ω_k}) out of the N DFT frequencies, a large number of observations ensures a good approximation of the normal modes frequencies.

Normal mode vectors

Let $1 \leq j \leq n$ and $k \in \{k_1, ..., k_n\}$. We want to determine the normal mode vector $\mathbf{u_k}$ (with frequency ω_k) based on the DFTs for the columns of X. Based on (6.5.6), the k^{th} entry of Y_j, to which we refer by $Y_{k,j}$, represents the normal mode $\mathbf{u_k}$ scaled contribution to the sample data. Moreover, (6.5.5) can be written as

$$Y_{j,k} = \sum_{n=1}^{N} X_{n,j} \left[\cos(\omega_k(t_n - t_1)) - i \sin(\omega_k(t_n - t_1))\right], \tag{6.5.8}$$

implying that the real part of $Y_{k,j}$ is specific to the *cosine* contribution, while the imaginary part represents the *sine* contribution.

Recall the normal modes data representation from (6.4.5), which in our case is $X(t_n) = \sum_{i=1}^{n} \left[C_i^1 \mathbf{u_i} \cos(\omega_i t_n) + C_i^2 \mathbf{u_i} \sin(\omega_i t_n)\right]^T$. Using (6.4.14)-(6.4.18) we obtain in terms of the matrix U of normal mode vectors

$$\begin{aligned} Y_{j,k} &= \sum_{n=1}^{N} \left[C_k^1 U_{j,k} \cos(\omega_k t_1) \cos^2(\omega_k t_n) + C_k^2 U_{j,k} \sin(\omega_k t_1) \sin^2(\omega_k t_n)\right] \\ &- i \sum_{n=1}^{N} \left[-C_k^1 U_{j,k} \sin(\omega_k t_1) \cos^2(\omega_k t_n) + C_k^2 U_{j,k} \cos(\omega_k t_1) \sin^2(\omega_k t_n)\right] \end{aligned}$$

and based on (6.4.24), we conclude that

$$\begin{aligned} Y_{j,k} &= \frac{N}{2} U_{j,k} \left[C_k^1 \cos(\omega_k t_1) + C_k^2 \sin(\omega_k t_1)\right] \\ &\quad -i \frac{N}{2} U_{j,k} \left[C_k^2 \cos(\omega_k t_1) - C_k^1 \sin(\omega_k t_1)\right]. \end{aligned} \tag{6.5.9}$$

Let $\mathbf{v_1} = (Re(Y_{1,k}), Re(Y_{2,k}), ..., Re(Y_{n,k}))^T$, with $Re(y)$ representing the real part of y and $\mathbf{v_2} = (Im(Y_{1,k}), Im(Y_{2,k}), ..., Im(Y_{n,k}))^T$, where $Im(y)$ is the imaginary part of y. Given that in (6.5.9) the coefficients $C_k^1 \cos(\omega_k t_1) + C_k^2 \sin(\omega_k t_1)$ and $C_k^2 \cos(\omega_k t_1) - C_k^1 \sin(\omega_k t_1)$ are independent of j, the unit vectors corresponding to $\mathbf{v_1}$ and $\mathbf{v_2}$ equal $\pm\mathbf{u_k}$. Hence, the Fourier transform also allows us to compute the normal modes vectors.

Note that when $t_1 > 0$ is known, (6.5.9) suggests that $C_k^1 = \frac{Re(Y_{j,k})}{NU_{j,k}} \cos(\omega_k t_1) + \frac{Im(Y_{j,k})}{NU_{j,k}} \sin(\omega_k t_1)$ and $C_k^2 = \frac{Re(Y_{j,k})}{NU_{j,k}} \sin(\omega_k t_1) - \frac{Im(Y_{j,k})}{NU_{j,k}} \cos(\omega_k t_1)$. This means we have all the information necessary to simulate the system starting from $t = 0$.

However, if t_1 is unknown, it is impossible to determine C_k^1 and C_k^2. In this case we define

$$\begin{aligned} \alpha_k(t) &= \left[C_k^1 \cos(\omega_k t_1) + C_k^2 \sin(\omega_k t_1)\right] \cos(\omega_k t) \\ &\quad + \left[C_k^2 \cos(\omega_k t_1) - C_k^1 \sin(\omega_k t_1)\right] \sin(\omega_k t), \end{aligned} \tag{6.5.10}$$

for some $t \geq 0$, which can be rewritten as

$$\alpha_k(t) = C_k^1 \cos(\omega_k (t + t_1)) + C_k^2 \sin(\omega_k (t + t_1)). \tag{6.5.11}$$

Note that the coefficients from (6.5.10) are determined through (6.5.9). Furthermore, we have

$$\mathbf{x}(t_1 + t) = \sum_{i=1}^{n} \alpha_i(t) \mathbf{u_i}, \tag{6.5.12}$$

which allows us to predict the system behaviour without knowing the normal modes amplitudes C_k^1 and C_k^2. Thus, we can reconstruct the initial data using the normal modes representation. We can also simulate new data for $t > t_N$, as well as new data not contained in X, for any t such that $t_1 < t < t_N$.

All in all, the Fourier transform allows one to determine the normal modes, the specific frequencies and information about the amplitudes and phase angle of each normal mode. This type of analysis is useful when an event under study repeats with a certain frequency or is the result of several repeated system events, with different frequencies.

6.6 Numerical example

In what follows, we present an example that shows how to apply each of the methods presented in this chapter. Starting from the Hamiltonian (6.4.1), we determine the normal modes and the specific frequencies for a four-particle system. Next, we simulate the system and compute the principal components and the autocorrelation function to determine the directions with highest variance in the system and the extent to which the system remembers its initial conditions, respectively. Finally, we try to reconstruct the normal modes and frequencies, using FFT algorithm and DFT methodology.

6.6.1 Normal modes representation

For $n = 4$, $m_1 = 0.25$, $m_2 = 0.65$, $m_3 = 0.48$, $m_4 = 0.53$, $k_1 = 17.8$, $k_2 = 3.3$, $k_3 = 14.9$, $k_4 = 4.6$, $M = diag(m_1, m_2, m_3, m_4)$, $K = diag(k_1, k_2, k_3, k_4)$ and

$$L = \begin{pmatrix} 0 & 0.25 & 0.95 & 0.78 \\ 0.25 & 0 & 0.65 & 0.85 \\ 0.95 & 0.65 & 0 & 0.5 \\ 0.78 & 0.85 & 0.5 & 0 \end{pmatrix}, \tag{6.6.1}$$

we obtain by solving (6.4.3) the frequencies $\omega_1 = 2.1316$, $\omega_2 = 3.0069$, $\omega_3 = 5.5624$, $\omega_4 = 8.4541$ and the following normal modes

$$U = \begin{pmatrix} 0.0350 & 0.0309 & 0.0292 & 0.0484 \\ 0.3983 & 0.5284 & -0.9155 & -0.3399 \\ 0.0120 & 0.0038 & 0.0626 & 0.0594 \\ -0.9165 & -0.8484 & -0.3963 & -0.9374 \end{pmatrix}. \tag{6.6.2}$$

The coefficients for each mode have absolute values equal to

$$C = (0.8957,\ 0.6267,\ 0.3203,\ 0.2270). \tag{6.6.3}$$

We simulate the system starting from $t = 0$, with $\Delta t = 10^{-3}$, considering $C_i^1 = C_i^2 = C_i > 0$, $\forall i = 1, ..., n$, and using (6.4.5). The four variables of our system are presented in Figure 6.3.

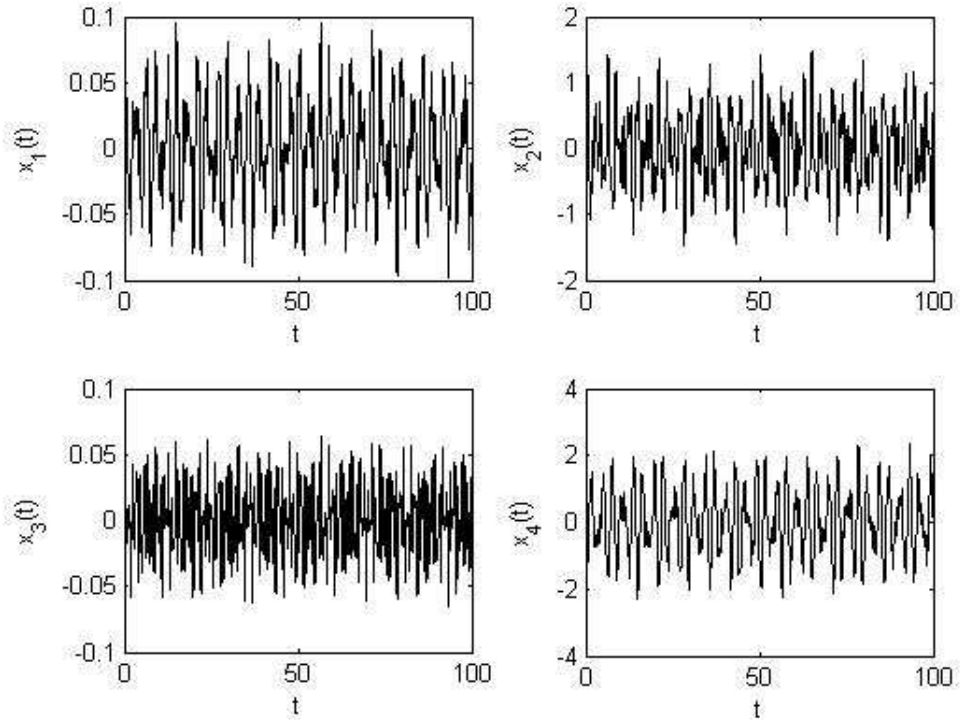

Figure 6.3: Illustration of data obtained using normal modes representation.

Note that in the case of normal modes the data is centered, since for $T \gg 1$ we have

$$\int_{t=0}^{T} \cos(\omega t) = \int_{t=0}^{T} \sin(\omega t) = 0, \ \forall \omega > 0, \tag{6.6.4}$$

Hence, in what follows we only need the standard deviation of the data to transform it into unit variance. However, in general before scaling the data, one should subtract the data's mean value.

6.6.2 PCA analysis

Applying PCA without any data pre-treatment (rescaling), the principal components and eigenvalues are

$$V = \begin{pmatrix} -0.0344 & 0.0308 & 0.3711 & -0.9275 \\ -0.4246 & -0.9037 & 0.0550 & 0.0078 \\ -0.0108 & 0.0646 & 0.9256 & 0.3728 \\ 0.9046 & -0.4222 & 0.0509 & -0.0272 \end{pmatrix} \tag{6.6.5}$$

and $\lambda = (1.2163,\ 0.1346,\ 0.0000,\ 0.0000)$, respectively.

Recall that the normal modes are not necessarily orthogonal, while the principal components form an orthogonal basis. Moreover, the first normal mode from (6.6.2) and the first principal component from (6.6.5) represent similar, but not identical vectors (but pointing in opposite directions). Hence, scaling the data might be helpful in obtaining a nonorthogonal set of principal components: after applying PCA to the scaled data, we can rescale the principal components based on the standard deviations previously determined, and so obtain nonorthogonal principal components.

In other words, if $\sigma = (\sigma_1, ..., \sigma_M)$ are the standard deviations of the columns of X and $\widehat{X}$ is the data scaled as presented in Section 6.1.1, then applying PCA we obtain the correlation matrix $\widehat{C}$ (rather than covariance matrix) and the matrix $\widehat{V}$ of principal components. If each column of $\widehat{V}$ represents a principal component and we multiply each row by the corresponding standard deviation from σ, we end with a non-orthogonal set of principal components. Observe that if the data is not centered, data scaling supposes the subtraction of means of columns of X, but rescaling does not require the addition of these means to the corresponding rows of $\widehat{V}$. Indeed, if we expect to obtain the normal modes, we need to analyse centered data. The mean for each column is then added to the expression from (6.4.5), to obtain the final system configuration.

For our numerical example we have $\sigma = (0.0396,\ 0.5738,\ 0.0265,\ 1.0097)$ and applying PCA we obtain the eigenvectors matrix

$$\widehat{V} = \begin{pmatrix} -0.5919 & 0.0671 & -0.1946 & -0.7793 \\ -0.3992 & -0.6869 & 0.5999 & 0.0943 \\ -0.3737 & 0.7212 & 0.5441 & 0.2100 \\ 0.5922 & 0.0591 & 0.5533 & -0.5828 \end{pmatrix}, \tag{6.6.6}$$

with corresponding eigenvalues

$$\lambda = (2.8393,\ 1.1597,\ 0.0009,\ 0.0000)\,. \tag{6.6.7}$$

As can be observed, we have the same number of principal components as for V from (6.6.5), more precisely, two PCs. Rescaling $\widehat{V}$ we obtain the nonorthogonal

principal components

$$\widetilde{V} = \begin{pmatrix} -0.0366 & 0.0068 & -0.0117 & -0.0519 \\ -0.3576 & -0.9880 & 0.5256 & 0.0919 \\ -0.0155 & 0.0482 & 0.0221 & 0.0094 \\ 0.9330 & 0.1468 & 0.8504 & -0.9944 \end{pmatrix}. \tag{6.6.8}$$

Compared to (6.6.5), $\widetilde{V}$ contains a similar first principal component, but differs in the other three components determined. Comparing with (6.6.2) we reach the conclusion that this method does not determine the normal modes either. Moreover, in both cases (with and without scaling), the first principal component is close to the first normal mode, but not equal to it, while neither of the second PC are like the second normal mode. The differences in principal components are given by the method used (with or without data scaling).

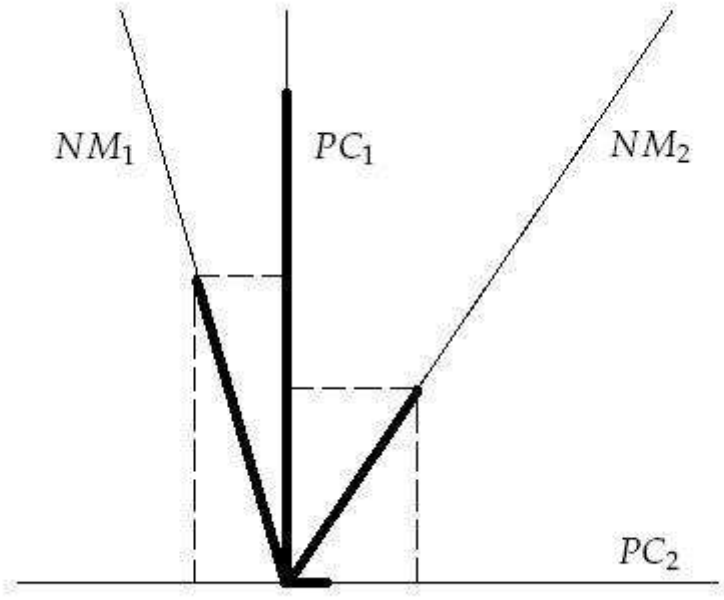

Figure 6.4: Illustration of normal modes, principal components and their variances.

The deviation from the first normal mode is explained in Figure 6.4. This simple example, illustrating the first two normal modes and principal components, suggests that the direction with the highest data variance is not necessarily the direction of the first normal mode. Indeed, the first principal component takes

into consideration the contribution of all normal modes and the orthogonality of the principal components ensures this data does not overlap, whilst the variance in the direction of one normal mode may involve other normal modes too.

6.6.3 Input data influence on principal components

It is interesting to see at this point how the principal components change if we do not take into consideration one of the four particles, for example. Applying PCA just for the first three particles, we obtain the following principal components

$$V = \begin{pmatrix} -0.0426 & 0.7637 & 0.6442 \\ -0.9991 & -0.0281 & -0.0328 \\ 0.0069 & 0.6450 & -0.7642 \end{pmatrix}, \tag{6.6.9}$$

with eigenvalues $\lambda = (0.3295, 0.0017, 0.0000)$. We observe important variations in data variance and principal components compared to (6.6.5). Scaling the data to unit variance, we obtain

$$\widetilde{V} = \begin{pmatrix} -0.1135 & 0.0016 & -0.0930 \\ -0.9921 & 0.9991 & 0.9944 \\ -0.0532 & -0.0415 & 0.0496 \end{pmatrix}, \tag{6.6.10}$$

which agrees in first principal component with (6.6.8), but the other two components point some different directions due to the orthogonality property of principal components. The variances in each directions are also different, given that $\lambda = (1.8495, 1.1500, 0.0005)$. This difference is due to the fourth particle whose variance contribution in the three directions is not taken into consideration in this case.

This suggests that eliminating some particles from the system analysis changes the principal components directions. However, considering the remaining particles to be equally important is more appropriate, given that, in this case, the first principal component points in the same direction as in the case when all system particles are considered. Hence, it makes sense to scale the data and have equal variances for all particles.

Finally, recall that for our SDE model we needed data each 2 fs to fit correctly our parameters. However, for this simple system of four particles, changing the time step from 10^{-3} to 10^{-1}, for example, minor deviations from principal components and variations in data variances were observed.

6.6.4 Trajectory and velocity in PCA

Another approach takes into account both, trajectory and velocity components. In this case, rescaling the data to unit variance is necessary, since the dimensions of the two quantities (displacement and velocity) are different.

Since our data has the form $\mathbf{x}(t) = \sum_{j=1}^{n} \left[C_j^1 \mathbf{u_j} \cos(\omega_j t) + C_j^2 \mathbf{u_j} \sin(\omega_j t) \right]^T$, then

$$\frac{d\mathbf{x}}{dt}(t) = \sum_{j=1}^{n} \left[-\omega_j C_j^1 \mathbf{u_j} \sin(\omega_j t) + \omega_j C_j^2 \mathbf{u_j} \cos(\omega_j t) \right]^T. \tag{6.6.11}$$

Then we can define the matrix Y, containing on the first n columns the trajectory data and on the last n columns the corresponding velocities. For each $1 \leq j \leq n$, we also define the vectors

$$\mathbf{u_j^1} = \begin{pmatrix} C_j^1 \mathbf{u_j} \\ \omega_j C_j^2 \mathbf{u_j} \end{pmatrix} \tag{6.6.12}$$

and

$$\mathbf{u_j^2} = \begin{pmatrix} C_j^2 \mathbf{u_j} \\ -\omega_j C_j^1 \mathbf{u_j} \end{pmatrix}. \tag{6.6.13}$$

Then, the rows of Y have the form

$$\mathbf{y}(t) = \sum_{j=1}^{n} \left[\mathbf{u_j^2} \cos(\omega_j t) + \mathbf{u_j^2} \sin(\omega_j t) \right]^T. \tag{6.6.14}$$

Let $\widehat{U}$ be the matrix containing $\mathbf{u_j^1}$ and $\mathbf{u_j^2}$ on columns. The new standard deviations for our four-particle system (needed for data scaling to unit variance) are

$$\sigma = (0.0396, 0.5738, 0.0265, 1.0097, 0.1385, 2.1584, 0.1612, 3.0587), \tag{6.6.15}$$

while the matrix of normal mode vectors becomes

$$\widehat{U} = \begin{pmatrix} 0.03 & 0.03 & 0.02 & 0.02 & 0.01 & 0.01 & 0.01 & 0.01 \\ 0.36 & 0.36 & 0.33 & 0.33 & -0.29 & -0.29 & -0.08 & -0.08 \\ 0.01 & 0.01 & 0.01 & 0.01 & 0.02 & 0.02 & 0.01 & 0.01 \\ -0.82 & -0.82 & -0.53 & -0.53 & -0.13 & -0.13 & -0.21 & -0.21 \\ 0.07 & -0.07 & 0.06 & -0.06 & 0.05 & -0.05 & 0.09 & -0.09 \\ 0.76 & -0.76 & 0.99 & -0.99 & -1.63 & 1.63 & -0.65 & 0.65 \\ 0.02 & -0.02 & 0.01 & -0.01 & 0.11 & -0.11 & 0.11 & -0.11 \\ -1.75 & 1.75 & -1.60 & 1.60 & -0.71 & 0.71 & -1.60 & 1.60 \end{pmatrix}.$$

As can be observed, the velocities have higher variances than the displacements, which is explained by the presence of an extra ω_j in the velocity components of $\mathbf{u_j^1}$ and $\mathbf{u_j^2}$. Applying PCA to the new set of data, we obtain the following eigenvalues

$$\lambda = (2.8394, 2.7476, 1.2504, 1.1597, 0.0020, 0.0009, 0.0000, 0.0000). \tag{6.6.16}$$

Observe that these values come in pairs, one of them being identical to the one specific to the trajectory component – see (6.6.7). Our results show that the extra eigenvalue in each pair comes from the velocity component. In fact, the principal components also split into trajectory and velocity vectors, respectively, as follows

$$V = \begin{pmatrix} -0.59 & 0.01 & -0.00 & 0.07 & -0.00 & 0.19 & 0.78 & 0.00 \\ -0.40 & 0.01 & -0.00 & -0.69 & 0.00 & -0.60 & -0.09 & -0.00 \\ -0.37 & 0.01 & 0.00 & 0.72 & 0.00 & -0.54 & -0.21 & -0.00 \\ 0.59 & -0.01 & 0.00 & 0.06 & 0.00 & -0.55 & 0.58 & 0.00 \\ 0.01 & 0.58 & -0.24 & 0.00 & -0.02 & 0.00 & 0.00 & -0.78 \\ -0.01 & -0.23 & -0.83 & 0.00 & -0.50 & -0.00 & 0.00 & 0.09 \\ 0.01 & 0.57 & 0.30 & -0.00 & -0.68 & -0.00 & -0.00 & 0.35 \\ -0.01 & -0.53 & 0.41 & -0.00 & -0.53 & -0.00 & 0.00 & -0.51 \end{pmatrix},$$

where 0.00 represents a positive value close to zero (and -0.0 represents a negative value close to zero). After rescaling and normalizing the column vectors from V, in order to return the data to the initial scales and dimensions, we ob-

tain

$$\widehat{V} = \begin{pmatrix} -0.02 & 0.00 & 0.00 & 0.01 & 0.00 & 0.01 & 0.03 & 0.00 \\ -0.23 & 0.01 & 0.00 & -0.39 & 0.00 & -0.34 & -0.05 & 0.00 \\ -0.01 & 0.00 & 0.00 & 0.02 & 0.00 & -0.01 & -0.01 & 0.00 \\ 0.60 & -0.01 & -0.00 & 0.06 & -0.01 & -0.56 & 0.59 & -0.01 \\ 0.01 & 0.08 & -0.03 & -0.00 & -0.01 & -0.00 & -0.00 & -0.11 \\ -0.01 & -0.50 & -1.78 & 0.00 & -1.09 & -0.00 & -0.00 & 0.20 \\ 0.00 & 0.09 & 0.05 & -0.00 & -0.11 & -0.00 & -0.00 & 0.06 \\ -0.02 & -1.63 & 1.27 & 0.00 & -1.62 & 0.01 & 0.01 & -1.56 \end{pmatrix}.$$

The first principal component is close to the trajectory component of $\mathbf{u_j^1}$ and $\mathbf{u_j^2}$ (they only differ by a constant), but the second principal component has nothing in common with the velocity component of the two vectors, as we might expect. Analysing the third principal component, no correspondence can be established with the normal modes. However, the result is not surprising: trajectory and velocity data are not correlated, hence, the principal components separate into trajectory and velocity specific vectors. In addition, the velocity data usually contains more noise then the displacement data and, thus, the results obtained have a higher computational error. In this simple case, the last four principal components represent noisy directions, as suggested by the PCA analysis.

Although this method does not help us obtain the normal modes, it is useful because it shows that a completely deterministic system can be wrongly represented in the form of a noisy one, if an inappropriate method is used.

6.6.5 Autocorrelation function

Computing the autocorrelation function for the simulated data we obtain for the four particles the expressions from Figure 6.5. Note that the data analysed is not periodic (Figure 6.3). However, the autocorrelation functions suggest the presence of some periodic signals. This suggests that for a noiseless system with a reduced number of frequencies, it is possible to prove the signals existence using data autocorrelation.

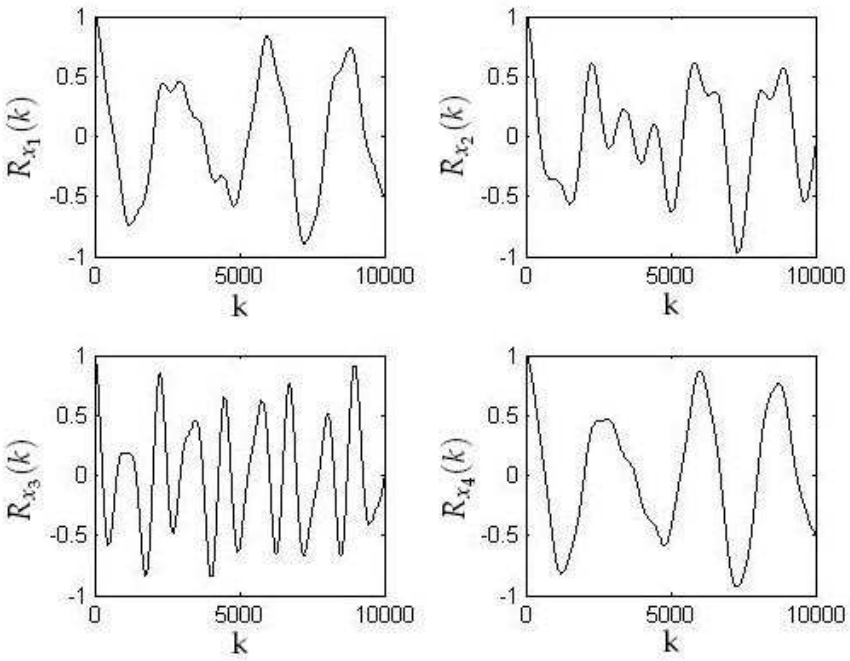

Figure 6.5: Illustration of the autocorrelation function for the displacements data of the four-particle system.

6.6.6 Fourier transform analysis

As already discussed, DFT is an alternative to determine the amplitudes, frequencies and vectors specific to each normal mode. Computing the DFT of each variable in the system, we obtain power spectra as shown in Figure 6.6. Taking into account *sine* and *cosine* properties and that DFT function is represented as a power series, we observe that the DFTs from Figure 6.6 are symmetric with respect to the middle of the frequencies interval. Thus, for our analysis we only use the first half of the frequency spectrum.

Using the four DFTs we obtain the following frequencies

$$\omega = (2.1322, 3.0066, 5.5607, 8.4522), \tag{6.6.17}$$

while the normal modes are the columns of the matrix

$$V = \begin{pmatrix} 0.0350 & 0.0309 & 0.0292 & 0.0484 \\ 0.3982 & 0.5279 & -0.9152 & -0.3392 \\ 0.0120 & 0.0039 & 0.0626 & 0.0593 \\ -0.9166 & -0.8487 & -0.3968 & -0.9376 \end{pmatrix} \tag{6.6.18}$$

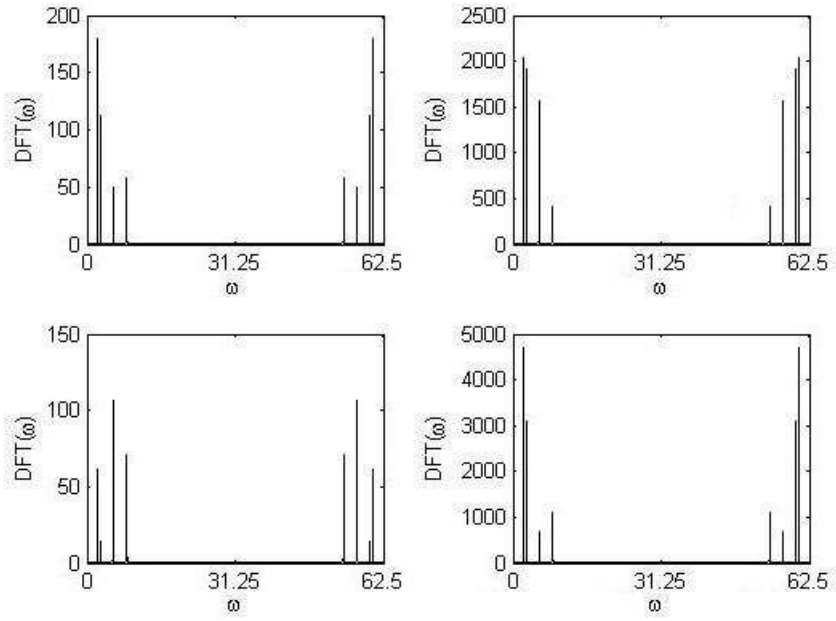

Figure 6.6: Illustration of DFT plotted against frequency, for a four-particle Hamiltonian system.

Also, taking into account that the simulation started at $t = 0$, we obtain the following amplitudes

$$C^1 = C^2 = (0.8584, 0.6216, 0.3208, 0.2273). \tag{6.6.19}$$

Finally, note that our results strongly agree with the ones from (6.6.2) and (6.6.3), the small differences in values being generated by computational errors.

6.7 Summary

In this chapter we introduced some traditional methods that can be used to analyse Hamiltonian systems. We applied these methods to data obtained for a four-particle system and discussed the results obtained. We discussed how scaling the data to unit variance influences the results obtained, as well as the influence of input data on principal components. Next, we discussed the differences between principal components and normal modes, as well as the data autocorrelation. Finally, we showed how to obtain the normal modes using the Fourier transform. All these methods are used in the next chapter to analyse

the DNA simulations obtained using AMBER and the SDE model proposed in Chapter 3.

CHAPTER 7

Traditional Analysis of DNA Dynamics

In this chapter, we use the traditional methods of PCA, data autocorrelation and normal modes, described in Chapter 6, to investigate the DNA behaviour. Recall that PCA is used in general as an aposteriori analysis method, but it can also be used as a predictive method. However, developing models capable to predict DNA trajectory at atomic level, based on PCA, can be difficult. We discuss these difficulties starting from an example of such a model. Next, we use PCA, data autocorrelation and normal modes method as alternatives of comparing results obtained using both AMBER and SDE system, to emphasize the strength of our stochastic mesoscopic model. Our analysis covers only the trajectory data obtained from DNA simulation, using the two approaches.

7.1 PCA method

Mesoscopic models of biological systems are useful for analysing measurable quantities like displacements from equilibrium, energy variations, force interactions or pressures. To reduce the complexity of the molecular system, our SDE model of DNA, introduced in Chapter 3, only considers the transverse displacements of bases. Moreover, each base of the DNA sequence is considered to be a separate point mass, thus it is impossible to obtain information about individual atoms of the DNA molecule.

For such an analysis a microscopic model is needed, which takes into consideration all atoms of a DNA sequence. Such a model has to incorporate the influence of the solvent on the system, and yet, we would like a model which reduces the simulation time compared to an all-atom MD simulation. In Chapter 2 we presented in details the DNA sequence analysed. Only 763 out of 16682 atoms in our system represent the DNA molecule, while the rest were water molecules. This means that a system with 3 degrees of freedom per DNA atom (one in each direction of the three dimensional space) and with parameters fitted to AMBER data, preserves more of the DNA properties than the SDE model.

However, reducing by twenty times the number of degrees of freedom compared to AMBER does not guarantee that the CPU time needed to simulate large systems can be reduced to days or weeks for a few microseconds of a DNA trajectory, as required to observe breathing events in a nondefective DNA molecule. First of all, recall that *Hydrogen* atoms have negligible mass compared to *Carbon*, *Nitrogen* or *Oxygen* atoms, hence such models contain hundreds of particles with different masses. Next, the DNA atoms define six different atom-to-atom interactions in each base. In addition, we need to model in a consistent manner the inter-strand interactions, based on DNA biological properties. All these tasks can be time consuming.

Yet, a reduction in dimensional space of data representation might solve the problem. As presented in the Chapter 6, this can be achieved using the PCA method, by determining the directions with highest variance and considering the other directions to be noise. Observe that the normal mode decomposition of the DNA atoms' displacements might also be useful. In what follows, we discuss how one can create a predictive model for DNA trajectory, based on PCA. However, given that PCA is mainly an a posteriori method of analysis, we do not create a new model from scratch and we only discuss the difficulties of analysing DNA trajectory data at atomic level.

7.1.1 PCA predictive models

When studying DNA dynamics, we are mostly interested in atoms' displacement coordinates and velocities. Since determining the directions with maxi-

mum variances using PCA might significantly reduce the complexity of a system, it makes sense to consider a PCA predictive model to simulate the displacements from equilibrium in our DNA sequence. Supposing that the system's collective motion is along the principal components, we do not need to specify the concrete along-chain and inter-strand interaction parameters. Such techniques can be applied at both the atomic and mesoscopic levels.

A PCA-based model that allows the prediction of DNA dynamics can be constructed as follows:

1. We first simulate the system using AMBER (or SDE) and obtain the coordinates and velocities for each atom (or base).

2. Next, we ignore the water box surrounding the DNA molecule and we consider only the dataset Y representing the displacements from equilibrium and, if needed, the corresponding velocities of the DNA particles studied. Let N be the number of particles that we analyse, that is, either the number of DNA atoms, or the number of base-pairs, for example. Then Y has $3N$ columns: one for each direction of the three-dimensional space for the N particles.

   Then, we apply PCA on Y and determine the $3N$ orthogonal directions $\{\mathbf{x_1}, ..., \mathbf{x_{3N}}\}$, from which we select the principal components based on the corresponding variances (eigenvalues) $\lambda_1, ..., \lambda_{3N}$. For consistency, if both trajectories and velocities are analysed, we need to apply data scaling before performing PCA. Let $n < 3N$ be the number of directions with high variances. Then, the principal components' space is spanned by $PC = \{\mathbf{x_1}, ..., \mathbf{x_n}\}$.

3. Assuming the DNA molecule moves along each principal component $\mathbf{x_i}$, $i \leq n$, according to simple harmonic motion, we can define the corresponding restoring forces f_i, $i \leq n$. Knowing the projections of Y on $\mathbf{x_i}$ at any moment, we consider the restoring force f_i to be proportional to the projection divided by the specific standard deviation $\sqrt{\lambda_i}$, that is,

$$f_i(Y(t_k)) \propto \frac{pr_{\mathbf{x_i}} Y(t_k)}{\sqrt{\lambda_i}}, \tag{7.1.1}$$

where t_k represents the point in time for which the data is analysed. In other words, for a row vector v of size $3N$, we have

$$f_i(v) \propto \frac{pr_{\mathbf{x_i}} v}{\sqrt{\lambda_i}}, \ \forall i \leq n. \tag{7.1.2}$$

Let $m_1, m_2, ..., m_N$ be the masses of the analysed particles and consider the vector $m = (m_1, m_1, m_1, m_2, m_2, m_2, ..., m_N, m_N, m_N)$. Then, the equations of motion are defined by

$$m. * \ddot{x} = \sum_{i=1}^{n} f_i(x), \tag{7.1.3}$$

where $.*$ represents the element by element vector multiplication.

4. Based on the equation of motion and some initial displacements from equilibrium and velocities, we can simulate the system in the space determined by $\{\mathbf{x_1}, ..., \mathbf{x_n}\}$.

Note that we can also add noise and damping to (7.1.3) to obtain a more accurate simulation, but this is not necessary if the signal-to-noise-ratio from (6.1.9) is used to determine the principal components, given that the noise amplitude in this case is negligible. In addition, an important condition needs to be satisfied at each time step: the Mahalanobis distance (see (6.2.4) for definition) computed with respect to the center of mass of our DNA sequence is supposed to place the new observation inside the volume of space defined by the principal components and the corresponding variances. This explains the choice of having the restoring force f_i proportional to $1/\sqrt{\lambda_i}$.

Recall that reducing the number of degrees of freedom might affect the principal components directions, unless unit variance data scaling is used, as discussed in Section 6.6.3. More precisely, reducing a three-dimensional system of 16682 particles to only 3×763 degrees of freedom and using a PCA-based method to predict the DNA behaviour might affect the simulations if the principal components and their eigenvalues λ_i are not correctly determined.

Even though, in Section 6.6.3, using different values for the time step needed to generate the initial datasample produces only minor changes in data variances and principal components, a simple analysis of the noisy system represented

by our 38° DNA sequence reveals the contrary. For the long simulations of 20 ns, with data obtained every 1 ps, we take into consideration only the displacements from equilibrium of A-F base-pair and its two neighbours, i.e. y_{-1}, y_0 and y_1. Applying PCA we obtain the following principal components

$$V = \begin{pmatrix} 0.0009 & 0.0203 & -0.9998 \\ -0.9999 & 0.0120 & -0.0007 \\ -0.0120 & -0.9997 & -0.0203 \end{pmatrix} \tag{7.1.4}$$

and eigenvalues $\lambda = (2.4894, 0.0172, 0.0102)$. Applying the same method for the dataset with information about each 2 fs over 2 ns, the principal components agree with (7.1.4) having values

$$V = \begin{pmatrix} 0.0012 & -0.0148 & 0.9999 \\ -0.9999 & -0.0139 & 0.0010 \\ -0.0139 & 0.9998 & 0.0148 \end{pmatrix} \tag{7.1.5}$$

while the eigenvalues become $\lambda = (3.2688, 0.0182, 0.0095)$, which implies a 31.3087% increase in variance for the first principal component. This suggests that the expressions of the restoring forces f_i, $i \leq n$, are sensitive to input data and similar reasoning as for SDE parameters fitting should be used.

However, it is impossible to apply PCA for 2 ns datasets, with data obtained each 2 fs. The three coordinates needed for each of the 763 DNA atoms represent about 17 GB of data and processing such a large dataset is time consuming and resources intensive and thus, impossible to be performed using the existent technology.

Applying PCA with unit variance scaling to the same two AMBER data samples we obtain only small differences. The principal components are

$$V = \begin{pmatrix} 0.0119 & 0.5885 & 0.0008 \\ -0.9964 & 0.8048 & 0.9965 \\ -0.0836 & 0.0769 & -0.0836 \end{pmatrix} \tag{7.1.6}$$

and

$$V = \begin{pmatrix} 0.0105 & -0.5879 & -0.0026 \\ -0.9971 & -0.8026 & -0.9972 \\ -0.0753 & -0.1012 & 0.0753 \end{pmatrix} \tag{7.1.7}$$

respectively, while the variances are $\lambda = (1.1446,\ 0.9975,\ 0.8579)$ and $\lambda = (1.1852,\ 0.9967,\ 0.8181)$, respectively, which suggests that all three components are important for our DNA breathing events hidden among the two datasets, with equal proportion of time spent breathing. Continuing our reasoning this implies all atoms and PCA directions are equally important for our DNA dynamics. Thus, the dimension of the space explored by the DNA system can not be reduced and our new system is not a viable alternative to existing MD packages, like AMBER.

Summarizing our discussion, we note that if PCA without data scaling is used, the results are sensitive to the input dataset and for a good analysis of a breathing event about 17 GB of data are needed, which is inaccessible using the existing technology. Scaling the data we obtain less sensitivity of our results to input sample and a reduction in AMBER data needed, for example, for 16 ns with information about each 1 ps only 280 MB being required. However, in this case all the directions determined are required to describe the principal component space and the desired spatial dimension reduction can not be achieved. We conclude that implemmeting a predictive model at atomic level, based on principal components, is difficult when breathing events are studied.

7.1.2 Principal component analysis of DNA trajectories

Next, we analyse all twelve base-pairs from our 38° overtwisted DNA sequence. The eigenvalues specific to the principal components obtained without data pre-treatment are

$$\begin{aligned}\lambda \;=\; &(2.5179, 0.0179, 0.0142, 0.0133, 0.0130, 0.0126,\\ &0.0118, 0.0099, 0.0089, 0.0083, 0.0080, 0.0043),\end{aligned} \tag{7.1.8}$$

which shows that only $\mathbf{PC_1}$ is important for our system. This is given by

$$\begin{aligned}\mathbf{PC_1} \;=\; &(-0.0015, 0.0011, -0.0009, -0.0029, 0.0002, -0.0008,\\ &0.9999, 0.0124, 0.0042, -0.0002, 0.0019, -0.0025)^T\end{aligned} \tag{7.1.9}$$

and clearly highlights the breathing being about $(0,0,0,0,0,0,1,0,0,0,0,0)^T$. In other words, the A-F pair breathing amplitude makes the other base-pairs dynamics less important for our system, having small variances. In fact, Figure 7.1

shows that compared to the middle site, all the other displacements from equilibrium can be considered as small amplitude noise.

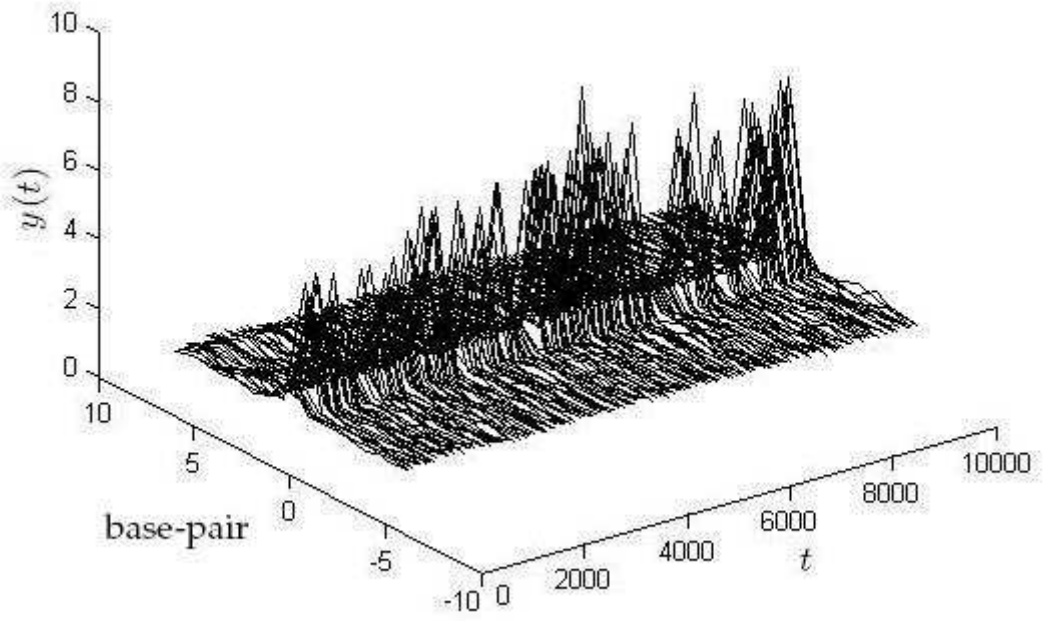

Figure 7.1: Illustration of DNA displacements from equilibrium, obtained from an AMBER simulation, plotted against time and base-pair number, for a 38° overtwisted DNA sequence.

In addition, the other directions are important only for the non-breathing base-pairs. For example, the second component as importance is

$$\begin{aligned}(7.1.10)\quad \mathbf{PC_2} &= (-0.0082, -0.0028, -0.0211, 0.0059, 0.0219, 0.0282,\\ &\quad 0.0132, -0.9691, -0.2239, -0.0213, -0.0513, 0.0752)^T,\end{aligned}$$

which is specific to y_1, while the third component is specific to y_2, being equal to

$$\begin{aligned}(7.1.11)\quad \mathbf{PC_3} &= (0.0075, 0.0065, 0.0523, -0.1234, -0.0332, 0.0093,\\ &\quad -0.0017, -0.2304, 0.9520, 0.0506, 0.1373, 0.0045)^T.\end{aligned}$$

On the other hand, when all base-pairs are considered to be equally important, that is, when data is scaled to unit variance, we obtain the following data variances

$$\begin{aligned}(7.1.12)\quad \lambda &= (1.2281, 1.0862, 1.0732, 1.0443, 0.9941, 0.9887,\\ &\quad 0.9713, 0.9660, 0.9474, 0.9422, 0.9219, 0.8366)\end{aligned}$$

and the first rescaled principal component is

$$\begin{aligned}(7.1.13)\quad \mathbf{PC_1} &= (0.0091, -0.0100, -0.0026, 0.0147, 0.0016, 0.0056,\\ &\quad -0.9946, -0.0833, -0.0432, -0.0067, -0.0196, 0.0331)^T,\end{aligned}$$

which is again almost equal to $(0,0,0,0,0,0,-1,0,0,0,0,0)^T$.

As can be seen, not only is the first PC different, but this result suggests that all directions are of similar importance in our system. In other words, unit variance data implies equally important PCA components. Note that the second component is important for most base-pairs except the breathing pair, since it has the following entries

$$\begin{aligned}(7.1.14)\quad \mathbf{PC_2} &= (-0.4137, 0.4147, 0.0515, 0.2917, 0.2775, 0.2827,\\ &\quad 0.0600, -0.1230, -0.2668, -0.3212, -0.3380, -0.3237)^T,\end{aligned}$$

but the third component is equal to

$$\begin{aligned}(7.1.15)\quad \mathbf{PC_3} &= (-0.0386, -0.0886, -0.2171, -0.1756, -0.0720, -0.0139,\\ &\quad 0.9314, 0.0165, -0.0339, -0.1144, -0.1538, 0.0412)^T,\end{aligned}$$

which means $\mathbf{PC_3}$ is mostly important for the breathing pair. This explains the importance of all directions in the system when data is scaled before applying PCA.

Next, we compare these results with the ones obtained by applying PCA for a SDE simulation for a 38° overtwisted DNA strand. The PCA results obtained for our SDE simulation are similar with AMBER case. If data pre-treatment is not applied, the data variances are similar with the ones from (7.1.8), that is,

$$\begin{aligned}(7.1.16)\quad \lambda &= (2.5260, 0.0325, 0.0178, 0.0176, 0.0171, 0.0164,\\ &\quad 0.0159, 0.0156, 0.0155, 0.0150, 0.0144, 0.0007),\end{aligned}$$

while the first principal component is similar to $(0,0,0,0,0,0,-1,0,0,0,0,0)^T$, being

$$\begin{aligned}(7.1.17)\quad \mathbf{PC_1} &= (-0.0003, -0.0009, 0.0009, 0.0017, 0.0006, -0.0119,\\ &\quad -0.9999, -0.0120, -0.0017, 0.0006, -0.0006, -0.0010)^T,\end{aligned}$$

which is also similar to (7.1.9). $\mathbf{PC_1}$ value confirms the conclusion from Chapter 5 that SDE simulations are more regular than AMBER results. Indeed, note that $\mathbf{PC_1}$ from (7.1.17) has almost symmetric entries with respect to its seventh entry and suggests that breathing influences the neighbouring pairs due to the defect in the along-chain interactions, but does not influence the other base pairs.

Moreover, in the SDE model we do not take into account which type of base-pairs compose the DNA sequence and use the same value for the inter-strands spring coefficient γ. On the other hand, AMBER simulations and base-pairs displacements variances depend on the type of bases composing the DNA sequence. The first, sixth and tenth base-pairs are C-G pairs and are represented by very small values in the principal component, which can be explained by the three hydrogen bonds compared with the two bonds of an A-T pair and only 1 bond for A-F base-pair. Also, note that the entries of the first principal component from (7.1.9) and (7.1.17) are different, which might be due to the different initial conditions in the two systems. However, the direction and the amplitude specific to each base-pair are more important for our comparison than a scaling factor of -1. In addition, the SDE and AMBER simulation results are similar (compare Figures 7.1 and 7.2), given that the base-pairs do not move just along $\mathbf{PC_1}$, since the PC eigenvalues suggest small variations in the other directions.

Note that, as in AMBER case, the others directions are specific to non-breathing base-pairs. For example, the second component, that is,

$$\begin{aligned}\mathbf{PC_2} \;=\; &(0.0203, 0.0172, -0.0030, -0.0232, 0.0617, 0.7026,\\ &-0.0170, 0.7047, 0.0617, 0.0145, 0.0191, 0.0032)^T,\end{aligned} \tag{7.1.18}$$

is again specific to y_1, but also to y_{-1} due to the SDE system symmetry.

Results confirming SDE regularity (compared to AMBER) are also obtained when data scaling is used with PCA. These results suggests that in such a case all principal components have to be taken into consideration, given the closeness of the eigenvalues

$$\begin{aligned}\lambda \;=\; &(2.0083, 1.0991, 1.0818, 1.0688, 1.0125, 0.9907,\\ &0.9695, 0.9646, 0.9503, 0.9243, 0.8871, 0.0430).\end{aligned} \tag{7.1.19}$$

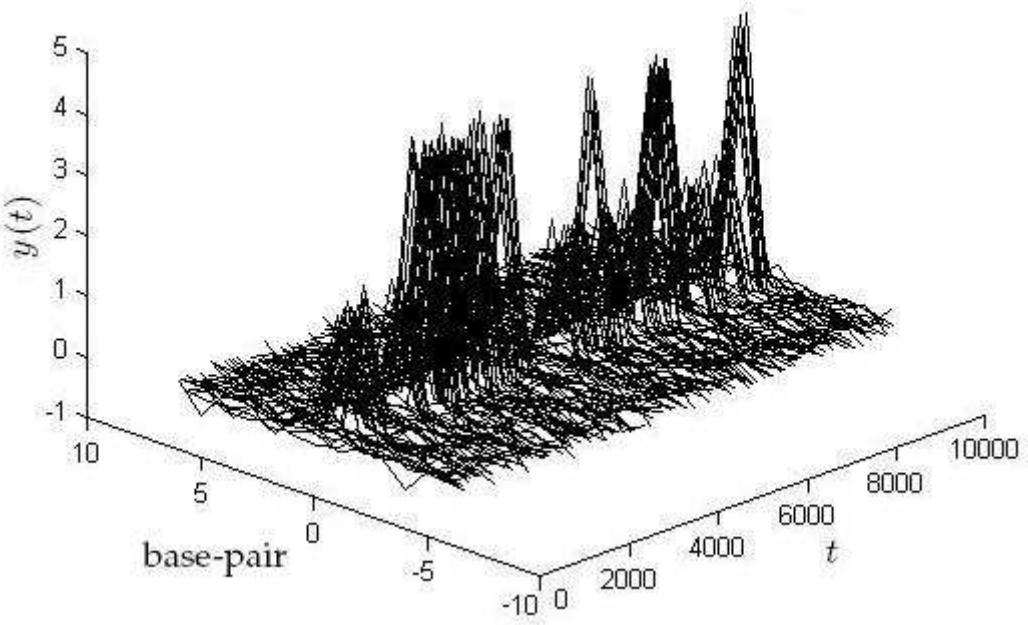

Figure 7.2: Illustration of DNA displacements from equilibrium, obtained from a SDE simulation, plotted against time and base-pair number, for a 38° overtwisted DNA sequence.

This suggests that the first PC is dominant, which happens due to breathing, given that it is similar to $(0,0,0,0,0,0,-1,0,0,0,0,0)^T$, more precisely,

$$\begin{aligned}(7.1.20)\quad \mathbf{PC_1} \;=\; & (-0.0081,-0.0076,0.0028,0.0117,-0.0213,-0.2621,\\ & -0.9278,-0.2629,-0.0249,-0.0043,-0.0080,-0.0032)^T.\end{aligned}$$

Even though the last principal component is noise with amplitude given by $\lambda = 0.0430$, the last ten components are of similar importance, having variances close to 1. Even if this result differs from (7.1.12), we still obtain that most components are important for our system, since the SNR value – see (6.1.9) for definition – is not large enough to ignore any of the orthogonal directions. Next, we observe the same decrease in entry values for the C-G base-pairs in (7.1.13), that is, first, sixth and tenth entry, but for the SDE system the rescaled first principal component can be considered again to be symmetric.

Finally, the importance of most of the PC directions is justified, for example, by the second component, which influences the breathing pair and equals

$$\text{(7.1.21)}\quad \mathbf{PC_2} = (-0.2737, -0.0644, 0.0929, 0.1436, 0.0170, 0.0431, -0.6385, 0.0336, -0.2193, -0.3800, -0.3429, -0.4131)^T.$$

These results emphasize again the strength of our SDE model. The similar variances and principal components values of our AMBER and SDE simulations, show that the two approaches are close one to the other.

7.2 Data autocorrelation

Similar results between AMBER and SDE simulations are obtained when computing the data autocorrelation function for the A-F base-pair. Figures 7.3, 7.4 and 7.5 present a comparison between the two cases, for the three of the eight twist angles in the range 30°-40° discussed in Part I, that is, 30°, 35° and 38°, respectively. The figures for the remaining angles can be found in Appendix B.1. This comparison emphasizes that for AMBER simulations, the information about the initial system conditions is lost after at most 1 ns, while for the SDE system only about 0.5 ns is necessary.

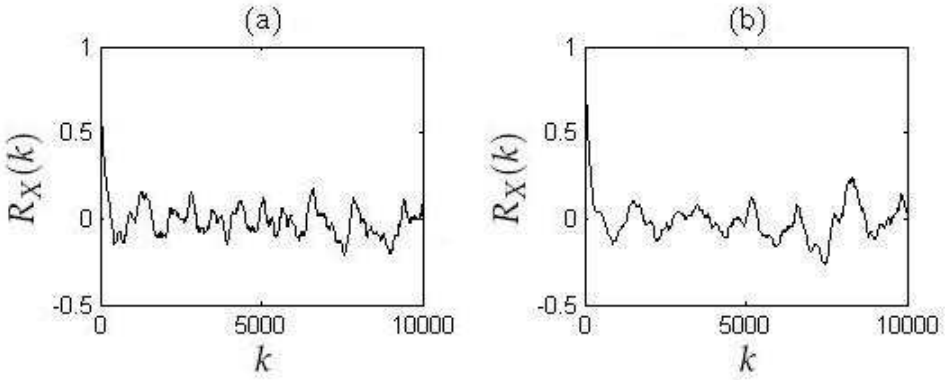

Figure 7.3: Illustration of autocorrelation function, for a 30° undertwisted DNA, obtained using (a) AMBER model and (b) SDE model.

Moreover, the spikes from each graph might be correlated with the formation of breathing events. Note that the SDE simulations of a 35° undertwisted DNA sequence from Figure 5.12 contains many short and frequent breathing events

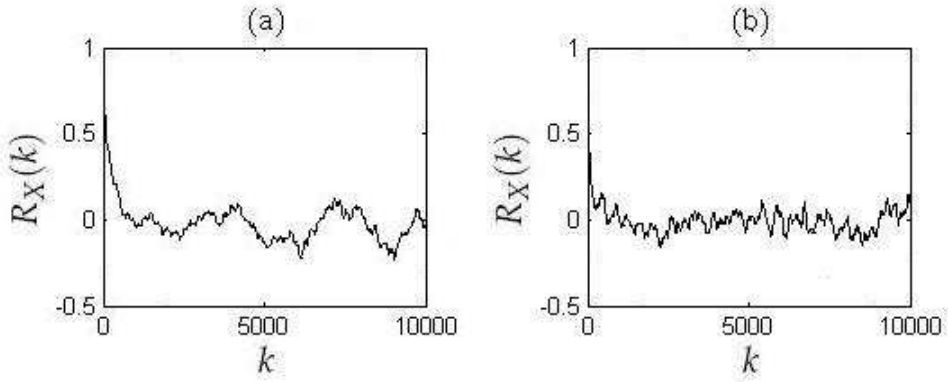

Figure 7.4: Illustration of autocorrelation function, for a 35° undertwisted DNA, obtained using (a) AMBER model and (b) SDE model.

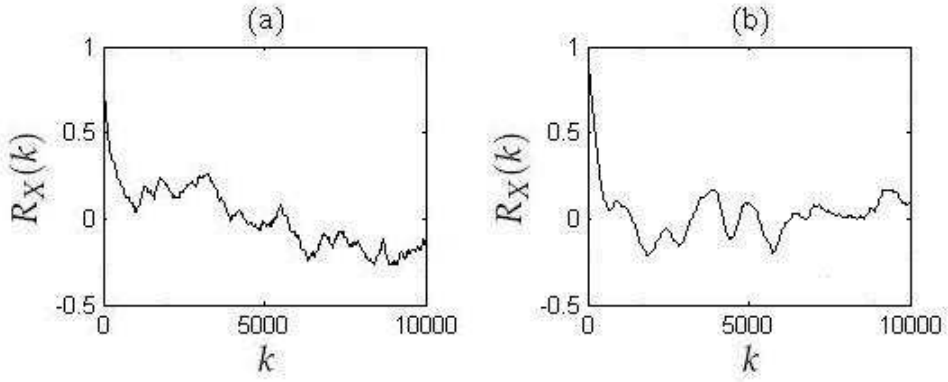

Figure 7.5: Illustration of autocorrelation function, for a 38° overtwisted DNA, obtained using (a) AMBER model and (b) SDE model.

and is the only simulation which can be considered inconsistent with the DNA behaviour observed in AMBER simulations. Figure 7.4(b) is also inconsistent when compared to the other autocorrelation functions presented here. Observe also that for the angle for which breathing represents an important proportion of the simulation time, that is, the normal twist angle of 36° and the overtwisting angles of 38° and 40°, the number of spikes from the data autocorrelation expression is reduced compared to the undertwisted angles. This can be explained by the reduced number of breathing events in these simulations – see Chapter 5 for more details. In addition, for the normal twisted and overtwisted DNA the oscillations from positive to negative values in the data-autocorrelation function are not that frequent as in the case of the undertwisted DNA strands.

Finally, recall that in Chapter 6 data autocorrelation is presented as a method for determining hidden periodic signals. However, we expect the displacements from equilibrium in our system to be a sum of several normal modes with different frequencies, since our DNA sequence, can be considered a Hamiltonian system.

7.3 Normal modes

The equations of motion of each base-pair were obtained in Chapter 3 from the system Hamiltonian described by (3.1.14). Hence, determining the normal modes and their corresponding frequencies is the next step in our comparison between AMBER and SDE simulations. We expect to determine similar representations of the two systems. More precisely, we hope to find a few large amplitude modes with low frequencies, related to DNA breathing, as well as DNA chain bending or twisting, and possibly corresponding to the first few principal components.

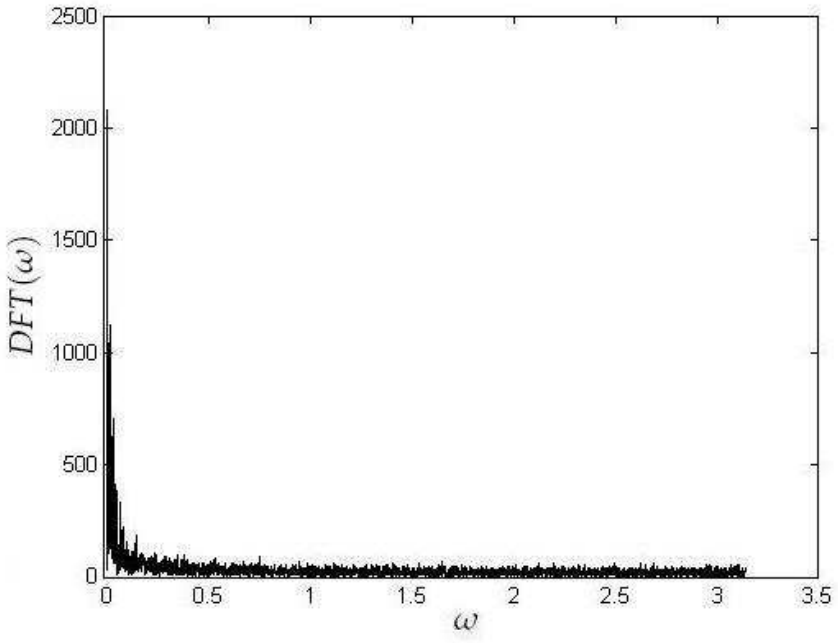

Figure 7.6: Illustration of the DFT, for a 35° undertwisted DNA, obtained using AMBER.

Using the FFT algorithm to determine the DFT of $y_0(t)$ for a 35° undertwisted

DNA, we obtain a surprising result. Figures 7.6 and 7.7 show that, for both AMBER and SDE simulations, the DFTs do not possess a few well-defined peaks.

The differences between the amplitudes of the DFT values from Figure 7.6, representing the AMBER simulation, and Figure 7.7, representing the SDE simulation, can be ignored, since the DFT expressions show a perfect agreement of the DNA behaviour in both cases. More precisely, the DFT expressions imply that DNA exhibits the so-called "self-organised criticality" property, that will be discussed in next chapter.

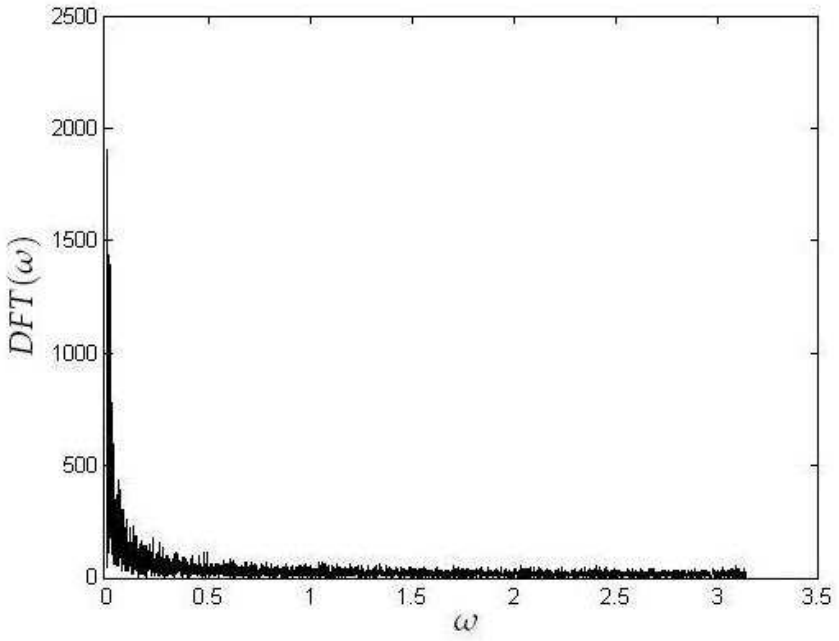

Figure 7.7: Illustration of the DFT, for a 35° undertwisted DNA, obtained using the SDE model.

Note that similar results, suggesting a self-organised DNA behaviour, are obtained for seven other twist angles in the range 30°-40°. Details can be found in Appendix B.2.

7.4 Summary

Constructing based on PCA a model to predict the DNA behaviour and which considers all atoms of a DNA sequence is difficult, since the data enclosing

representative information on breathing events and needed to obtain accurate results can not be processed with existing technology. However, we can use PCA to compare the AMBER and SDE simulations in terms of principal components and to emphasize the similar behaviour of the two models, as well as some homogeneity differences. Moreover, in both cases, PCA applied without scaling the data suggest that only one direction is important for our analysis, while using data scaling we obtain that all directions in the system are equally important – an unexpected result.

The autocorrelation function also confirmed the similarities between the two approaches. This final result was also confirmed by the form of the Fourier Transform found for the displacements from equilibrium of the A-F base-pair, but which does not reveal any clear breathing frequency. In the next chapter we further investigate this property.

CHAPTER 8

Self-organized criticality

A lot of dynamical systems evolve to a steady state (equilibrium solution) or a limit cycle. More complicated large-time attracting sets are often characterised by strange attractors [104], which exhibit chaos. However, there are other ways of characterising large time behaviour such as the classes found by Wolfram [127] when studying cellular automaton. Based on an empirical study, he identifies four qualitative classes of systems, that is, spatially homogeneous systems, periodic structures, systems with chaotic aperiodic behaviour and complicated localised and possibly propagating structures. In what follow, we focus on the latter category.

Self-organized criticality (SOC) [6] is a property specific to certain dynamical systems which have a critical point as an attractor. In physics, a critical point specifies the conditions, such as temperature, pressure or composition, at which a phase boundary is not valid anymore. Here, by phase we understand a state of a system for which the physical properties of a component are uniform. In other words, a critical point refers to a system configuration to which the system evolves without ever approaching one fixed equilibrium state. For more details and definitions for critical phenomena and phase transition see [44].

When analysing a large system, we aim to reduce its complexity to a few degrees of freedom, for which the coupling can be defined in a general manner by obtaining some averaged behaviour over the ignored quantities and their corresponding interactions within the system or with the surrounding environment. For dynamical systems, dimensional reduction is also called "slaving

principle" [53] and leads to the the study of low-dimensional attractors. This is often a straight forward method. For example, "fast modes" at equilibrium can be slaved to a few slowly evolving modes. However, sometimes a system responds on both fast and slow timescales, even at large times, and we require a new theory, such as the idea of self-organised systems, whose behaviour cannot be explained using reduced models.

Systems having the SOC property present a spatial or temporal macroscopic behaviour invariant when a scale factor is used. This property is called scale-invariance [141] and suggests that we do not require exact parameter values to characterise the critical points of a phase transition. Phase transition actually means passing from a steady state to a non-equilibrium one. In general, the total number of states is finite and the transitions can be characterised using a cellular automaton structure [30].

Although there is not a well defined class of systems having SOC property, it is typically observed in complex systems with slowly-driven non-equilibrium behaviour, for which the causes of an event taking place in a system cannot be explained through some parameter values. Several studies of SOC show that scale-invariant phenomena can be determined at critical points, but not necessarily at any critical point. There are two important categories of such phenomena: fractals [92] and power laws [88]. The first category involves geometric shapes, which can be split into parts that are reduced-size copies of the initial shape. The second deals with frequency dependent quantities and, hence, is relevant for some Hamiltonian systems analysis. However, note that self-organised systems are always at criticality, but not all critical systems are self-organised.

Bak et al. [5] demonstrate numerically that systems with extended spatial degrees of freedom evolve into barely stable states and claim that SOC is the mechanism behind such behaviour. The attractor in their system is not dependent on the model parameters and suggests that the so-called "flicker noise", also known as $1/f$ noise, does not require fine tuning. In [6], they use a simple automaton to determine the relation between critical phenomena and features like power laws, fractals, and, last but not least, $1/f$ noise. They discuss the dynamics of critical states, for which the power spectrum $S(f)$ (where f represents

a frequency) scales with $1/f$ at low frequencies. They note again that changing the value of system parameters does not affect the critical point emergence, which implies that systems with such features present SOC behaviour.

In general, for a noisy system the power spectrum has the form $S(f) = cf^{-\beta}$, where c is a constant. The noise present in the system can be classified in three important categories as follows:

- white noise, for $\beta = 0$;
- pink noise, for $\beta = 1$;
- red noise (also known as Brownian noise), for $\beta = 2$.

However, the term "$1/f$ noise" is widely used to refer to any noise with a power spectral density $S(f) \propto f^{-\beta}$, with $0 < \beta < 2$. For $1/f$ noise that occurs in nature, β is usually close to 1.

In other words, a system exhibits a self-organised behaviour if the dynamics of the critical states scale into a power law at low frequencies, emphasizing the presence of flicker noise. Such systems also have the scale-invariance property, that is, the emergence of critical points is not affected by changes in system parameters values. This implies that, in general, the SOC behaviour of a system cannot be explained only be the parameters values of a reduced model and, thus, a more detailed analysis is needed. Recall that, for the DNA system analysed in Part I of this thesis, the parameters vary with twist angle. However, breathing (the critical state) occurs for all twist angles analysed, hence the spectrum analysis could suggests a self-organised DNA behaviour.

8.1 Power laws

As already mentioned, power laws sometimes arise in frequency analysis. A power law defines a relation between two quantities and when one of these quantities is the frequency of an event, this relation becomes a power-law distribution, with the effect that increasing an event's size results in decrease in

its frequency. In most cases, these mathematical relations are defined using a polynomial-like representation. Note that not any polynomial preserves the scale-invariance property, only ones having the form

$$PL(x) = cx^n + o(x^n), \tag{8.1.1}$$

where c and n are two real constants, while $o(x^n)$ is an asymptotic function. Observe that such functions are indeed scale-invariant, given that

$$PL(\alpha x) = c\alpha^n x^n + o(\alpha^n x^n) = \alpha^n(cx^n + o(x^n)) = \alpha^n PL(x), \tag{8.1.2}$$

for some constant α.

The most common way of identifying a power law representation is the logarithmic one. Applying the natural logarithm function to both sides of (8.1.1) we obtain

$$\log(PL(x)) = n \log x + \log c, \tag{8.1.3}$$

which shows that the logarithmic representation of the frequency of an event is a linear function of the log-frequency.

Note that we defined the power law in a single variable, but it is also possible to have multi-variables power laws. Moreover, power laws are characteristic to natural processes, and the asymptotic function $o(x^n)$ in fact represents small deviations from the polynomial expression, possibly caused by noise or measurement errors.

Finally, observe that a power law for which $-2 < n < 0$ is characteristic to $1/f$ noise, which means that the critical points in a system can be determined whenever the frequency spectrum has the form of a power law.

8.2 SOC examples

In the scientific literature, several systems exhibiting SOC behaviour have been identified. Bak et al. [5, 6] study the dynamics of a damped pendulum and the slope of a sandpile, respectively, and determine critical points in the systems.

Avalanches in a one dimensional sandpile are also analysed by Chapman et al. [27, 28]. They determine the distribution of energy discharges due to internal reorganization, whose power law form shows that the system is self-organized. Moreover, in [28] a one-dimensional avalanche sandpile algorithm is presented for transport in a driven dissipative confinement system, which allows further SOC analysis. However, in [29] they classify a broad range of systems that fall under the general description of SOC and argue that some, but not all, of the results related to the magnetosphere are suggestive of, but not sufficient to confirm SOC behaviour.

The range of systems presenting SOC properties varies from sandpiles to biological systems and even electric current. Banerjee et al. [8], for example, study the noise profile of a Voltage-dependent anion channel in open channel state and the power spectrum of current indicates power law noise of $1/f$ nature. The widespread self-organized phenomena of earthquakes has also attracted the attention of scientists. Olami, Feder and Christensen [91] developed one of the first models of earthquakes, known as the OFC model. This cellular automaton model is based on the Gutenburg-Richter law, which represents a statistical statement expressing the relationship between the magnitude and total number of earthquakes in a given region over a fixed time period. A power-law relationship is observed for the number of earthquakes with energy greater than a fixed energy E_0. Caruso et al. [24, 25] use this model to investigate the SOC properties of small-world and scale-free networks. However, the critical behaviour of the OFC system is later analysed by Klein and Rundle [73], as well as by Christensen [31], one of the model developers.

Bak et al. [7] investigate the distribution of waiting times between earthquakes occurring in California and reach the conclusion that it obeys a simple unified scaling law, valid from tens of seconds to tens of years. Weatherley et al. [123] study the dynamics of a crack-like automaton, in which all stress is transferred from a rupture zone to the surroundings, as well as a partial stress drop automaton, in which only a proportion of the stress within a rupture zone is transferred to the surroundings. The mean spectral density of a stress deficit field exhibits in both cases a power-law relationship with respect to the spatial wavenumber.

Bak et al. [4] developed one of the first forest-fire models. Starting from a d-

dimensional hypercubic lattice with L^d sites, they define a probabilistic cellular automaton, in which a site is either a tree or an empty space. A tree starts burning only if one of its neighbours burns and after a tree is burned its site becomes empty. Moreover, a tree grows at an empty site with probability p. Using this model, they find that the fire-fire correlation function is a power law and for the limit $p \to 0$ the fire correlation length diverges and the system becomes critical. Drossel and Schwabl [45] improve this forest-fire model adding a tree lightning probability f. Given that the time scales of tree growth and burning down of forest clusters are separated, when $f \to 0$ the system is driven into a self-organized critical state. They reach the conclusion that for a two-dimensional system, the critical state assumes the maximum energy dissipation.

This example shows that separation of the timescales is present in many self-organising systems. Fires spread on a fast timescale, but trees grow on a slow timescale. After a large time, much empty space is created and there is a long time before it is repopulated with trees, but eventually it will become vulnerable to another fire. Hence, there is repetition, but not at any fixed frequency. Rather, there is a random timing of fires and this is related to the size of fires.

Next, Sinha-Raya et al. [113] replace the stochastic ignition generated by lightning with a deterministic threshold for auto-ignition, but the system properties remain unchanged. In addition, they find using this model multifractality in the trees distribution. Another model closely related to the Drossel-Schwabl was developed by van den Berg and Jsrai [16], by considering instantaneous ignition of the trees. This allows them to prove that regardless of the initial system configuration, after a time of order $\log(1/f)$ the density function is of order $1/\log(1/f)$. Brouwer and van den Berg [15] developed another forest-fire lattice model, in which tree lightning implicitly makes vacant the occupied cluster, and study the system using the rates of a site being hit by lightning. The self-organized critical behaviour is observed again for lightning rates close to zero.

Pueyo [99] developed a wild-land fire model to forecast the effects of climate change on catastrophic events. He studies the fire size statistical distribution for weather fluctuations in a boreal forest region and predicts the fire regime in this region, for an instance of possible climate change scenario, to have much

larger burning surfaces than the largest fires that currently occur. Caldarelli et al. [20] investigate the statistical properties of wild-land fires to determine whether spread dynamics relate to a simple invasion model. Using satellite images of three fire scars they study the fractal dimension and observe that the burned clusters behave similarly to percolation clusters on boundaries and look denser in their core.

Note that forest-fire approaches were adapted to study several other natural phenomena. Consolini and De Michelis [35], for example, used a revised forest-fire cellular automaton to study the nonlinear dynamics of the Earth's magneto-tail, while Rhodes and Anderson [103] define individual-based lattice epidemic models, starting from a forest-fire automaton, to simulate the spreading of epidemic processes, such as measles.

On the other hand, Krink et al. [75] apply the SOC concept to control the mutation at an individual level and extinction at the population level in evolutionary algorithms (EA), which improves a previously introduced mass extinction model, without any additional computational costs. Maslov et al. [82] also study SOC properties of a simple evolution model by establishing the relationship between spatial fractal behavior and long-range temporal correlations. They also discuss similar relationships for several other self-organized (and non-self-organized) critical phenomena, such as directed percolation or interface depinning.

Biological systems represent another category interesting from SOC point of view. Kishimoto et al. [72], for example, present a critical gradient transport in a tokamak plasma model that describes self-organized relaxed states, as well as some of the important aspects in tokamak transport, such as Bohm diffusion, radially increasing fluctuation energy, heat diffusivity, or intermittency of the wave excitation. The brain is another biological system placed in the category of self-organisation, as discussed by Werner in [125]. He states that the theory of non-equilibrium phase transitions can serve as an informative approach for elucidating the nature of underlying neural mechanisms.

Next, the self-organisation characteristics of proteins were studied by Phillips [97, 98], for example, who considers that regarding proteins as archetypical examples of SOC, their complexity is simplified. Nykter et al. [89] developed an

algorithm to assess gene expression dynamics in macrophage criticality, providing in this way a compelling evidence for this general principle of dynamics in biological systems. This method, based on algorithmic information theory, is validated using several networks with well-known self-organised behaviour.

A biological structure, directly involved in protein and genes related processes, is represented by DNA, which also exhibits SOC property. Selvam [108–110] studies the distribution of bases in a DNA sequence. Analysis of frequency distributions of bases in Drosophila DNA [108] show that the fractal fluctuations self-organize to form an overall logarithmic spiral trajectory with the quasiperiodic Penrose tiling pattern, for the internal structure. In [109], the power spectra of human DNA shows that the C-G base-pair frequency distribution exhibits the universal inverse power law form of the statistical normal distribution for the 24 chromosomes. Similar results are obtained in [110] about the C-G base-pair frequency distribution in the DNA of *Takifugu rubripes*, which is the Puffer fish.

Cingolani et al. [32] use DNA bases to describe a new strategy to exploit self-assembled solid-state biomolecular materials. The biomolecular semiconductors consisting of DNA bases in this top-down approach are self-organized and interconnected by planar metallic nanopatterns. Jan et al. [66] propose a design and realization method to solve the constrained multi-objective problem via a self-organizing PID (proportional, integral and derivative feedback) control design. Their algorithm is based on an idea using the structure of biological DNA molecules to map the parameters and the structure of PID controllers into DNA strings.

Another study made by Sotolongo-Costa et al. [114] uses irradiation of DNA molecules with electrons and neutrons at different doses to obtain the DNA double strand breaking. They measure the length of the resulting fragments and reach the conclusion that the collection of fragment sizes obeys a power law distribution. Naimark [87] discusses possible relations of structural-scaling transitions in ensembles of localized distortion modes within the replication and transcription phenomena. They state that the unique properties of DNA might be explained by the inhomogeneity of DNA fluctuations and their evolution into collective modes, since the localised distortion modes can be as-

sociated with structural-scaling transitions, which represent a type of critical phenomena. But waves are not important only in DNA. Jung et al. [67] study noise-induced spiral waves in Astrocyte Syncytia and find a power law distribution of wave sizes, reaching the conclusion that the process that creates the waves has no preferred spatial or temporal (size or lifetime) scale.

Finally, Harris et al. [55] analyse the configurational entropy of a DNA molecule based on the entropy estimation for a Gaussian configuration given by Schlitter [106], which helps investigating if a steady state has been reached during a simulation. They show that the estimate of the entropy S_n depends on the number of data points n and this relation is a power law. Moreover, they determine that the gradient of the line characterising $\log(S_n)$ is $-2/3$.

In conclusion, studying in detail the SOC behaviour of DNA dynamics might give more information about the nature of bubbles, as well as about wave formation and nucleation. Indeed, from bubbles size to breathing frequency, a whole range of measurable quantities might reveal the SOC properties in DNA. However, we note that Frigg [49] considers that SOC cannot be a general theory, like Newtonian mechanics, for example, and the gross simplifications of the models presenting SOC behaviour cannot represent a description of the target system. Therefore, from Frigg's point of view, SOC models can only be of heuristic value, opening new doors for scientific research.

8.3 Fourier Transform and power law

Recall that breathing, in general, is charactherised by identifying the specific modes, which, as discussed in Chapter 6, can be determined using the Fourier transform. Moreover, a frequency-based analysis is useful for determining the self-organised behaviour of a system. Thus, if the Fourier Transform, also called power spectrum, can be written in the form of a power-law, then our system has SOC property, which might suggest that breathing is caused by some natural complex process.

Kertkszt and Kiss [71], for example, analyse the model proposed by Bak et al. [6], which explains the fractality emerging spontaneously in nature and the

flicker noise. Starting from the Fourier Transform they determine the mean energy density spectrum of sandpile avalanches with a given size s, but given that the avalanches do not interact, the total power density spectrum is in fact the weighted sum of the individual contributions. Finally, they reach the conclusion that the values of the weights influence the noise spectrum exponent, when certain conditions are satisfied.

Suppose that we have a data sample X for which we compute the DFT. Denote by ω the frequency variable and by $DFT(\omega)$ the corresponding Discrete Fourier Transform. If $DFT(\omega)$ has the form $c\omega^n$ as required by (8.1.1), then our system is supposed to be self-organized. Such analysis is crucial in determining if an event, such as DNA breathing, is caused by a judicious fit of parameters to real data or is in some sense more generic. In our case, if SOC behaviour is observed in both MD and SDE simulations, then this suggests that the models are robust in their parameters values, that is, some change in their values will not affect the critical behaviour.

The frequency-based analysis of breathing events from Chapter 7 suggests that our DNA molecule exhibits self-organised criticality, which we shall now investigate in more detail by finding the DFTs and the form of their associated power laws. This could be due to complex interactions taking place at atomic level in DNA, which can not be fully explained by the parameters of the reduced SDE model.

8.3.1 DFT power law coefficients

To determine how the DFT of the A-F base-pair dynamics, that is, $y_0(t)$ defined in Chapter 3, depend on the frequency ω, we plot the log-frequency against $\log(\omega)$ to investigate the validity of the power law assumption, that is,

$$\log(DFT(\omega)) = -\beta \log(\omega) + c, \tag{8.3.1}$$

where $c \approx log(DFT(1))$ is a scaling factor.

Figures 8.1(a) and 8.2(a) illustrate the log-DFT representation obtained from MD simulations, while Figures 8.1(b) and 8.2(b) show the results obtained by

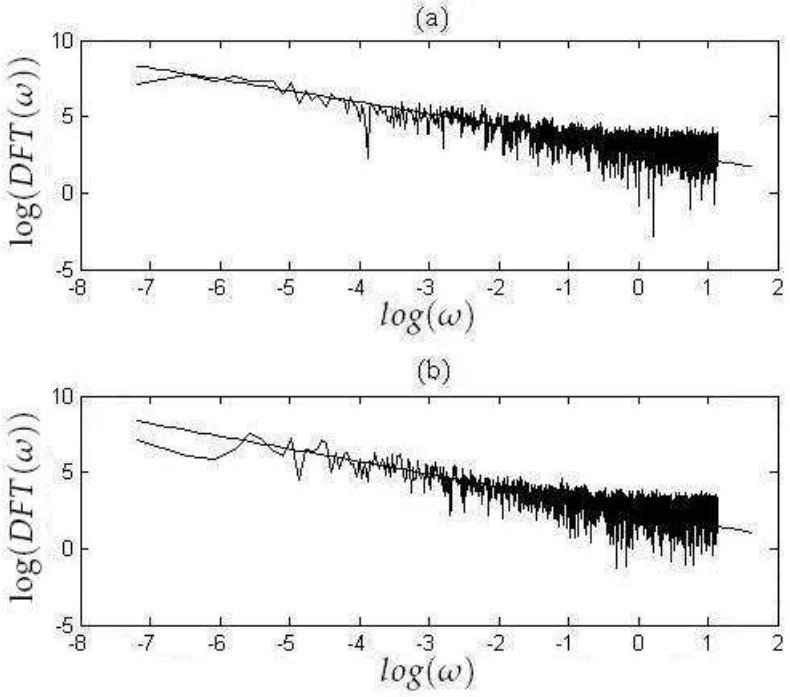

Figure 8.1: Illustration of the DFT, for a 34° undertwisted DNA, obtained using (a) AMBER model and (b) SDE model.

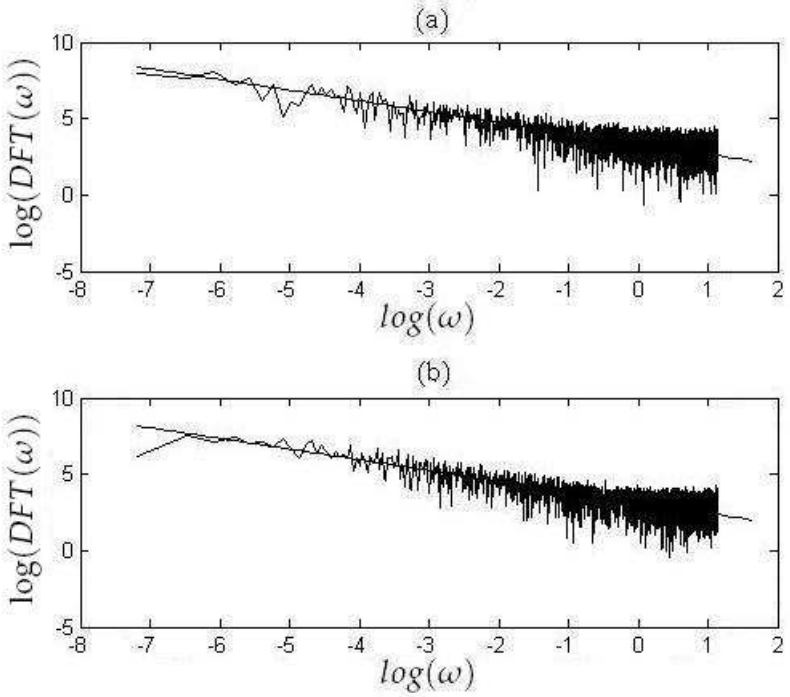

Figure 8.2: Illustration of the log-DFT function, for a 40° overtwisted DNA, obtained using (a) AMBER model and (b) SDE model.

analysing SDE data, for the 34° and 40°, respectively, twist angles. As can be seen there is an excellent agreement between the two sets of data at both twist angles analysed (see also Appendix B.3 for the rest of the twist angles). Note that $\omega = 0$ was not considered in our analysis, given that in this case $\log(\omega) \to -\infty$.

Recall that, in general, the critical behaviour is observed for smaller frequencies, but we also plotted the DFT values for large frequencies in our data samples. Typically, we observe power law behaviour for $-7 < \log(\omega) < -2$, representing a range of 0.1353 in ω. Moreover, for large ω, that is, $\log(\omega) > -2$ the log-DFT has an increased range (of about 4 units) compared to the low frequency values.

The best fit of the slope and the intercept of the lines corresponding to each log-DFT is obtained by minimizing the total deviation of the data from the line. However, given the form of the log-DFTs we fitted these values by eye, considering the low frequencies more important than the large ones. The gradients of the log-log plot of DFT versus ω are summarised in Table 8.1 and suggest the presence of $1/f$ noise in our data.

Twist angle	β_{AMBER}	c_{AMBER}	β_{SDE}	c_{SDE}
30°	0.725	3.1949	0.750	2.7463
32°	0.700	3.4574	0.725	3.1548
33°	0.725	3.1591	0.775	2.7920
34°	0.750	2.9417	0.825	2.4083
35°	0.750	2.8807	0.775	2.9226
36°	0.775	2.9626	0.825	2.4838
38°	0.700	3.4530	0.875	2.4566
40°	0.700	3.3524	0.700	3.1524

Table 8.1: Values of the gradient β and the intercept c of the log-log plot of $DFT(y_0)$ against ω, for 10 ns of data, with information about each 1 ps obtained using the AMBER and SDE models, respectively.

If the constant c is just a scaling factor which is not relevant for our analysis, the β values suggest an average value of $3/4$ for the AMBER data (since all lie

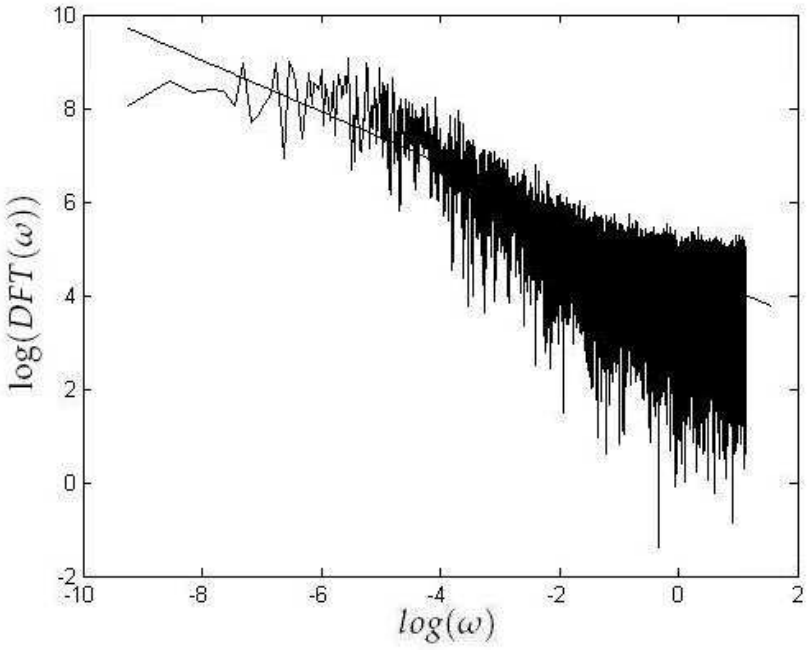

Figure 8.3: Illustration of the DFT, for a 30° undertwisted DNA, obtained from a 100 ns SDE simulation.

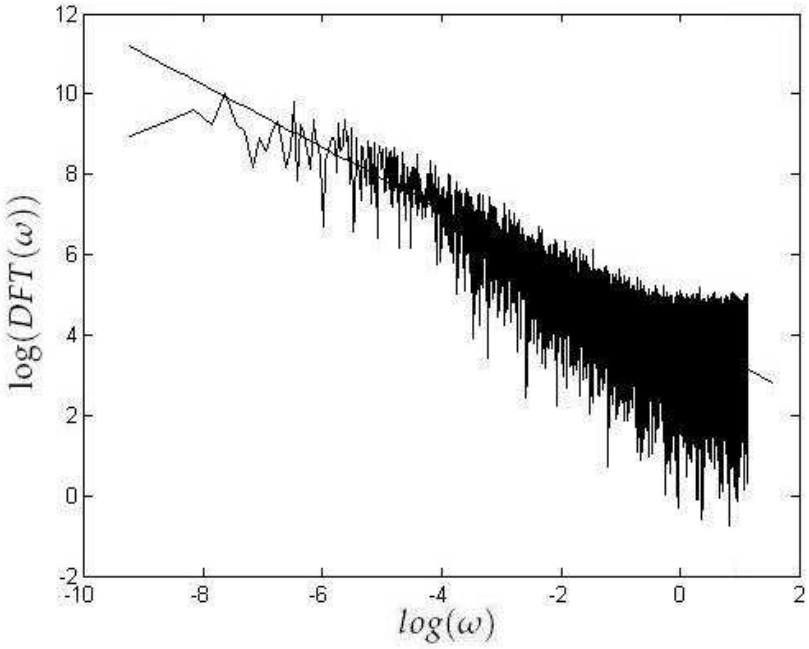

Figure 8.4: Illustration of the DFT, for a 38° overtwisted DNA, obtained from a 100 ns SDE simulation.

in the range $(0.7, 0.775)$) and $4/5$ for SDE model, respectively (the latter being more widely distributed across the interval $(0.7, 0.875)$). Analysing the two SDE

simulations of 100 ns presented in Section 5.5 we observe the same behaviour for long-time dynamics, as can be observed in Figures 8.3 and 8.4. We obtain $\beta = 0.725$ for the 30° undertwisted DNA sequence and $\beta = 0.775$ for the 38° overtwisted DNA molecule.

Even though these values are close to 1, further analysis shows that it is possible to improve them, by analysing a more detailed layer dataset, namely, the short 2 ns AMBER simulations, with data about each 2 fs, that were used in Chapter 4 to fit the SDE model parameters. Indeed, Table 8.2 suggests that the average value of β is in fact 0.93, which is much closer to 1 that the values presented in Table 8.1, the small difference being possibly generated by numerical computational errors. The increase in β from 0.75 in Table 8.1 (where a timestep $\Delta t = 1$ ps was used) to 0.93 in Table 8.2 (where we use a $\Delta t = 2$ fs timestep) suggests that we might expect $\beta \approx 1$ when $\Delta t \to 0$. Note also the slightly tighter clustering of data in Table 8.2 for AMBER model. Analysing a similar dataset obtained using the SDE model, we observe the same increase in β, that is, from an average of 0.79 in Table 8.1 to 0.91 in Table 8.2. This shows once again that our mesoscopic model is capable of reproducing with accuracy the DNA behaviour.

Twist angle	β_{AMBER}	β_{SDE}
30°	0.920	0.900
32°	0.920	0.895
33°	0.900	0.910
34°	0.940	0.920
35°	0.930	0.900
36°	0.930	0.910
38°	0.940	0.940
40°	0.950	0.905

Table 8.2: Values of the gradient β of the log-DFT function, for data obtained using AMBER and SDE data over 2 ns, with information about each 2 fs.

Next, we analysed the nonbreathing pairs in our system, for a 38° overtwisted

DNA, for both AMBER and SDE simulations. Figure 8.5 suggests that the DFTs of $y_2(t)$ also have power law forms.

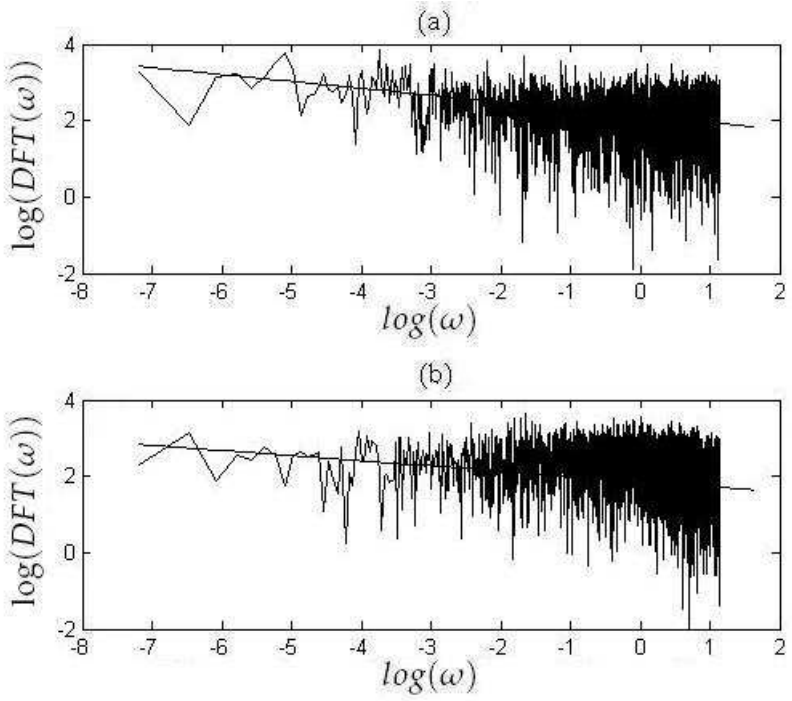

Figure 8.5: Illustration of the log-DFT function plotted for $y_2(t)$, for a 38° overtwisted DNA, obtained using (a) AMBER model and (b) SDE model.

Moreover, the results from Table 8.3 show that β decreases as we move further away from the defect site, which is due to the reduced influence of breathing on the other base-pairs.

Base-pair	β_{AMBER}	β_{SDE}
$y_1(t)$	0.225	0.210
$y_2(t)$	0.180	0.135
$y_3(t)$	0.180	0.130
$y_4(t)$	0.180	0.130

Table 8.3: Values of the gradient β of the log-log representation of $DFT(y_0)$ in terms of ω, for 10 ns of data, with information about each 1 ps obtained using the AMBER and SDE models, respectively.

Observe that for the mesoscopic model we have a slightly reduction in β's

value, but this behaviour is preserved. However, these values were obtained by analysing the long 10 ns simulations with data for each 1 ps. Analysing the short 2 ns simulations, but with information about each 2 fs, we observe that β doubles, as can be seen in Table 8.4.

Base-pair	β_{AMBER}	β_{SDE}
$y_1(t)$	0.450	0.445
$y_2(t)$	0.425	0.205
$y_3(t)$	0.410	0.180
$y_4(t)$	0.390	0.155

Table 8.4: Values of the gradient β of the log-DFT function, for data obtained using AMBER data over 2 ns, with information about each 2 fs.

Note again that β decreases as we move to the end of the DNA sequence, but for the SDE system the decrease happens much faster than in the case of AMBER data. We use in our model short range forces to describe the along-chain interactions, while these results suggests that it might be more appropriate to define long range interactions instead. Moreover, these results confirm once again that decreasing the timestep Δt takes β's value closer to 1 and hence, it is easier to analyse the self-organised DNA characteristics.

8.3.2 Self-organised behaviour in DNA

The DNA simulations discussed in Chapter 5 revealed that a DNA sequence spends an important proportion of its simulation time in the closed state and the rest of time is represented by two or three open states depending on the degree of DNA twist. Hence, we can model a DNA molecule using a cellular automaton in which the transitions are defined between the closed and open states. The breathing states can be considered the critical states of our system and act as attractors, given that they emerge after a period of time. Indeed, in Chapter 4 we ignore the first 5 or 6 ns of AMBER simulations, since the data shows an unrepresentative initial transient. In reality, these 5 ns represent just a transition period from the initial conditions to the critical states – see Figure 4.1

for more details. Hence, our DNA sequence exhibits the slowly-driven non-equilibrium behaviour characteristic of self-organised systems.

The frequency-based analysis reveals that the DFT computed at the defect site has the form of a power-law. Moreover, Figures 8.1 and 8.2 suggest the presence of flicker noise in our data, for both AMBER and SDE simulations. Recall that in Chapter 3 we mentioned that even with the wrong values for our system parameters, very short breathing events were obtained; this confirms the DNA scale-invariance property. In other words, small changes in SDE parameter values affect details of breathing statistics, but not its self-organised structure. Moreover, in our SDE system, random oscillations were modelled using white noise. However, the breathing pairs DFTs show the presence of pink noise in our system.

One might think that the defect site is the cause of the SOC DNA behaviour. We replaced a thymine (T) base with the difluorotoluene (F) base to weaken the inter-chain interactions and allow breathing to occur on the nanosecond timescale compared to a normal DNA sequence in which breathing occurs on the microsecond timescale. But, this change does not affect the DNA structure or behaviour, as discussed in several papers, such as [52], for example. In addition, the DFTs charactherisic for the trajectory of the nonbreathing pairs in our system also suggest the presence of flicker noise among AMBER and SDE data, even though the values of β from Table 8.4 are reduced to half compared to the A-F base-pair values presented in Table 8.2.

All these results obtained using a frequency-based analysis imply that breathing is not generated by a particular mode. In most papers studying DNA bubbles or waves, the specific modes are determined either analytically or numerically and the system is characterised based on the results obtained. Our results show that this strategy should be revised, since it might be the wrong approach for such studies. Recall also that in Chapter 5 we obtained that the AMBER and SDE simulations are random, which also suggests that DNA might be considered a self-organised system.

Finally, we conclude that, given these aspects, DNA is a self-organised system, since AMBER data shows SOC property. In addition, although the reduced SDE model that we propose cannot fully elucidate breathing causes, it is able to

predict with accuracy DNA behaviour, including SOC. However, studying the SOC properties of DNA might offer some information about bubble nucleation and growth via travelling waves, as well as about widely studied events, such as DNA transcription and replication.

8.4 Summary

In scientific literature several cellular automaton models have been created to study the self-organised behaviour characteristic to different systems, such as sandpiles, earthquakes or forest-fires, for example. These models connect critical phenomena to features like power laws, fractals, and flicker noise. We have analysed the Fourier transform of the AMBER and SDE data and we have reached the conclusion that it scales into a power law emphasizing $1/f$ noise in our system. Hence, we conclude that DNA exhibits a self-organised behaviour, as many other complex systems.

CHAPTER 9

Conclusions

In this thesis we have studied the dynamics of a 12-mer DNA duplex, for which a thymine base (T) was replaced with the 'rogue' base difluorotoluene (F) so as to obtain breathing on the nanosecond time-scale instead of the microsecond time-scale, as obtained in all-atom simulations of a nondefective DNA molecule. The time spent simulating such systems, using MD programs, is large due to the solvent presence, which in our case is water. For a 20 nanosecond simulation, for example, we need about 2 weeks and 4 processors working in parallel, as well as about 8000 gigabytes to store the information. However, a simplified model can also be used to study with accuracy the DNA properties by using less resources.

The DNA sequence was analysed for twist angles in the range 30°-40° per base-pair, which revealed that the length and frequency of the breathing events varies with twist angle. We decided to develop a model based on a system of stochastic ordinary differential equations, which might explain this twist dependence and also reduce the simulation time, as discussed above.

Adding noise and damping to a nonlinear Klein-Gordon lattice model, we obtain a new mesoscopic model of the DNA duplex, with a defect at the middle site of the lattice. Previously, it has been thought that breathing events were caused by inhomogeneities in the *inter-strand* interactions. However, our results show that there is, in addition, a significant change in the *along-chain* interactions, which contributes to the breathing. Thus, we consider the defect in both along-chain and inter-strand interactions. The system parameters were

fitted to AMBER data using the maximum likelihood method. The fitting process revealed several interesting features of our system. First of all, the noise and damping coefficients are related to the system's temperature through the classic fluctuation-dissipation relation from (3.4.10). However, our N equations of motion are obtained using a change of variables form $2N$ other equations. In such transformations the damping coefficient is left invariant, but the noise terms are added or subtracted, changing the fluctuation-dissipation relationship. Moreover, we reduce each base (containing on average 32 atoms) to one point mass. This simplification requires the use of an alternative fluctuation-dissipation relation (the one defined in (3.4.12)), which takes all these aspects into consideration, as well as the solvent interactions with the DNA atoms.

Next, the MLE method, which we use to derive parameter values of reduced model from AMBER data, is sensitive to input data. We first tried to obtained the parameter values using AMBER data over 20 ns, with information about each 1 ps. We were able to obtain only rare and short breathing events, even when MLE with a penalty term or smooth splines representation of $E_0(y_0)$ (see (3.2.14) for definition) were used. However, the timestep used in AMBER simulation is $\Delta t = 2$ fs. Due to the large storage capacity required to store information each 2 fs, we selected for each twist angle a 2 ns simulation representative of our data in terms of the breathing length and frequency. Applying MLE to these datasets we obtained improved parameter values for our system, which allowed us to simulate breathing events with good accuracy.

Our analysis of parameter values revealed that, for an undertwisted DNA sequence, the along-chain bonds become weaker and the inter-chain bonds become stronger, as the twist angle is increased from 30° to 35°. At 36° we have the weakest and strongest, respectively, of the two types of interactions. As DNA is overtwisted, this behaviour is reversed, that is, the along-chain bonds decrease with twist angle, while the inter-strand bonds become stronger as we approach 40°. For the noise coefficient of the nonbreathing base-pairs we only observe small fluctuations, whilst for the variation of the noise coefficient specific to the A-F pair we observe a dependence on the time spent breathing – see Figure 4.4 and Table 4.1.

The height of the breathing barrier ΔB and the energy difference between open

and closed states ΔE, that characterise $E_0(y_0)$ – see Table 4.8 for details – are responsible of the breathing frequency and length, respectively. However, breathing can be considered as a competition between the along-chain elastic energy, the inter-chain binding energy and the system's entropy term, which in our case is the damping term. Hence, the the length and frequency of breathing events is given by the potential of mean force, whose expression can be approximated via (3.3.22). The variation of breathing is interesting: at 34°-35° breathing events are relatively rare, whilst for undertwisted DNA plasmids we observe an increase in the breathing frequency, due to a reduction in the energy difference ΔE (see Table 4.8). For overtwisted plasmids there is again a reduction in ΔE, but also a decrease in ΔB and an increase in the fluctuation-dissipation parameter C (see Table 4.5), hence less damping. This leads to a larger residence time in the breathing state, thus longer breathing events.

Next, we compare the SDE simulations to data obtained from AMBER and we observe that the DNA behaviour predicted by our mesoscopic model is close to that observed in experiments and all-atom MD simulations. This underlines the capability of the SDE system to simulate breathing events with reasonably good accuracy. We classify breathing by the mean μ and standard deviation σ of the time spent in the closed state by our DNA molecule between two breathing events: if $\mu \approx \sigma$ we refer to it as 'random', while for $\mu > \sigma$ and $\mu < \sigma$ we refer to it as 'regular' and 'clustered', respectively. The long timescale analysis reveals that SDE simulations are more regular than the AMBER simulations. However, the statistical tests show that both AMBER and SDE simulations are 'random'. In addition, a *slight* reduction in the amplitude of fluctuations in the reduced SDE model is observed, when compared with AMBER data. This is due to the massively reduced number of degrees of freedom in our SDE system. Also, the analysis of long time dynamics using the SDE system revealed the increase with time of the percentage of breathing in a simulation.

We also used traditional methods to compare the simulations obtained using the two methods. Principal component analysis (PCA) is a tool that allows one to filter the noise from data and determine those directions with high data variances. The AMBER and SDE datasets have similar properties in terms of principal components, and PCA also confirms the small difference in the degree of randomness between the SDE and AMBER simulations. Next, comput-

ing the data autocorrelation function we observed that the data is correlated with the system's initial conditions for about 0.5 ns in the SDE system and for at most 1 ns in the AMBER system. Finally, we tried to determine the normal modes vectors and their corresponding frequencies based on the discrete Fourier transform. However, in both cases, rather than exhibiting a few spikes corresponding to collective oscillations, the Fourier transform exhibits a power law behavior, suggesting that DNA might be a self-organised system.

Analysing in detail the log-log representation of the DFTs for AMBER, we note the presence of $1/f$ noise in our system. Even though we introduced white noise in our SDE system as a random forcing term, we end up with *pink* noise as the output $y_0(t)$, since the DFTs for the A-F base pair are proportional to $1/\omega^{\beta}$, where ω is the frequency and β has values close to, but bellow 1, as presented in Table 8.2. Analysing the DFTs for the nonbreathing pairs of our DNA molecule, we observe a decrease in β to an average value of 0.4. In general, when $0 < \beta < 2$ it is considered that $1/f$ noise is present into our system. This confirms the self-organised DNA behaviour and also reduces the doubts that the defect site might actually be the cause of the SOC features observed.

Thus, the proposed SDE model is capable not only of predicting the DNA behaviour, but it also preserves most of the system properties, such as self-organisation. The importance of the fluctuation-dissipation relation in reduced models is also discussed by considering both deterministic and random forces in our system, in which the energy is conserved on the long timescale via a balance between damping and stochastic forcing terms. We conclude that our mesoscopic model allows us to study breathing events in detail and, in addition, it is also useful to analyse how the along-chain and inter-chain interactions vary with helical twist. Finally, the SDE model is helpful in illustrating the self-organised behaviour of DNA.

In conclusion, many complex systems, such as DNA, need to be analysed in detail in order to determine all their hidden features. A reduced model sometimes uncovers some of the system's properties, but, as in our case, might not give a clear explanation about their origins. However, the answers we are looking for might be found by studying in more detail the SOC properties of DNA.

Given that the SDE simulations are close to the AMBER results and that the

DNA properties are preserved by the reduced system, the work presented in this thesis can be extended by analysing long SDE simulations of hundreds of nanoseconds. Based on these simulations we could analyse in more detail other characteristics of the breathing events, as well as their length and frequency. For example, we could analyse the number of breathing events having the length greater that a given length L_0 and investigate how this varies with L_0. Another quantity worth analysing is the distribution of the time intervals between breathing events. In this way, we hope to obtain a power law form of the corresponding functions, which might further confirm self-organising behaviour in the DNA molecule. On the other hand, the analysis might show that at larger scales the self-organisation properties cease. Note that we require long simulation, since the datasets that we have analysed contain only tens of breathing events, which is insufficient for an accurate analysis of these distributions.

It would also be interesting to analyse bubble length and their preferred formation sites in a DNA sequence. Note that our model cannot be used to perform such a task and one should take care when trying to construct a new model, since a bubble supposes several defect sites and the along-chain interactions between two such sites might differ to the ones we determined. In addition, the inter-chain potentials might also be different and we might expect to have larger damping coefficients to preserve the total energy in our system.

Finally, note that our SDE model contains an important restriction, that is, all base-pairs in our DNA sequence are characterised by the same twist angle. Over a short DNA molecule of only twelve base-pairs, this should be a reasonably accurate approximation. However, it will be less accurate over a DNA duplex of hundreds of bases. One can improve the mesoscopic model by considering an extra degree of freedom for each base-pair, representing the local twist angle. In addition, considering longer range interactions in our system might also improve the model as suggested in Chapter 8. In Section 7.1.1 we discuss the difficulties of using an approach based on a PCA-based predictive models. However, similar techniques, which allow one to predict the behaviour of all atoms in a DNA molecule, might also be useful. Note that we could reduce the complexity in all these models by incorporating the solvent effect through some parameters and by ignoring the water molecules when we simulate the system using the new approach. Moreover, increasing the number of

degrees of freedom means more accurate results. In conclusion, such general models might take us closer to the real DNA behaviour.

Part III

Appendix and References

APPENDIX A

Details of Amber Simulations

In what follows, we describe the files needed to simulate the DNA sequence using AMBER (as described in Chapter 2), that is, topology and coordinates files, SANDER input files, as well as *pdb* files.

A.1 Amber topology files

AMBER topology files contain information about atom types and several flag values, as defined in the topology file specific to the 30° twist angle:

```
%VERSION  VERSION_STAMP = V0001.000
%DATE = 04/23/07  15:16:53
%FLAG TITLE
%FORMAT(20a4)

%FLAG POINTERS
%FORMAT(10I8)
16682      20  .....    0

%FLAG ATOM_NAME
%FORMAT(20a4)
H5T O5' C5' ... (bases atoms) ... H2'2O3' H3T Na+ ...
... Na+ O   H1  H2 ... O  H1  H2
```

```
%FLAG CHARGE
%FORMAT(5E16.8)
8.05790106E+00 -1.15128491E+01 ...  7.59869910E+00

%FLAG MASS
%FORMAT(5E16.8)
1.00800000E+00  1.60000000E+01 ...  1.00800000E+00

%FLAG ATOM_TYPE_INDEX
%FORMAT(10I8)
1       2       3       4       4 ...       1       1

%FLAG NONBONDED_PARM_INDEX
%FORMAT(10I8)
1       2       4       7      11 ...      -1     210

%FLAG RESIDUE_LABEL
%FORMAT(20a4)
DC5 DT  DT  DT  DT  DG  F   DA  DT  DC  DT  DT3 DA5
DA  DG  DA  DT  DA  DC  DA  DA  DA  DA  DG3 Na+ ...
... Na+ WAT ... WAT

%FLAG RESIDUE_POINTER
%FORMAT(10I8)
1      29      61      93     125 ...   16677   16680

%FLAG BOND_FORCE_CONSTANT
%FORMAT(5E16.8)
2.30000000E+02  3.40000000E+02 ...  5.53000000E+02

%FLAG BOND_EQUIL_VALUE
%FORMAT(5E16.8)
1.61000000E+00  1.09000000E+00 ...  1.51360000E+00

%FLAG ANGLE_FORCE_CONSTANT
%FORMAT(5E16.8)
```

```
1.00000000E+02  4.50000000E+01 ...  5.50000000E+01

%FLAG ANGLE_EQUIL_VALUE
%FORMAT(5E16.8)
1.88897066E+00  1.79070858E+00 ...  1.89368305E+00

%FLAG DIHEDRAL_FORCE_CONSTANT
%FORMAT(5E16.8)
1.85181000E-01  1.25653100E+00 ...  1.10000000E+00

%FLAG DIHEDRAL_PERIODICITY
%FORMAT(5E16.8)
1.00000000E+00  2.00000000E+00 ...  2.00000000E+00

%FLAG DIHEDRAL_PHASE
%FORMAT(5E16.8)
5.54929070E-01  6.14285649E+00 ...  3.14159400E+00

%FLAG SOLTY
%FORMAT(5E16.8)
0.00000000E+00  0.00000000E+00 ...  0.00000000E+00

%FLAG LENNARD_JONES_ACOEF
%FORMAT(5E16.8)
0.00000000E+00  0.00000000E+00  5.81803229E+05 ...
... 0.00000000E+00  0.00000000E+00  0.00000000E+00

%FLAG LENNARD_JONES_BCOEF
%FORMAT(5E16.8)
0.00000000E+00  0.00000000E+00  6.99746810E+02 ...
... 0.00000000E+00  0.00000000E+00  0.00000000E+00

%FLAG BONDS_INC_HYDROGEN
%FORMAT(10I8)
72      75       2      72      78 ...      2196       1

%FLAG ANGLES_INC_HYDROGEN
```

```
%FORMAT(10I8)
75      72      78       3      69 ...     -2229      49

%FLAG DIHEDRALS_WITHOUT_HYDROGEN
%FORMAT(10I8)
81      84      93      96       1 ...     16682       0

%FLAG HBOND_ACOEF
%FORMAT(5E16.8)
0.00000000E+00

%FLAG HBOND_BCOEF
%FORMAT(5E16.8)
0.00000000E+00

%FLAG HBCUT
%FORMAT(5E16.8)
0.00000000E+00

%FLAG AMBER_ATOM_TYPE
%FORMAT(20a4)
HO  OH  CI  H1  H1  CT  ...  OW  HW  HW  OW  HW  HW

%FLAG TREE_CHAIN_CLASSIFICATION
%FORMAT(20a4)
M   M   M   E   E   M  ...  BLA  BLA  BLA

%FLAG JOIN_ARRAY
%FORMAT(10I8)
  0       0       0       0       0 ...      0       0

%FLAG IROTAT
%FORMAT(10I8)
  0       0       0       0       0 ...      0       0

%FLAG SOLVENT_POINTERS
%FORMAT(3I8)
```

```
46    5323      25

%FLAG ATOMS_PER_MOLECULE
%FORMAT(10I8)
380     383       1       1       1 ...      3       3

%FLAG BOX_DIMENSIONS
%FORMAT(5E16.8)
1.09471219E+02  6.30285105E+01 ...  6.30285105E+01

%FLAG RADIUS_SET
%FORMAT(1a80)
modified Bondi radii (mbondi)

%FLAG RADII
%FORMAT(5E16.8)
8.00000000E-01  1.50000000E+00 ...  8.00000000E-01

%FLAG SCREEN
%FORMAT(5E16.8)
8.50000000E-01  8.50000000E-01 ...  8.50000000E-01
```

Comparing with the *pdb* file from Appendix A.4, we observe that the information from the two files agree.

A.2 Amber coordinates files

The coordinates files contain information about the initial position each atom in the three-dimensional space. The first line indicates the number of atoms described. The coordinates file specific to the 30° twist angle is defined as follows:

```
16682
26.2037475  46.5614036  46.8884444
25.7511594  45.0593591  47.1714946
```

```
25.4457442  44.6675612  48.5339170
24.3743302  44.7683058  48.7113050

.....

26.3507133  36.4888215  -1.4859173
30.8987473  32.8987150   2.0372416
31.7618935  32.5423955   2.2475740
30.3252884  32.5601045   2.7247820
```

Comparing this file with the one from Appendix A.4, we observe that the coordinates are the same with the ones described in the *pbd* file. This information is not redundant as one may think, since the *pbd* file is not involved in the simulation process.

A.3 SANDER input files

SANDER input files contain one or several namelists and control variables that determine the type of simulations to be processed. An example of an energy minimization input file is the following:

```
Minimization of water atoms
&cntrl
      imin=1, maxcyc=5000, ncyc=50,
      drms=0.5, ibelly=1, ntb=1x
&end
Residues that are going to move in the minimization
RES 25 534
END
END
```

The MD simulation files specify more or less the same parameters:

```
100ps MD with cartesian restrain on
the four terminal bases atoms only
&cntrl
      irest=1, ntx=7, ntf=2,
      ntb=2, scee=1.2, cut=9.0,
      ntr=1, nstlim=500000, dt=0.002,
      ntwx=500, ntwe=500, ntwv=500
      ntp=1, ntc=2,
&end
Cartesian restrain on the four terminal bases
10.0
ATOM 11 22 360 373 391 404 742 756
END
END
```

A.4 Amber *pdb* files

The *pbd* files obtained after creating the DNA sequence to be analysed contain on each line a description of the system atoms, except the lines containing reserved words, such as `REMARK`, `TER`, and `END`, and eight corresponding columns describing:

1. the type of the residue analysed (`ATOM` in our case)
2. the unique identification number of each atom
3. atom type
4. type of the base containing the atom
5. the residue (in our case is a base) number containing the atom
6. the position of each atom in the three-dimensional space

For a 30° twist angle, the *pdb* file contains the following information:

```
REMARK
ATOM      1  H5T DC5     1      26.204  46.561  46.888

.....

(continue defining C base atoms)

.....

ATOM     28  O3* DC5     1      27.725  41.647  48.695
ATOM     29  P   DT      2      28.764  41.460  47.383

.....

(continue defining T base atoms)

.....

ATOM     60  O3* DT      2      28.076  36.066  47.793
ATOM     61  P   DT      3      29.327  35.799  46.700

.....

(continue defining T base atoms)

.....

ATOM     92  O3* DT      3      26.878  31.146  45.258
ATOM     93  P   DT      4      28.234  30.675  44.380

.....

(continue defining T base atoms)

.....
```

```
ATOM    124  O3* DT      4      24.947  27.782  41.131
ATOM    125  P   DT      5      26.275  27.039  40.409

.....

(continue defining T base atoms)

.....

ATOM    156  O3* DT      5      23.298  26.455  35.878
ATOM    157  P   DG      6      24.467  25.445  35.210

.....

(continue defining G base atoms)

.....

ATOM    189  O3* DG      6      22.864  27.101  30.269
ATOM    190  P   F       7      23.792  25.902  29.539
ATOM    191  O1P F       7      22.690  25.456  28.663
ATOM    192  O2P F       7      24.807  24.872  29.875
ATOM    193  O5* F       7      24.515  27.172  28.904
ATOM    194  C5* F       7      23.800  28.432  28.860
ATOM    195 1H5* F       7      24.053  29.027  29.739
ATOM    196 2H5* F       7      22.726  28.245  28.847
ATOM    197  C4* F       7      24.186  29.196  27.615
ATOM    198  H4* F       7      23.473  30.005  27.457
ATOM    199  O4* F       7      25.460  29.873  27.857
ATOM    200  C1* F       7      26.495  29.212  27.155
ATOM    201  H1* F       7      26.876  30.062  26.589
ATOM    202  C1  F       7      27.580  28.907  28.125
ATOM    203  C6  F       7      27.504  27.802  28.937
ATOM    204  H6  F       7      26.607  27.166  28.898
ATOM    205  C5  F       7      28.487  27.517  29.813
ATOM    206  C5M F       7      28.451  26.328  30.726
ATOM    207 1H5M F       7      27.407  25.935  30.830
```

```
ATOM    208 2H5M F       7      28.823  26.607  31.745
ATOM    209 3H5M F       7      29.100  25.510  30.321
ATOM    210  C4  F       7      29.643  28.372  29.932
ATOM    211  F4  F       7      30.585  28.187  30.712
ATOM    212  C3  F       7      29.633  29.454  29.083
ATOM    213  H3  F       7      30.403  30.110  29.091
ATOM    214  C2  F       7      28.651  29.775  28.168
ATOM    215  F2  F       7      28.730  30.758  27.453
ATOM    216  C3* F       7      24.434  28.365  26.359
ATOM    217  H3* F       7      23.821  27.464  26.314
ATOM    218  C2* F       7      25.896  27.976  26.494
ATOM    219 1H2* F       7      26.184  27.342  25.654
ATOM    220 2H2* F       7      26.045  27.433  27.427
ATOM    221  O3* F       7      24.259  29.125  25.166
ATOM    222  P   DA      8      24.924  27.861  24.274

.....

(continue defining A base atoms)

.....

ATOM    253  O3* DA      8      27.600  31.564  21.299
ATOM    254  P   DT      9      28.054  30.381  20.191

.....

(continue defining T base atoms)

.....

ATOM    285  O3* DT      9      32.491  33.343  19.066
ATOM    286  P   DC     10      32.840  32.364  17.742

.....

(continue defining C base atoms)
```

```
.....

ATOM    315  O3* DC     10      38.115  33.567  18.425
ATOM    316  P   DT     11      38.492  32.858  16.945

.....

(continue defining T base atoms)

.....

ATOM    347  O3* DT     11      43.458  31.752  18.910
ATOM    348  P   DT3    12      43.993  31.311  17.376

.....

(continue defining T base atoms)

.....

ATOM    380  H3T DT3    12      48.361  27.715  18.279
TER
ATOM    381  H5T DA5    13      36.760  14.588  16.857

.....

(continue defining A base atoms)

.....

ATOM    410  O3* DA5    13      39.637  14.919  21.477
ATOM    411  P   DA     14      38.275  15.442  22.317

.....

(continue defining A base atoms)
```

```
.....

ATOM    442  O3* DA     14      41.584  18.106  25.735
ATOM    443  P   DG     15      40.301  18.407  26.783

.....

(continue defining G base atoms)

.....

ATOM    475  O3* DG     15      42.890  22.915  28.427
ATOM    476  P   DA     16      41.801  23.107  29.697

.....

(continue defining A base atoms)

.....

ATOM    507  O3* DA     16      42.708  28.478  29.471
ATOM    508  P   DT     17      41.878  28.705  30.917

.....

(continue defining T base atoms)

.....

ATOM    539  O3* DT     17      40.594  33.727  29.225
ATOM    540  P   DA     18      40.015  34.120  30.756
ATOM    541  O1P DA     18      40.758  35.395  30.805
ATOM    542  O2P DA     18      39.716  33.492  32.065
ATOM    543  O5* DA     18      38.659  34.311  29.936
ATOM    544  C5* DA     18      38.734  34.535  28.507
ATOM    545 1H5* DA     18      38.634  33.583  27.982
```

```
ATOM    546 2H5* DA     18      39.693  34.989  28.255
ATOM    547  C4* DA     18      37.619  35.459  28.075
ATOM    548  H4* DA     18      37.825  35.821  27.069
ATOM    549  O4* DA     18      36.390  34.676  27.938
ATOM    550  C1* DA     18      35.519  34.961  29.015
ATOM    551  H1* DA     18      34.565  35.381  28.697
ATOM    552  N9  DA     18      35.124  33.666  29.634
ATOM    553  C8  DA     18      35.815  32.917  30.548
ATOM    554  H8  DA     18      36.752  33.276  30.949
ATOM    555  N7  DA     18      35.204  31.825  30.895
ATOM    556  C5  DA     18      34.026  31.862  30.171
ATOM    557  C6  DA     18      32.930  30.976  30.091
ATOM    558  N6  DA     18      32.856  29.836  30.799
ATOM    559 1H6  DA     18      32.046  29.242  30.707
ATOM    560 2H6  DA     18      33.611  29.578  31.426
ATOM    561  N1  DA     18      31.926  31.308  29.267
ATOM    562  C2  DA     18      32.010  32.439  28.573
ATOM    563  H2  DA     18      31.206  32.721  27.909
ATOM    564  N3  DA     18      32.971  33.339  28.553
ATOM    565  C4  DA     18      33.962  32.978  29.388
ATOM    566  C3* DA     18      37.248  36.576  29.048
ATOM    567  H3* DA     18      38.097  36.931  29.634
ATOM    568  C2* DA     18      36.244  35.902  29.969
ATOM    569 1H2* DA     18      36.648  35.860  30.982
ATOM    570 2H2* DA     18      35.314  36.471  29.973
ATOM    571  O3* DA     18      36.620  37.675  28.393
ATOM    572  P   DC     19      36.218  38.325  29.893

.....

(continue defining C base atoms)

.....

ATOM    601  O3* DC     19      31.353  39.686  27.841
ATOM    602  P   DA     20      31.010  40.613  29.202
```

```
.....

(continue defining A base atoms)

.....

ATOM    633  O3* DA     20      25.713  39.639  28.351
ATOM    634  P   DA     21      25.292  40.789  29.506

.....

(continue defining A base atoms)

.....

ATOM    665  O3* DA     21      20.715  37.971  30.426
ATOM    666  P   DA     22      20.100  39.230  31.359

.....

(continue defining A base atoms)

.....

ATOM    697  O3* DA     22      17.203  35.549  34.151
ATOM    698  P   DA     23      16.329  36.774  34.908

.....

(continue defining A base atoms)

.....

ATOM    729  O3* DA     23      15.625  33.443  39.166
ATOM    730  P   DG3    24      14.499  34.498  39.837

.....
```

```
(continue defining G base atoms)

.....

ATOM    763  H3T DG3    24      14.603  33.436  45.466
TER
ATOM    764 Na+  Na+    25      34.357  33.630  24.034
TER

.....

(continue defining Na+ molecules)

.....

TER
ATOM    785 Na+  Na+    46      44.059  18.914  31.800
TER
ATOM    786  O   WAT    47      27.312  29.344  63.734
ATOM    787  H1  WAT    47      26.661  29.929  63.347
ATOM    788  H2  WAT    47      28.107  29.875  63.793
TER

.....

(continue defining water molecules)

.....

TER
ATOM  16680  O   WAT  5345      30.899  32.899   2.037
ATOM  16681  H1  WAT  5345      31.762  32.542   2.248
ATOM  16682  H2  WAT  5345      30.325  32.560   2.725
TER
END
```

APPENDIX B

Data plots for the full range of twist angles

In what follows, we present the data autocorrelation functions, the Fourier transforms and their log-representation for the twist angles not shown in Part II.

B.1 Data autocorrelation figures

Figures B.1-B.5 present a comparison between the autocorrelation functions for AMBER and SDE data, specific to the A-F base-pair. A detailed discussion on this comparison is made in Section 7.2.

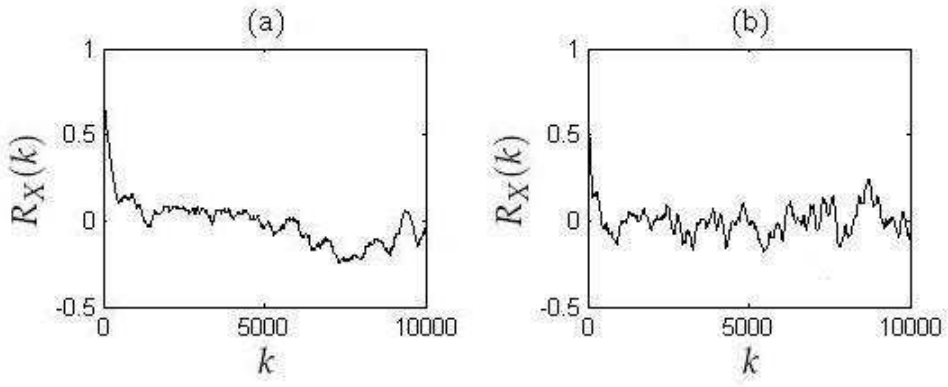

Figure B.1: Illustration of autocorrelation function, for a 32° undertwisted DNA, obtained using (a) AMBER model and (b) SDE model.

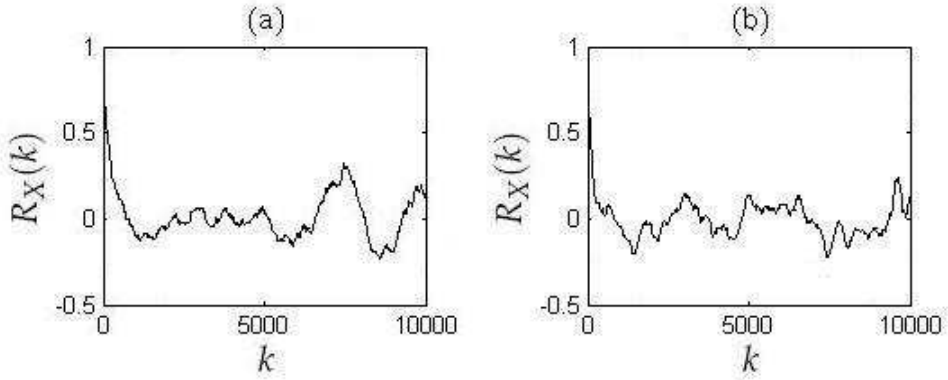

Figure B.2: Illustration of autocorrelation function, for a 33° undertwisted DNA, obtained using (a) AMBER model and (b) SDE model.

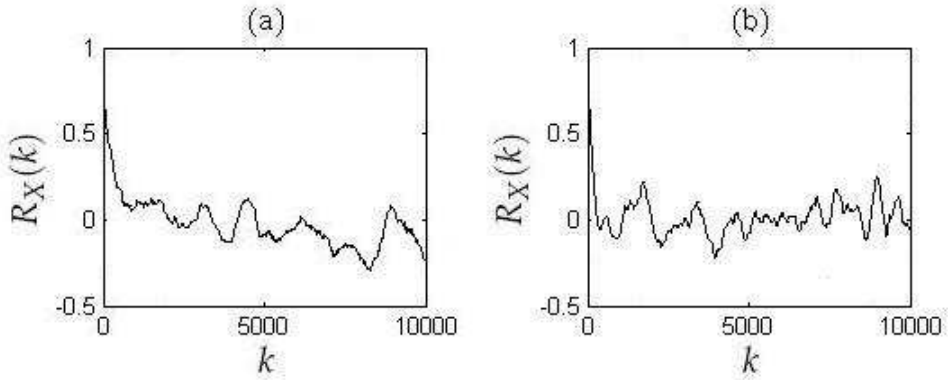

Figure B.3: Illustration of autocorrelation function, for a 34° undertwisted DNA, obtained using (a) AMBER model and (b) SDE model.

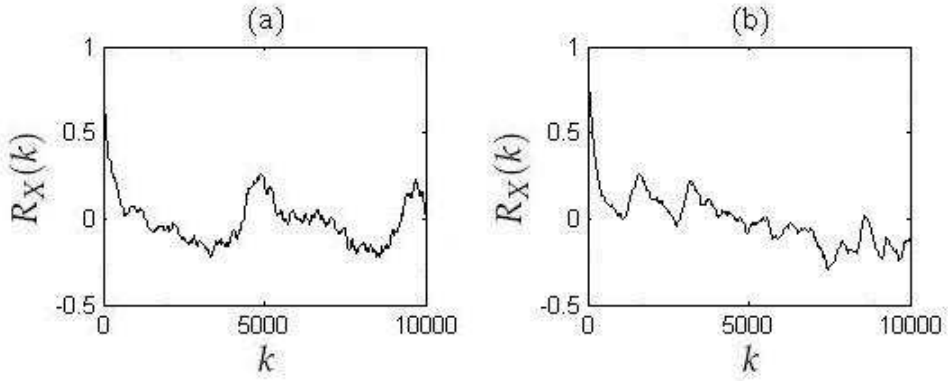

Figure B.4: Illustration of autocorrelation function, for a 36° twisted DNA, obtained using (a) AMBER model and (b) SDE model.

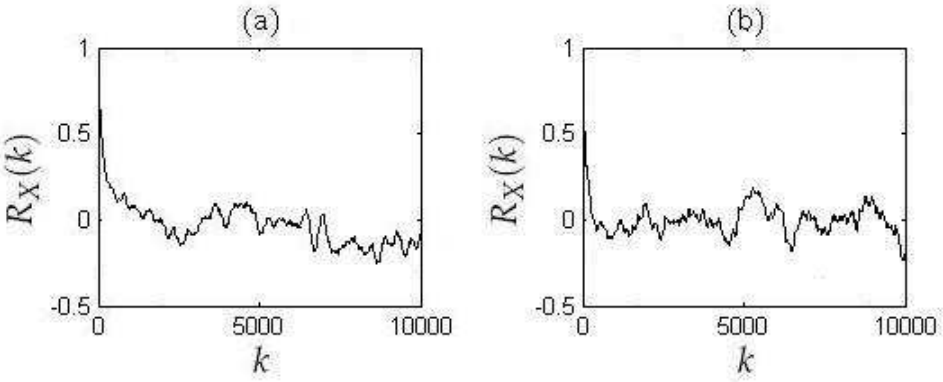

Figure B.5: Illustration of autocorrelation function, for a 40° overtwisted DNA, obtained using (a) AMBER model and (b) SDE model.

B.2 Discrete Fourier Transform figures

Figures B.6-B.12 present a comparison between the DFT of AMBER and SDE data, specific to the A-F base-pair, discussed in Section 7.3.

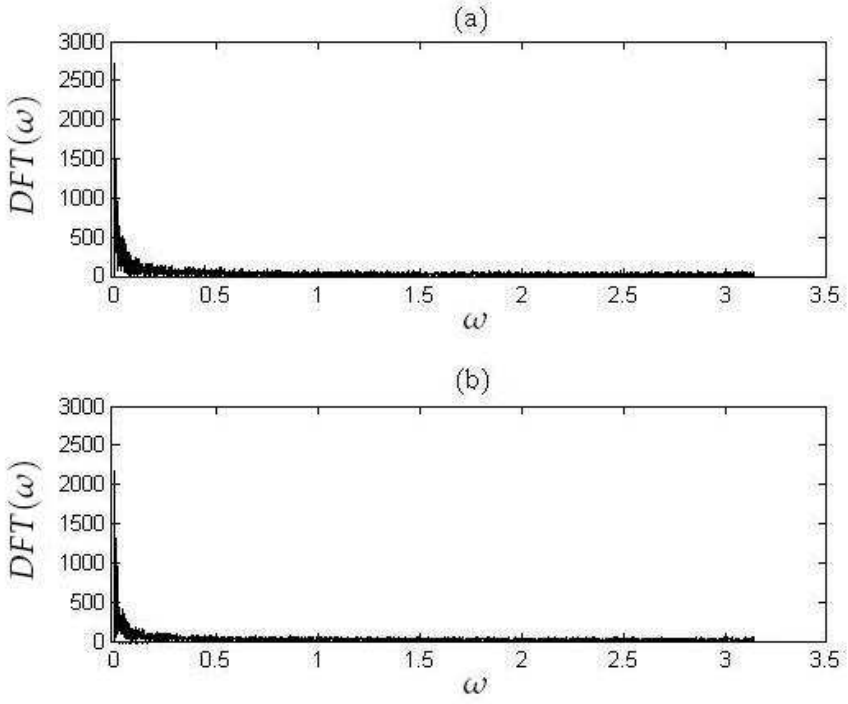

Figure B.6: Illustration of the DFT, for a 30° undertwisted DNA, obtained using (a) AMBER model and (b) SDE model.

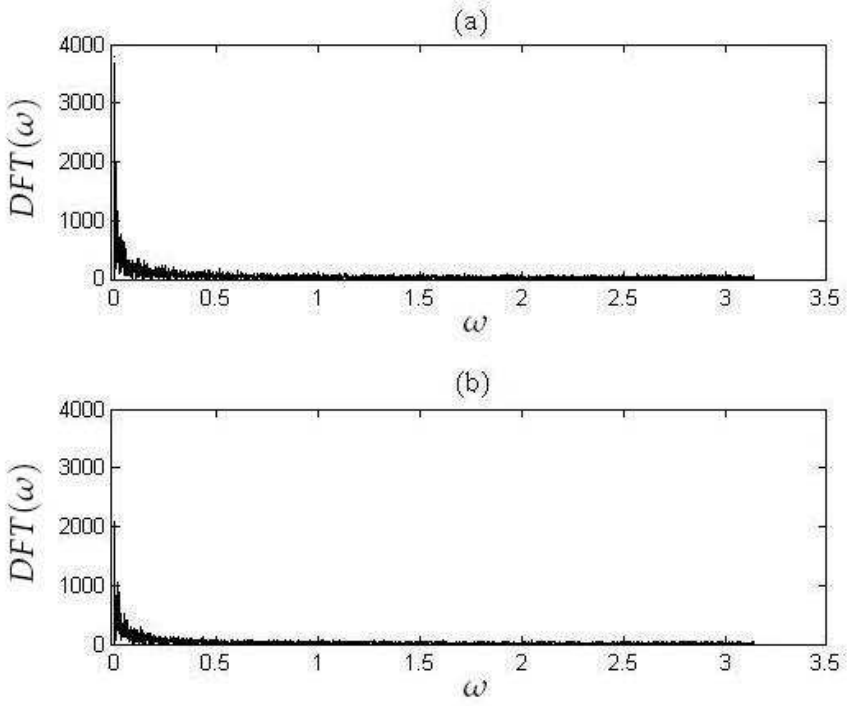

Figure B.7: Illustration of the DFT, for a 32° undertwisted DNA, obtained using (a) AMBER model and (b) SDE model.

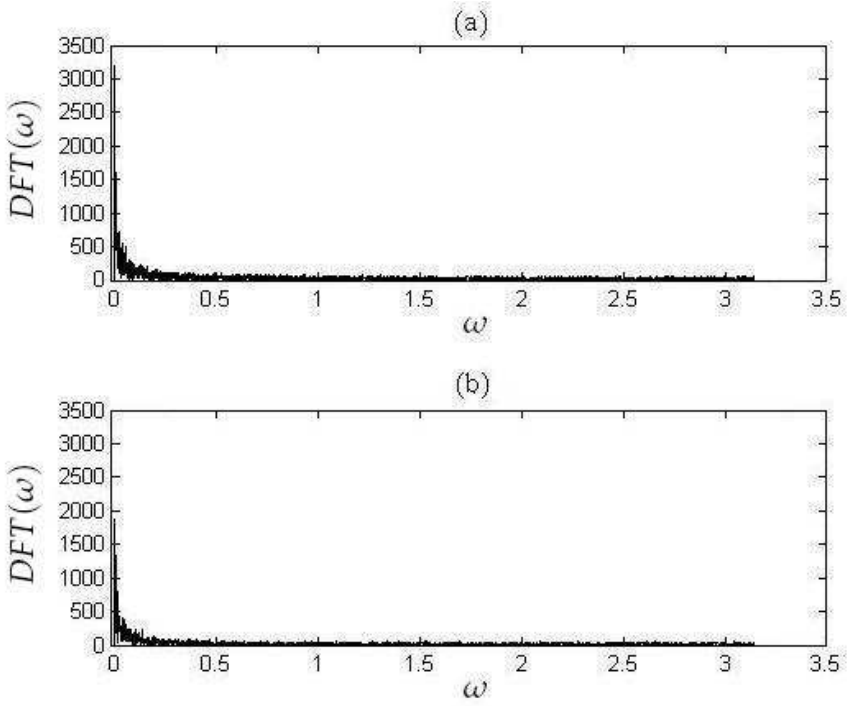

Figure B.8: Illustration of the DFT, for a 33° undertwisted DNA, obtained using (a) AMBER model and (b) SDE model.

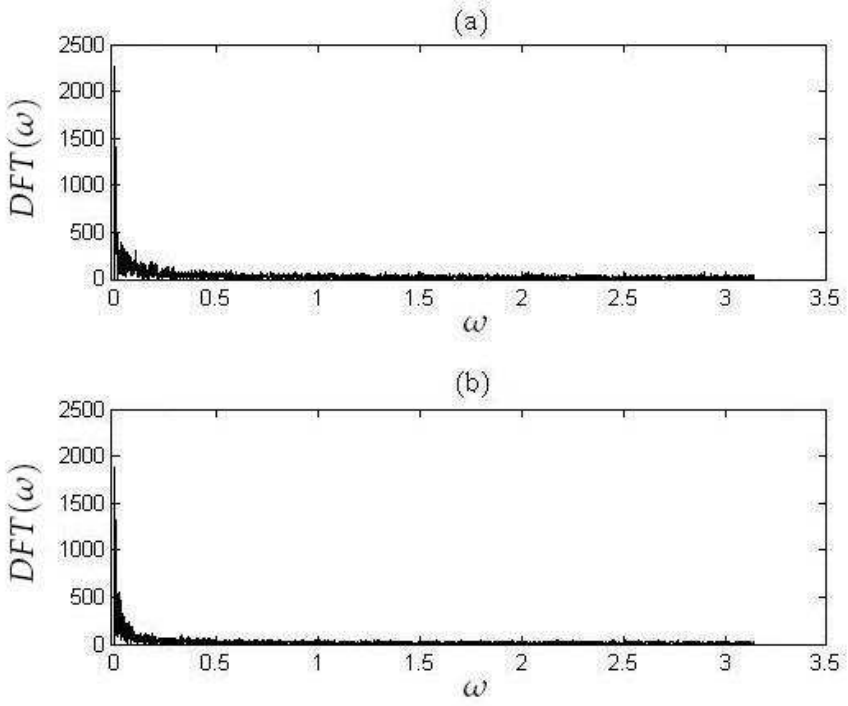

Figure B.9: Illustration of the DFT, for a 34° undertwisted DNA, obtained using (a) AMBER model and (b) SDE model.

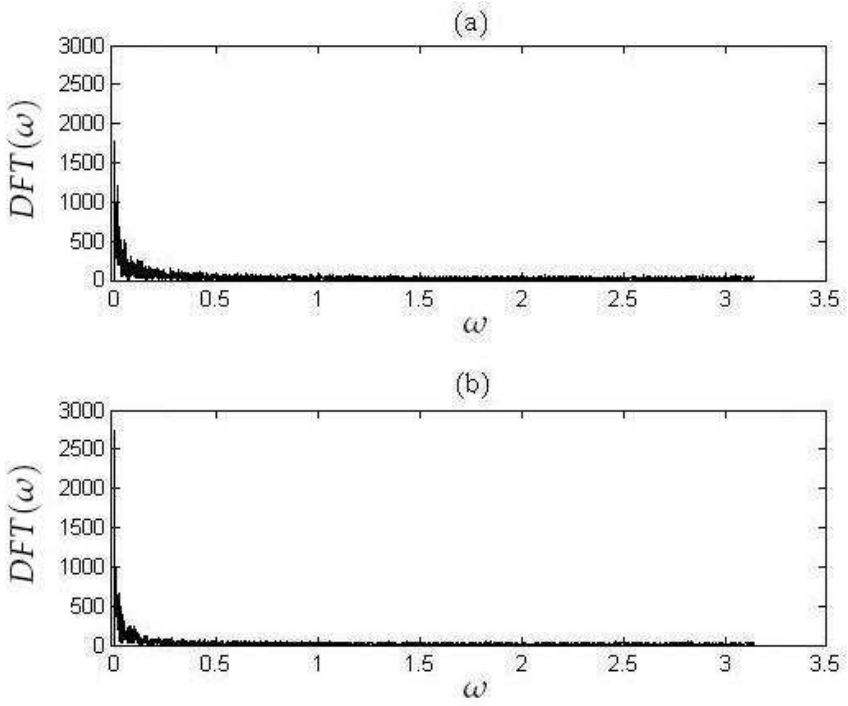

Figure B.10: Illustration of the DFT, for a 36° twisted DNA, obtained using (a) AMBER model and (b) SDE model.

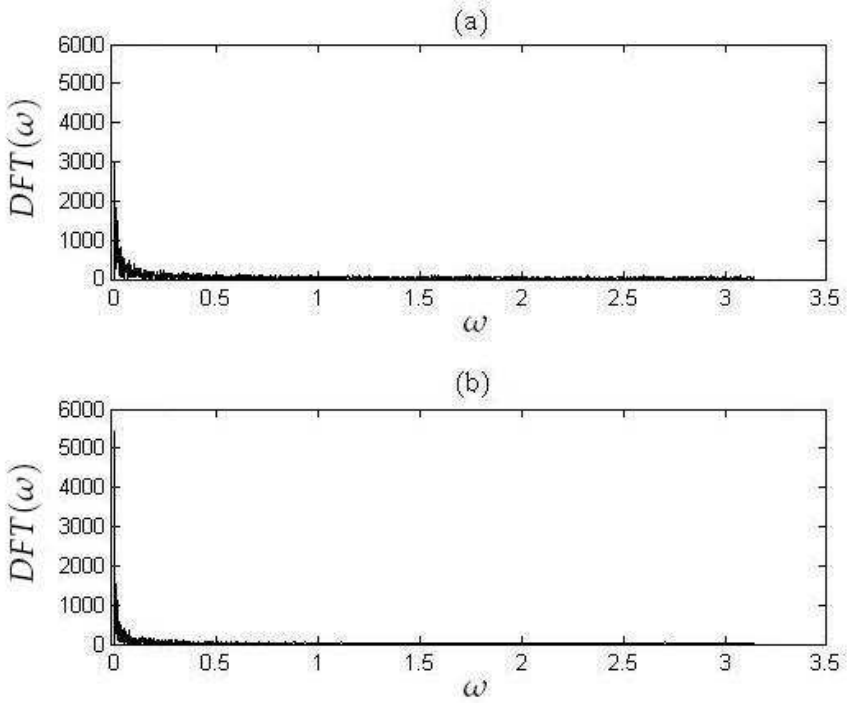

Figure B.11: Illustration of the DFT, for a 38° overtwisted DNA, obtained using (a) AMBER model and (b) SDE model.

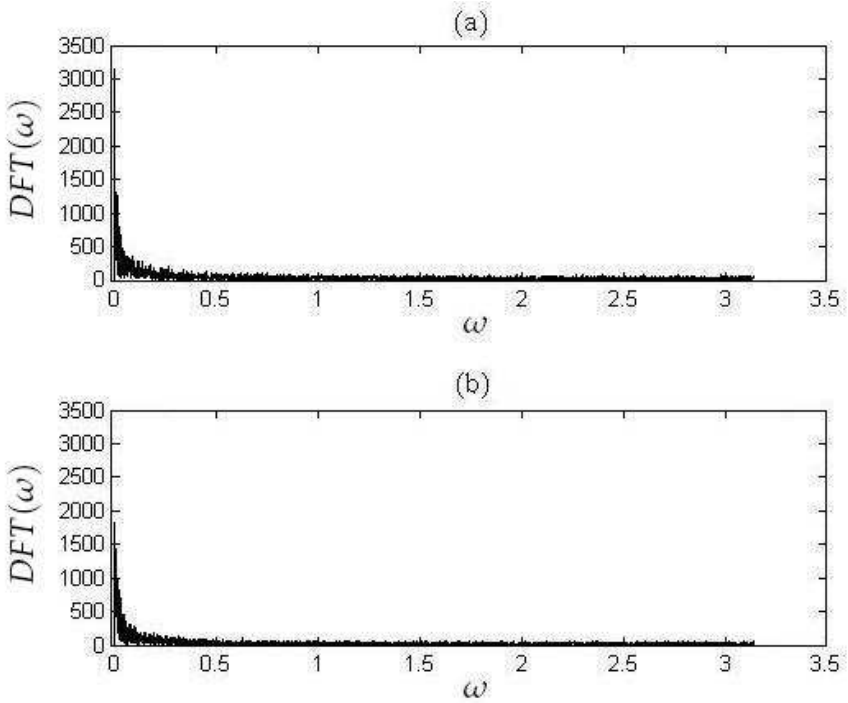

Figure B.12: Illustration of the DFT, for a 40° overtwisted DNA, obtained using (a) AMBER model and (b) SDE model.

B.3 Log-DFT figures

The expression of the DFTs from the figures presented in Appendix B.2 suggest that DNA might exhibit a self-organised behaviour, which is further analysed in Chapter 8. Figures B.13-B.18 illustrate a comparison between the log-DFT of AMBER and SDE data, specific to the A-F base-pair. This comparison is discussed in detail in Section 8.3.1, where we investigate the self-organised DNA behaviour obseved in both AMBER and SDE simulations, as well as the coefficients emphesizing the power law form of the DFTs.

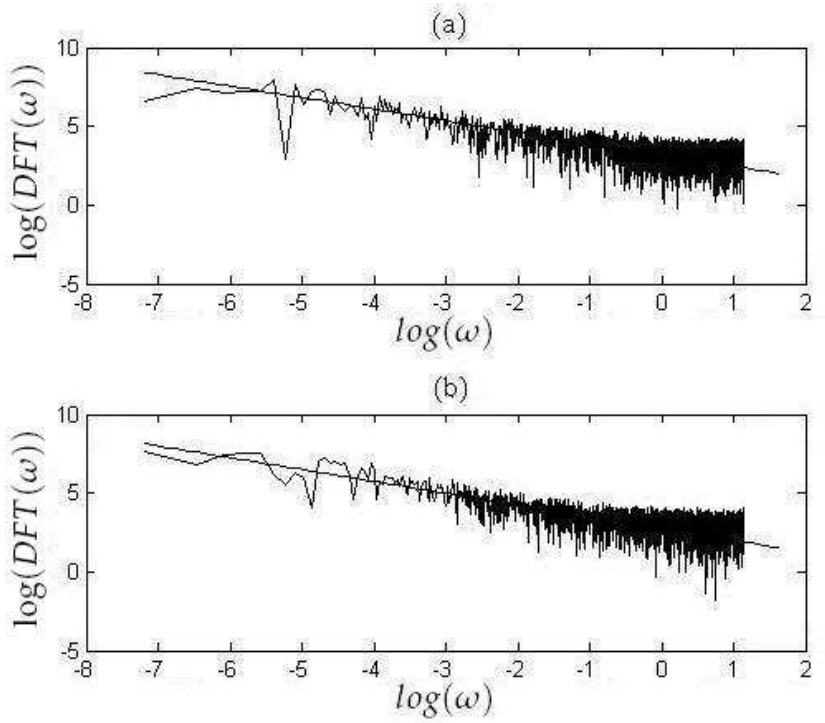

Figure B.13: Illustration of the log-DFT function, for a 30° undertwisted DNA, obtained using (a) AMBER model and (b) SDE model.

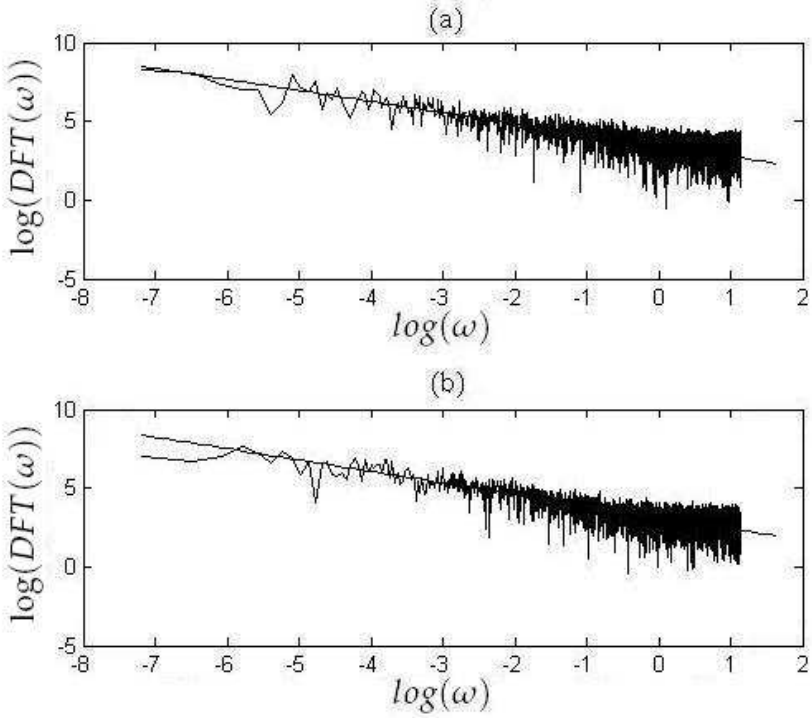

Figure B.14: Illustration of the log-DFT function, for a 32° undertwisted DNA, obtained using (a) AMBER model and (b) SDE model.

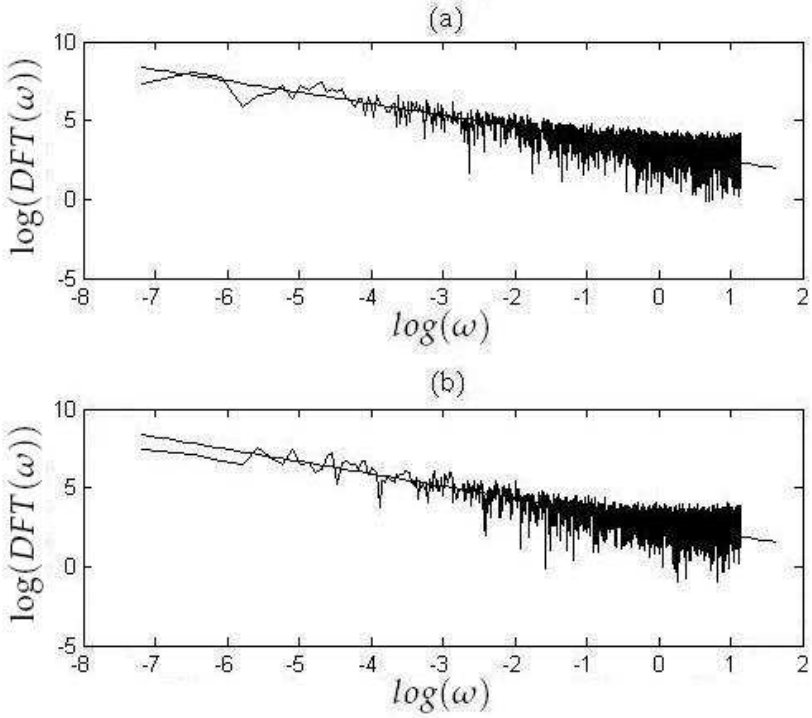

Figure B.15: Illustration of the log-DFT function, for a 33° undertwisted DNA, obtained using (a) AMBER model and (b) SDE model.

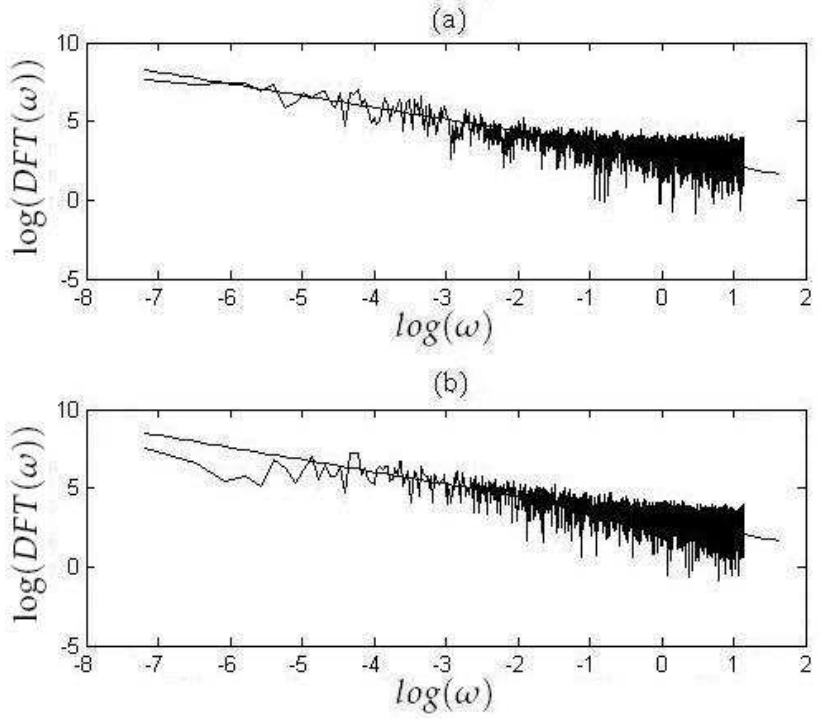

Figure B.16: Illustration of the log-DFT function, for a 35° undertwisted DNA, obtained using (a) AMBER model and (b) SDE model.

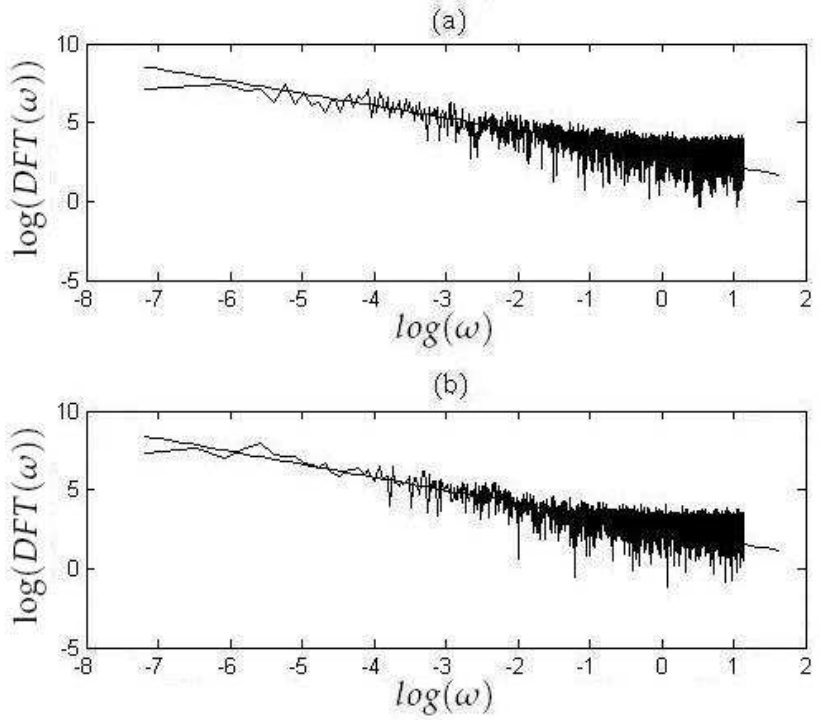

Figure B.17: Illustration of the log-DFT function, for a 36° twisted DNA, obtained using (a) AMBER model and (b) SDE model.

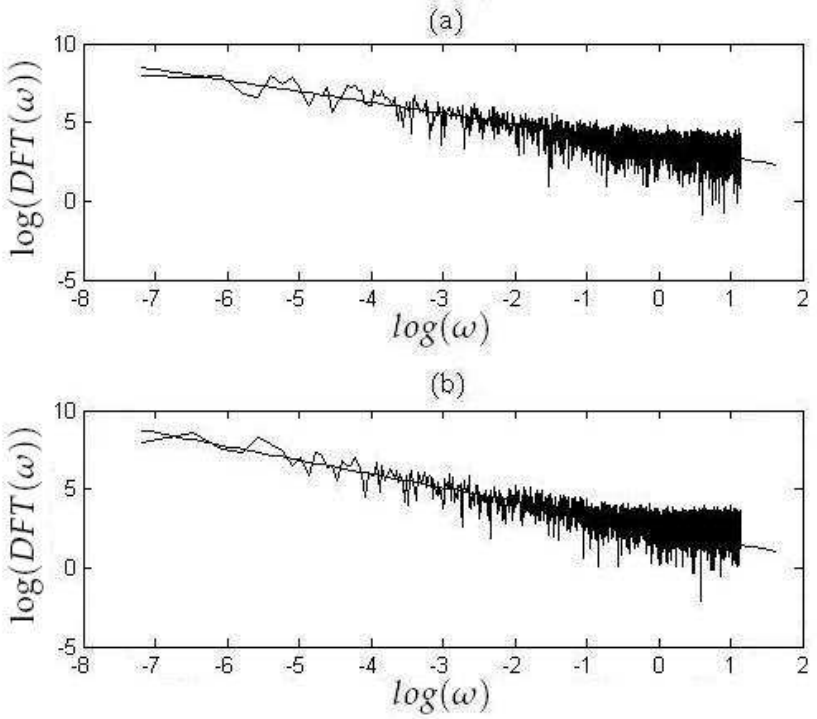

Figure B.18: Illustration of the log-DFT function, for a 38° overtwisted DNA, obtained using (a) AMBER model and (b) SDE model.

References

[1] A. Alvarez, F. R. Romero, J. F. R. Archilla, J. Cuevas, P.V. Larsen - *Breather trapping and breather transmission in a DNA model with an interface*, Eur Phys J B 51, 119-130 (2006).

[2] T. Ambjörnsson, R. Metzler - *Coupled dynamics of DNA-breathing and of proteins that selectively bind to single-stranded DNA*, Phys Rev E 72, 030901 (2005).

[3] T. Ambjörnsson, S. K. Banik, M. A. Lomholt, R. Metzler - *Master equation approach to DNA breathing in heteropolymer DNA*, Phys Rev E 75, 021908 (2007).

[4] P. Bak, K. Chen, C. Tang - *A forest-fire model and some thoughts on turbulence*, Phys Lett A 147, 297-300 (1990).

[5] P. Bak, C. Tang, K. Wiesenfeld - *Self-organized criticality: An explanation of the 1/f noise*, Phys Rev Lett 59, 381 - 384 (1987).

[6] P. Bak, C. Tang, K. Wiesenfeld - *Self-organized criticality*, Phys Rev A 38, 364-374 (1988).

[7] P. Bak, K. Christensen, L. Danon, T. Scanlon - *Unified scaling law for earthquakes*, Phys Rev Lett 88, 178501 (2002).

[8] A. Banerjee - *Self-organised criticality and* $1/f$ *noise in single-channel current of voltage-dependent anion channel*, Europhys Lett 73, 457-463 (2006).

[9] A. Banerjee, S. Merugu, I. S. Dhillon, J. Ghosh - *Clustering with Bregman divergences*, The Journal of Machine Learning Research 6, 1705-1749 (2005).

[10] S. K. Banik, T. Ambjörnsson, R. Metzler - *Stochastic approach to DNA breathing dynamics*, Europhys Lett 71, 852-858 (2005).

[11] M. Barbi, S. Cocco, M. Peyrard - *Helicoidal model for DNA opening*, Phys Lett A 253, 358-369 (1999).

[12] M. Barbi, S. Cocco, M. Peyrard, S. Ruffo - *A twist opening model for DNA*, J Bio Phys, 24, 97-114 (1999).

[13] M. Barbi, S. Lepri, M. Peyrard, N. Theodorakopoulos - *Thermal denaturation of a helicoidal DNA model*, Phys Rev E 68, 061909 (2003).

[14] J. D. Bashford - *Salerno's model of DNA re-analyzed: Could breather solitons have biological signifiance?*, J Bio Phys, 32, 27-47 (2006).

[15] J. van den Berg, R. Brouwer - *Self-organized forest-fires near the critical time*, Comm Math Phys 267, 265-277 (2006).

[16] J. van den Berg, A.A. Jsrai - *On the asymptotic density in a one-dimensional self-organized critical forest-fire model*, Comm Math Phys 254, 633-644 (2004).

[17] R. Bracewell, P. B. Kahn - *The Fourier Transform and its applications*, Am J Phys, 34 (8), 712-712 (1966).

[18] E. O. Brigham, C. K. Yuen - *The fast Fourier Transform*, IEEE Transactions on Systems, Man and Cybernetics, 8 (2), 146-146 (1978).

[19] K. Burrage, I. Lenane, G. Lythe - *Numerical methods for second-order stochastic differential equations*, SIAM Journal on Scientific Computing 29, 245-264 (2007).

[20] G. Caldarelli, R. Frondoni, A. Gabrielli1, M. Montuori1, R. Retzlaff, C. Ricotta - *Percolation in real wildfires*, Europhys Lett 56, 510-516 (2001).

[21] M. Cadoni, R. De Leo, G. Gaeta - *A composite model for DNA torsion dynamics*, Phys Rev E 75 021919 (2007).

[22] M. Cadoni, R. De Leo, G. Gaeta - *Sine-Gordon solitons, auxiliary fields and singular limit of a double pendulums chain*, J Phys A: Math Theor 40, 12917-12929 (2007).

[23] M. Cadoni, R. De Leo, G. Gaeta - *A symmetry breaking mechanism for selecting the speed of relativistic solitons*, J Phys A: Math Theor 40, 8517-8534 (2007).

[24] F. Caruso, A. Pluchino, V. Latora, S. Vinciguerra, A. Rapisarda - *Analysis of self-organized criticality in the Olami-Feder-Christensen model and in real earthquakes*, Phys Rev E 75, 055101(R) (2007).

[25] F. Caruso, V. Latora, A. Pluchino, A. Rapisarda, B. Tadic - *Olami-Feder-Christensen model on different networks*, Eur Phys J B 50, 243-247 (2006).

[26] D.A. Case, T.A. Darden, T.E. Cheatham, III, C.L. Simmerling, J. Wang, R.E. Duke, R. Luo, K.M. Merz, D.A. Pearlman, M. Crowley, R.C. Walker, W. Zhang, B. Wang, S. Hayik, A. Roitberg, G. Seabra, K.F. Wong, F. Paesani, X. Wu, S. Brozell, V. Tsui, H. Gohlke, L. Yang, C. Tan, J. Mongan, V. Hornak, G. Cui, P. Beroza, D.H. Mathews, C. Schafmeister, W.S. Ross, P.A. Kollman - *AMBER 9*, University of California, San Francisco (2006).

[27] S. C. Chapman, N. W. Watkins, R. O. Dendy, P. Helander, G. Rowlands - *A simple avalanche model as an analogue for magnetospheric activity*, Geophys Res Lett 25, 2397-2400 (1998).

[28] S. C. Chapman - *Inverse cascade avalanche model with limit cycle exhibiting period doubling, intermittency, and self-similarity*, Phys Rev E 62, 1905 - 1911 (2000).

[29] S. C. Chapman, N. Watkins - *Avalanching and self-organised criticality, a paradigm for geomagnetic activity?*, Space Science Reviews 95, 293-307 (2001).

[30] B. Chopard, M. Droz - *Cellular Automata Modeling of Physical Systems*, Cambridge University Press (1998)

[31] K. Christensen, *Reply on Klein and Rundle comment*, Phys Rev Lett 71, 1289 (1993).

[32] R. Cingolani, R. Rinaldi, G. Maruccio, A. Biasco - *Nanotechnology approaches to self-organized bio-molecular devices*, Phys E: Low-dimensional Systems and Nanostructures 13, 1229-1235 (2002).

[33] S. Cocco, M. Barbi, M. Peyrard - *Vector nonlinear Klein-Gordon lattices: General derivation of small amplitude envelope soliton solutions*, Phys Lett A 253, 161-167 (1999).

[34] S. Cocco, R. Monasson - *Statistical mechanics of torque induced denaturation of DNA*, Phys Rev Lett 83, 5178-5181 (1999).

[35] G. Consolini, P. De Michelis - *A revised forest-fire cellular automaton for the nonlinear dynamics of the Earth's magnetotail*, Journal of Atmospheric and Solar-Terrestrial Physics 63, 1371-1377 (2001).

[36] J. W. Cooley, J. W. Tukey - *An algorithm for the machine calculation of complex Fourier series*, Math Comput 19, 297-301 (1965).

[37] E. Cubero, E. C. Sherer, F. J. Luque, M. Orozco, C. A. Laughton - *Observation of spontaneous base pair breathing events in the molecular dynamics simulations of a difluorotoluene-containing DNA oligonucleotide*, J Am Chem Soc, 121, 8653-8654, (1999).

[38] E. Cubero, J. Cuevas, P. G. Kevrekidis - *Nucleation of breathers via stochastic resonance in nonlinear lattices*, preprint (2009).

[39] S. Cuenda, A. Sánchez, N. R. Quintero - *Does the dynamics of sine-Gordon solitons predict active regions of DNA?*, Physica D 223, 214-221 (2006).

[40] J. Cuevas, F. Palermo, J. F. R. Archilla, F. R. Romero - *Moving breathers in a bent DNA model*, Phys Lett A 299, 221-225 (2002).

[41] J. Cuevas, F. Palermo, J. F. R. Archilla, F. R. Romero - *Moving breathers in bent DNA with realistic parameters*, Mod Phys Lett B 18(25), 1319-1326 (2004).

[42] T. Dauxois, M. Peyrard, A. R. Bishop - *Dynamics and thermodynamics of a nonlinear model for DNA denaturation*, Phys Rev E 47, 684-695 (1993).

[43] M. L. Deng, W. Q. Zhu - *Stochastic dynamics and denaturation of thermalized DNA*, Phys Rev E 77, 021918 (2008).

[44] C. Domb, M.S. Green, J.L. Lebowitz - *Phase transitions and critical phenomena*, Academic Press (2001).

[45] B. Drossel, F. Schwabl - *Self-organized critical forest-fire model*, Phys Rev Lett 69, 1629-1632 (1992).

[46] S. W. Englander, N. R. Kallenbach, A. J. Heeger, J. A. Krumhansl, S. Litwin - *Nature of the open state in long polynucleotide double helices: Possibility of soliton excitations*, Proc Nat Acad Sci 77, 7222-7226 (1980).

[47] L. Eriksson, E. Johansson, N. Kettaneh-Wold, S. Wold - *Multi- and megavariate data analysis: principles and applications*, Umetrics Academy (2001).

[48] T. A. Evans, K. R. Seddon - *Hydrogen bonding in DNA – a return to the status quo*, Chem Commun, 2023-2024 (1997).

[49] R. Frigg - *Self-organised criticality – what it is and what it isn't*, Studies In History and Philosophy of Science Part A 34, 613-632 (2003).

[50] G. Gaeta, L. Venier - *Solitary waves in twist-opening models of DNA dynamics*, Phys Rev E 78, 011901 (2008)

[51] C. W. Gardiner - *Handbook of stochastic methods for physics, chemistry and the natural sciences*, Springer, third edition, Berlin (2004).

[52] K. M. Guckian, T. R. Krugh, E. T. Kool - *Solution structure of a DNA duplex containing a replicable difluorotoluene-adenine pair*, Nature Structural Biology 5, 954-959 (1998).

[53] H. Haken - *Cooperative phenomena in systems far from thermal equilibrium and in nonphysical systems*, Rev Mod Phys 47, 67-121 (1975).

[54] A. Hanke, R. Metzler - *Bubble dynamics in DNA*, J Phys A: Math Gen 36, 473-480 (2003).

[55] S. A. Harris, E. Gavathiotis, M. S. Searle, M. Orozco, C. A. Laughton - *Cooperativity in drug-DNA recognition: a molecular dynamics study*, J Am Chem Soc 123, 12658-12663 (2001).

[56] T. Hastie, J. Friedman, R. Tibshirani - *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*, Edition 9, Springer (2007).

[57] D. Hennig - *Formation and propagation of oscillating bubbles in DNA initiated by structural distortions*, Eur Phys J B 37, 391-397 (2004).

[58] D. Hennig, J. F. R. Archilla - *Streching and relaxation dynamics in double stranded DNA*, Physica A, 579-601 (2004).

[59] D. Hennig, J. F. R. Archilla - *Multi-site H-bridge breathers in a DNA-shaped double strand*, Physica Scripta 69, 150-160 (2004).

[60] D. Hennig, J. F. R. Archilla, J. M. Romero - *Modelling the thermal evolution of enzyme-created bubbles in DNA*, J R Soc Interface 2, 89-95 (2005).

[61] T. Hida - *Brownian motion*, Springer-Verlag, New York (1980).

[62] D. L. Hien, N. T. Nhan, V. Thanh Ngo, N. A. Viet - *Simple combined model for nonlinear excitations in DNA*, Phys Rev E 76, 021921 (2007).

[63] S. Homma, S. Takeno - *A coupled base-rotator model for structure and dynamics of DNA*, Prog Theor Phys, 72, 679-693 (1984).

[64] W. G. Hoover - *Canonical dynamics: Equilibrium phase-space distributions*, Phys Rev A 31, 1695-1697 (1985)

[65] J. E. Jackson - *A user's guide to principal components*, John Wiley and Sons (1991).

[66] H.-Y. Jan, C.-L. Lin, T.-S. Hwang - *Self-organized PID control design using DNA computing approach*, Journal of the Chinese Institute of Engineers 29(2), 251-261 (2006).

[67] P. Jung, A. Cornell-Bell, K. S. Madden, F. Moss - *Noise-induced spiral waves in astrocyte syncytia show evidence of self-organized criticality*, J Neurophysiol 79, 1098-1101 (1998).

[68] Y. Kafri, D. Mukamel, L. Peliti - *Melting and unzipping of DNA*, Eur Phys J B - Condensed Matter and Complex Systems, 27(1), 135-146 (2002).

[69] G. Kalosakas, K. Ö. Rasmussen, A. R. Bishop - *Nonlinear excitations in DNA: polarons and bubbles*, Synthetic Metals 141, 93-97 (2004).

[70] G. Kalosakas, S. Ares - *Dependence on temperature and GC content of bubble length distributions in DNA*, J Chem Phys 130, 235104 (2009).

[71] J. Kertkszt, L. B. Kiss - *The noise spectrum in the model of self-organised criticality*, J Phys A: Math Gen 23, L433-L440 (1990).

[72] Y. Kishimoto, T. Tajima, W. Horton, M. J. LeBrun, J. Y. Kim - *Theory of self-organized critical transport in tokamak plasmas*, Phys Plasmas 3, 1289 (1996).

[73] W. Klein, J. Rundle - *Comment on "Self-organized criticality in a continuous, nonconservative cellular automaton modeling earthquakes"*, Phys Rev Lett 71, 1288 (1993).

[74] M. A. Kramer - *Nonlinear principal component analysis using autoassociative neural networks*, A I Ch E J 37(2), 233-243 (1991).

[75] T. Krink, P. Rickers, R. Thomsen - *Applying Self-Organised Criticality to Evolutionary Algorithms*, Parallel Problem Solving from Nature PPSN VI, 375-384 (2007).

[76] P.V. Larsen, P. L. Christiansen, O. Bang, J. F. R. Archilla, Yu. B. Gaididei -*Bubble generation in a twisted and bent DNA-like model*, Phys Rev E 70, 036609 (2004).

[77] E. Lennholm, M. Hornquist - *Revisiting Salerno's sine-Gordon model of DNA: Active regions and robustness*, Physica D, 177, 233-241 (2003).

[78] A. Matsumoto, W. K. Olson - *Sequence-dependent motions of DNA: A normal mode analysis at the base-pair level*, Biophys, 83, 22-41 (2002).

[79] R. De Maesschalck, D. Jouan-Rimbaud, D. L. Massart - *The Mahalanobis distance*, Chemometrics and Intelligent Laboratory Systems 50, 1-18 (2000).

[80] P. C. Mahalanobis - *On the generalized distance in statistics*, Proceedings of the National Institute of Science of India 12, 49-55 (1936).

[81] P. C. Mahalanobis - *On tests and measures of group divergences*, J. Asiat. Soc. Bengal 26, 541-588 (1930).

[82] S. Maslov, M. Paczuski, P. Bak - *Avalanches and 1 / f noise in evolution and growth models*, Phys Rev Lett 73, 2162-2165 (1994).

[83] T. Mendes, C. Laughton - *Modelling how long-range charge transfer in DNA can improve lesion detection by repair proteins*, preprint (2007).

[84] R. Metzler, T. Ambjörnsson - *Dynamic approach to DNA breathing*, J Bio Phys 31, 399-350 (2005)

[85] J. Montaldi, M. Roberts and I. Stewart - *Existence of nonlinear normal modes of symmetric Hamiltonian systems*, Nonlinearity 3, 695-730 (1990).

[86] V. Muto, P. S. Lomdahl, P. L. Christiansen - *Two-dimensional discrete model for DNA dynamics: Longitudinal wave propagation and denaturation*, Phys Rev A, 42, 7452-7458 (1990).

[87] O.B. Naimark - *Structural-scaling transitions and localized distortion modes in the DNA double helix*, Phys Mesomec 10, 33-45 (2007).

[88] M. E. J. Newman - *Power laws, Pareto distributions and Zipf's law*, Contemporary Physics 46(5), 323-351 (2005).

[89] M. Nykter, N. D. Price, M. Aldana, S. A. Ramsey, S. A. Kauffman, L. E. Hood, O. Yli-Harja, I. Shmulevich - *Gene expression dynamics in the macrophage exhibit criticality*, Proc Natl Acad Sci 105(6), 1897-1900 (2008).

[90] B. Øksendal - *Stochastic differential equations*, Springer, sixth edition, New York (2005).

[91] Z. Olami, H. J. S. Feder and K. Christensen - *Self-organised criticality in a continuous, nonconservative cellular automaton modelling earthquakes*, Phys Rev Lett 68, 1244-1247 (1992).

[92] G. Peng, D.Tian - *The fractal nature of a fracture surface*, J Phys A: Math Gen 23, 3257-3261 (1990)

[93] M. Peyrard, A. R. Bishop - *Staticstical mechanics of a nonlinear model for DNA denaturation*, Phys Rev, 62, 2755-2758 (1989).

[94] M. Peyrard, J. Farago - *Nonlinear localization in thermalized lattices: Application to DNA*, Physica A 288, 199-217 (2000).

[95] M. Peyrard, S. C. López, D. Angelov - *Fluctuations in the DNA double helix*, Eur Phys J Special Topics 147, 173-189 (2007).

[96] M. Peyrard, S. C. López, G. James - *Modelling DNA at the mesoscale: a challange for nonlinear science?*, Nonlinearity 21, T91-T100 (2008).

[97] J. C. Phillips - *Scaling and self-organized criticality in proteins I*, Proc Natl Acad Sci 106(9), 3107-3112 (2009).

[98] J. C. Phillips - *Scaling and self-organized criticality in proteins II*, Proc Natl Acad Sci 106(9), 3113-3118 (2009).

[99] S. Pueyo - *Self-organised criticality and the response of wildland fires to climate change*, Climatic Change 82, 131-161 (2007).

[100] N. R. Quintero, A. Sánchez, F. G. Mertens - *Thermal diffusion of sine-Gordon solitons*, Euro Phys Jour B 16, 361-368 (2000).

[101] Z. Rapti, A. Smerzi, K. Ö. Rasmussen, A. R. Bishop, C. H. Choi, A. Usheva - *Lengthscales and cooperativity in DNA bubble formation*, Europhys Lett 74(3), 540 (2006).

[102] M. Remoissenet - *Low-amplitude breather and envelope solitons in quasi-one-dimensional physical models*, Phys Rev B 33, 2386-2392 (1985).

[103] C. J. Rhodes, R. M. Anderson - *Forest-fire as a model for the dynamics of disease epidemics*, Journal of the Franklin Institute 335, 199-211 (1998).

[104] D. Ruelle - *Small random perturbations of dynamical systems and the definition of attractors*, Commun Math Phys 82, 137-151 (1981).

[105] M. Salerno - *Discrete model for DNA-promoter dynamics*, Phys Rev A, 44, 5292-5297 (1991).

[106] J. Schlitter - *Estimation of absolute and relative entropies of macromolecules using the covariance matrix*, Chem Phys Lett 215, 617-621 (1993).

[107] B. Scholkopf, A. Smola, K.-R. Muller - *Nonlinear component analysis as a kernel eigenvalue problem*, Spemannstraße 38, 44 (1996).

[108] A. M. Selvam - *Quantumlike Chaos in the Frequency Distributions of the Bases A, C, G, T in Drosophila DNA*, arXiv:physics/0210068 (2002).

[109] A. M. Selvam - *Universal spectrum for DNA base C-G frequency distribution in Human chromosomes 1 to 24*, arXiv:physics/0701079 (2007).

[110] A. M. Selvam - *Universal spectrum for DNA base C-G frequency distribution in Takifugu rubripes (Puffer fish) genome*, arXiv:physics/07042114 (2007).

[111] M. S. Sharma, N. D. Arora - *OPTIMA: A nonlinear model parameter extraction program with statistical confidence region algorithms*, IEE Transactions 12, 0278-0070/93$03 (1993).

[112] J. Shlens - *A Tutorial on Principal Component Analysis*, Version 2, http://www.cs.cmu.edu/~elaw/papers/pca.pdf (2005).

[113] P. Sinha-Raya, L. de Aguab, H. J. Jensen - *Threshold dynamics, multifractality and universal fluctuations in the SOC forest-fire: facets of an auto-ignition model*, Phys D: Nonl Phen 157, 186-196 (2001).

[114] O. Sotolongo-Costa, F. Guzman, J. C. Antoranz, G. J. Rodgers, O. Rodriguez, J. D. T. Arruda Neto, A. Deepman - *A non extensive approach for DNA breaking by ionizing radiation*, arXiv:cond-mat/0201289 (2002).

[115] S. Takeno - *Nonlinear modes in helical lattices: Localized modes and kinks*, Phys Lett A 358, 390-395 (2006).

[116] S. Takeno, S. V. Dimitriev, P. G. Kevrekidis, A. R. Bishop - *Nonlinear lattices generated from harmonic lattices with geometric constraints*, Phys Rev B 71, 014304 (2005).

[117] S. Takeno, S. Homma - *Topological solitons and modulated structures of bases in DNA double helices*, Prog Theor Phys 70, 308-311 (1983).

[118] J. J.-L. Ting, M. Peyrard - *Effective breather trapping mechanism for DNA transcription*, Phys Rev E 53, 1011-1020 (1996).

[119] J. D. Watson, F. H. C. Crick - *Molecular structure of nucleic acids*, Nature 171, 737 (1953).

[120] J. A. D. Wattis - *Stationary breather modes of generalized nonlinear Klein-Gordon lattices*, J Phys A: Math Gen 31, 3301-3323 (1998).

[121] J. A. D. Wattis, S. A. Harris, C. R. Grindon, C. A. Laughton - *Dynamic model of base pair breathing in a DNA chain with a defect*, Phys Rev E, 63, 061903 (2001).

[122] J. A. D. Wattis - *Nonlinear breathing modes due to a defect in a DNA chain*, Phil Trans Roy Soc Lond A 362, 1461-1477 (2004).

[123] D. Weatherley, S. C. Jaume, P. Mora - *Evolution of Stress Deficit and Changing Rates of Seismicity in Cellular Automaton Models of Earthquake Faults*, Pure Appl Geophys 157, 2183-2207 (2000).

[124] A. Weinstein - *Normal modes for nonlinear Hamiltonian systems*, Inventiones Mathematicae, 20 (1), 47-57 (1973)

[125] G. Werner - *Metastability, criticality and phase transitions in brain and its models*, Biosystems 90, 496-508 (2007).

[126] S. Wold - *Principal component analysis*, Chemometrics and Intelligent Laboratory Systems 2, 37-52 (1987).

[127] S. Wolfram - *Cellular automata as model of complexity*, Nature 311, 419 (1984).

[128] L. V. Yakushevich, *Nonlinear DNA dynamics: a new model*, Phys Lett A 136, 413-417 (1989).

[129] L. V. Yakushevich - *Nonlinear DNA dynamics: hierarchy of the models*, Phys D 79, 77 (1994).

[130] L. V. Yakushevich - *Nonlinear Physics of DNA*, John Wiley & Sons, Chichester, UK (1998).

[131] L. V. Yakushevich, A. V. Savin, L. I. Manevitch - *Nonlinear dynamics of topological solitons in DNA*, Phys Rev E 66, 016614 (2002).

[132] S. Yomosa - *Soliton excitations in deoxyribonucleic acid (DNA) double helices*, Phys Rev A 27, 2120-2125 (1983).

[133] S. Yomosa - *Solitary excitations in deoxyribonucleic acid (DNA) double helices*, Phys Rev A 30, 474-480 (1984).

[134] M. Zacharias - *Rapid protein-ligand docking using soft modes from molecular dynamics simulations to account for protein deformability: binding of FK506 to FKBP*, PROTEINS: Structure, Function, and Bioinformatics 54, 759-767 (2004)

[135] L. L. van Zandt - *DNA solitons with realistic parameters values*, Phys Rev A 40, 6134-6137 (1989).

[136] S. Zdravković, M. V. Satarić - *High amplitude mode and DNA opening*, EuroPhys Lett 78, 38004 (2007).

[137] S. Zdravković, M. V. Satarić - *Resonance mode in DNA dynamics*, EuroPhys Lett 80, 38003 (2007).

[138] C. T. Zhang - *Soliton excitations in deoxyribonucleic acid (DNA) double helices*, Phys Rev A 35, 886-891 (1987).

[139] L.-Y. Zhang, H. Sun, J.-T. Lin - *Stretching vibration influence of the hydrogen bond on a localized excitation and thermodynamic properties of DNA double helices*, Phys Lett A 259, 71-79 (1999).

[140] W. Q. Zhu, Z. L. Huang, and Y. Suzuki - *Stochastic averaging and Lyapunov exponent of quasi partially integrable Hamiltonian systems*, Int J Non-Lin Mech 37, 419 (2002).

[141] J. Zinn-Justin - *Quantum Field Theory and Critical Phenomena*, Oxford University Press (2002).

[142] www.ambermd.org - Molecular dynamics package AMBER official website.