

keys : $2\phi + \alpha = \pi$
 $\theta + \frac{1}{2}\pi + \alpha + \frac{1}{2}\pi = 2\pi$

$$2\phi + \alpha = \pi$$

$$2\phi + (\pi - \theta) = \pi$$

$$\phi = \frac{1}{2}\theta$$

$$r_s^2 = 2r^2(1 - \cos\theta)$$

$$\gamma = \pi - 2\theta$$

$$\cos\gamma = -\cos(2\theta)$$

$$\omega = \pi - 2\phi = \alpha = \pi - \theta$$

$$\cos\omega = -\cos\theta$$

Chapter 1

INTRODUCTION

1.1 DNA

DNA is a double helical structure composed of ribose sugar (deoxiribose), aromatic bases (adenine, guanine, thymine and cytosine), and phosphate groups. The double helix structure is formed by two individual DNA strands held together by hydrogen bonds between individual bases [1]. DNA exists in several forms, for example B-, A-, C-, D-, T-, and Z-form. The B-form is the most common structure found in eukaryotic cells. In this thesis, the term DNA denotes the structure and dimension of the B-form. The general structure of DNA is illustrated in Figure 1.1(a)(b).

The nucleosome core particle (NCP), with a DNA strand wrapped around it, is approximately a disk-like cylinder of diameter 110 \AA and length 57 \AA . The disk center contains protein of two H3 and H4 subunits and H2A-H2B dimers [2]. This disk-like cylinder is also known as an "octameric histone core" because it contains eight proteins in total. The term "H2A", as well as "H2B, H3, and H4" does not denote a specific particle or structure. Instead, each refers to a variety of closely related structures or genetic roles. For example, H2A is coded by many genes including H2AFB1, H2AFB1, H2AFBJ, H2AFBX, etc.

In a cell, DNA wraps around the NCP by about 1.6-1.8 complete turns of the NCP circumference. At the points of contact of the DNA strand and NCP complex (when the DNA initially winds about the NCP and when it departs from the NCP surface) another single fifth histone called H1 is thought to be present, binding onto the proximate NCP surface. It is known that the regular DNA-protein combinations can be found once every

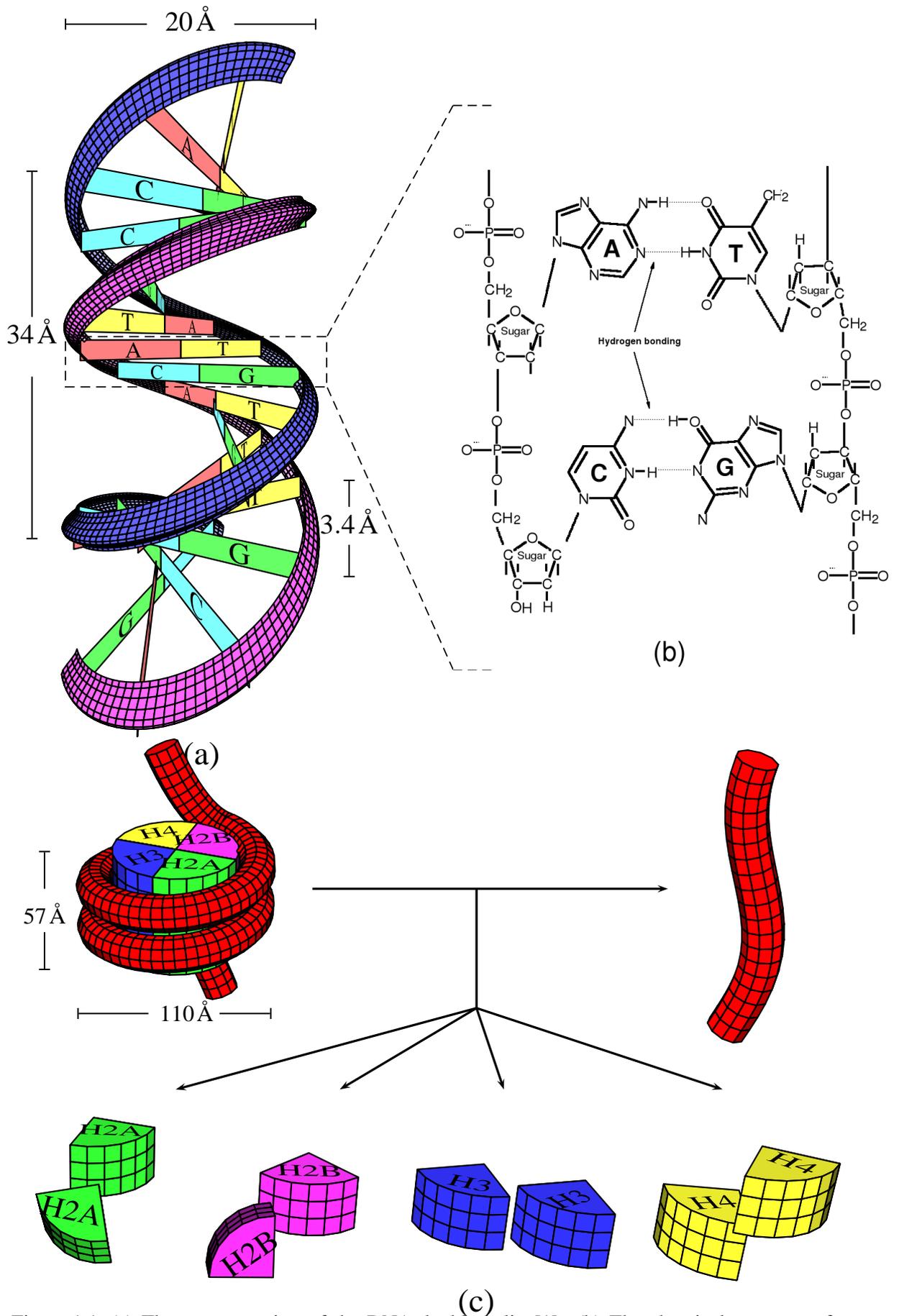


Figure 1.1: (a) The representation of the DNA double helix [1]. (b) The chemical structure of sugar (deoxyribose), bases (adenine, guanine, thymine and cytosine), and phosphate groups. (c) Dissociation of a DNA-NCP complex yields a DNA strand and eight histone proteins. [2, 3]

200 base-pairs [3]. The NCP or histone core binds to about 150 base-pairs of DNA, while the histone H1 binds to the remaining 50 base-pairs. Fig 1.1(c) illustrates the DNA–NCP wrapping and dissociation structure.

The nucleosomal DNA assemblage has primary roles in its ability to serve as a template for essential enzymatic activities such as replication, recombination, repair and transcription [4]. In nature, plants are subjected to salt concentration changes because they are largely immobile, but the environment conditions fluctuate. The high salinity of soils are attributable to natural processes such as weathering of mineral rocks and human intervention. Some experimental work has revealed the effect of Na, Cd and As ions on genotoxic cell damage. The genotoxic cell effect could be due to DNA strand breaks, DNA-protein cross-linking, oxidative DNA-damage, enhanced proliferation, depressed apoptosis and inhibited DNA repair [5–8]. However the mechanism of ions in damaging the nucleosomal DNA and the mechanism of certain molecules in increasing the salt tolerance of the organism are not well understood.

1.2 Molecular Dynamics Simulation

Following Yonezawa [9], there exists limitations in experimental research such as:

- a. The experimental inaccessibility of materials or situational setup
- b. The non-observability of a physical property
- c. The difficulty of controlling and defining the experimental environment
- d. The limitation of state-of-art apparatus.

On the other hand, computer simulations can address all these points and even provide a much wider scope in the elucidation of fundamental physics [9].

In molecular dynamics (MD) the spatial coordinates are obtained by numerically solving differential equations of motion and, hence, the positions are functions of time. The positions reveal the dynamics of individual molecules as in a motion picture. In other simulation methods the molecular positions are not temporally related. For instance, in *Monte Carlo* simulations the positions are generated stochastically such that a molecular configuration \mathbf{r}^N depends only on the previous configuration [10].

MD is applied to a system containing several hundred to several thousand atoms. This

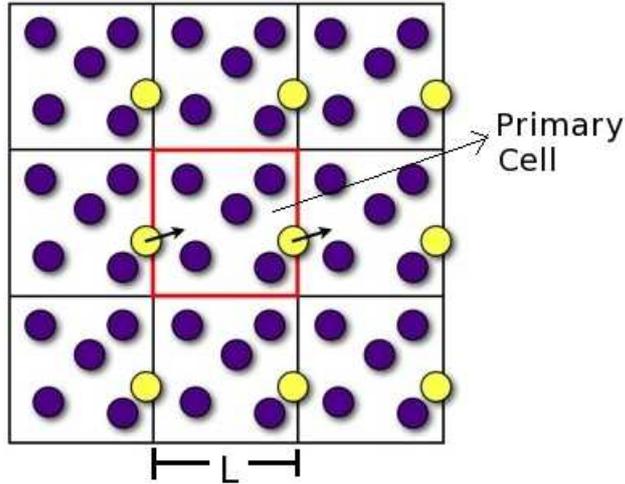


Figure 1.2: Periodic Boundary Condition.

system will be much affected by the surface of the MD cell that contain the particles. Bulk properties cannot be obtained by sampling over the entire cell because of these effects. To remove the surface effects, periodic boundary conditions (PBC) are used. In applying the PBC, we define our system containing N molecules inside a cubic box with volume V as the *primary cell*, where this primary cell is surrounded by its exact replicas in all directions. These replicas are called *image cells*. This cell replication is periodically extended and forms a macroscopic system which represents the bulk system of interest [11]. Fig. 1.2 depicts the application of periodic boundary conditions. As a mathematical illustration, take any reference point within the surface of a cubic box with box length L , where the initial space coordinate of particle i is $r_i(x_i^0, y_i^0, z_i^0)$, where x_i^0, y_i^0, z_i^0 are the distances between particle i and the origin over the three Cartesian coordinates. Suppose a particle moves outside the cubic box (i.e. in the x direction), where the new coordinate of particle i is $(r_i(x_i^t + L), y_i^t, z_i^t)$. If PBC are applied, the new coordinate of particle i becomes $r_i(x_i^t, y_i^t, z_i^t)$. Any atoms can freely move to other adjacent cells but the sum total of atoms in each cell will remain constant since any atom leaving a cell wall will spontaneously enter the same cell through the opposite wall[12].

In the canonical ensemble the number of particles, N , the volume, V , and the temperature, T , are fixed. Because the temperature is defined by the ensemble average of the kinetic energy, it is possible to fix T by adjusting the particles' velocity. Several types of thermostats, such as due to Berendsen, Langevin and Nose-Hoover have been proposed. The

Langevin thermostat (used here) utilizes the Langevin equation:

$$ma = -\xi v + f(r) + f', \quad (1.1)$$

where m is the particle mass, a is the acceleration, $f(r)$ is the conservative force, v is the velocity, ξ is a frictional constant, and f' is a random force. The random force is randomly determined from a Gaussian distribution.

The interactions between particles are determined by some equations called force fields. Force field methods (also known as molecular mechanics) ignore the electronic motions, and calculate the energy of the system as a functions of nuclear positions only [13]. Some examples of force fields are those generated by bond stretching, angle bending, torsional twist, out-of-plane bending, cross interactions term, electrostatic interactions, and van der Walls phenomena. Further details about the force fields used in our study is given in Chapter 2.

It is often convenient in MD simulation to use reduced units. Reduced units are obtained by converting constant values to a preferred scaled constant (e.g. 1.00) for a minimum set of independent variables. An important reason to use reduced units [see Frenkel and Smit [12]] that many combination of variables like density, temperature, energy and length all correspond to the same state in reduced units. Another reason is the numerical values of the quantities that we are computing (e.g. energy and acceleration) are either much less or much larger than 1. If we operate using such quantities in our standard floating point arithmetic, we face the risk that we might obtain results that were due to overflows and underflows. Further descriptions of reduced unit used in this research is given in Appendix A.

Chapter 2

RESEARCH METHODOLOGY

This chapter give details of the simulation methodology. It also discloses how a DNA–NCP salt solution system is modeled in a manner suitable for MD implementation.

2.1 The DNA, NCP and NaCl Salt Models

The DNA double helix is modeled as 360 negatively charged monomer spheres with radius 10 \AA and charge -12 (all in reduced units) linked linearly by a harmonic bonding potential. Each of these spheres represent 6 base pairs. In a cell, the DNA strand wraps around the so called nucleosome core particle (NCP). In this study, the NCP is represented by a large sphere with radius 35 \AA and charge $+150$. The simulations are performed both with and without the NCP particles, which when present are 12 in number. A number of counterions are added to neutralize the charge of the system. The NaCl salt is represented as a radius 2 \AA charged sphere of either $+1$ or -1 charge modeling single Na^+ or Cl^- ions respectively. These size reflect the actual size of the respective particles as determined by structured analysis [2]. For DNA concentration 0.005 mg/ml , the salt concentration was chosen to be in the $0.0\text{-}0.25 \text{ mM}$ range while for DNA concentration 2.0 mg/ml the salt concentration is within the $0.0\text{-}100 \text{ mM}$ range. The upper limit corresponds to the maximum computational resources available here. The Bjerrum length of 7.13 \AA at temperature 300 K corresponds to water solvent dielectric. The Langevin thermostat is applied to regulate the equilibrium temperature in the NVT ensemble. Periodic boundary conditions are applied to avoid surface effects.

2.2 Research Equipment

The computers used to perform the simulations included the following:

1. High performance computing (HPC) cluster system at Pusat Teknologi Maklumat (PTM), University of Malaya. This HPC system consist of 1 master node with 4 processors and 4 compute nodes with 8 processors each. The master Intel Xeon X5272 3.40GHz processors and total memory of 16.5 GB. All compute nodes has Intel Xeon E5440 2.83GHz processors with total memory 16.5 GB per node.
2. Single multiprocessing (SMP) machine at Pusat Teknologi Maklumat (PTM), University of Malaya. This SMP consist of Intel Dual-Core Itanium 9130M processors. The total memory of this SMP machine is 128 GB.
3. IBM-Cluster at MIMOS Berhad. This cluster 1 master node with 4 processors and 8 compute nodes with 4 processors each. The master node has Dual-Core AMD Opteron 2218 processors with total memory of 4 GB. All compute nodes have Dual-Core AMD Opteron 2220 processors with total memory 16.5 GB per node.

2.3 Research Method

ESPResSo¹, *Extensible Simulation Package for Research on Soft Matter*, is the package used for molecular dynamics simulation. ESPResSo runs on linux platforms which require additional software which include TCL², FFTW³ and MPI library (i.e.: OpenMPI⁴, LAM/MPI⁵, MPICH⁶). In order to run a simulation, we need to submit a TCL script to an ESPResSo executable file. Two TCL scripts were used for our simulation. The first form is used for 'equilibrium' run. The first form declares the particles properties, system parameters, force fields and the energy measurement commands. We run the TCL script until the system reaches equilibrium before using the second TCL script for production runs which contain commands for the sampling of energy, end-to-end distance, radius of

¹http://espressowiki.mpip-mainz.mpg.de/wiki/index.php/Main_Page

²<http://www.tcl.tk/software/tcltk/>

³<http://www.fftw.org/>

⁴<http://www.open-mpi.org/software/ompi/v1.4/>

⁵<http://www.lam-mpi.org/7.1/download.php>

⁶<http://www.mcs.anl.gov/research/projects/mpich2/>

gyration, radial distribution function, contour length, average bond angle and particle coordinates for snapshots. Examples of both TCL scripts are given in Appendix B.

For further and more detailed analysis utilizing hypernetted chain approximation, Percus-Yevick approximation, persistence length algorithms, linear regression and standard deviation formulas and expression for harmonic bond constant determination, coordination number, determination using Debye-Huckel approximation, etc, we write our own C++ or TCL codes to numerically derive the above properties [20–22].

2.4 Particles Properties

The table below lists the physical variables used in simulation. These values are chosen to mimic the particles properties obtained from experiment. The radii of DNA and NCP are

Parameter	DNA monomer	NCP	Na	Cl
Radius (Å)	10.0	35.0	2.0	2.0
Soft core radius (Å)	2.0	2.0	2.0	2.0
Hard core radius (Å)	8.0	33.0	0.0	0.0
Charge (e)	-12	+150	+1	-1
Mass ($\times 10^{-26}$ kg)	612.62	18026.68	3.819	5.889
Mass (reduced unit)	160.41	4270.26	1.0	1.54

Table 2.1: The properties of DNA, NCP, Na^+ and Cl^- in simulation

taken from the experimental data which are presented in the Introduction Chapter. The radii of Na^+ and Cl^- are chosen from the data given by Simonin et al. [18] and [19]. In their work, they fitted the experimental mean activity coefficient of some simple salt solutions with theoretical calculation. They applied mean spherical approximation (MSA) as the theoretical framework with only one adjustable parameter, the effective ionic diameter. Simonin et al. [18] obtained the effective diameter for the cation Na^+ in NaCl solution as 3.90 Å. Fawcett and Tikanen [19] obtained 3.88 Å for the diameter for Cl^- ion as the best fit to the experimental mean activity coefficient of NaCl solution. The radius of a particle i defines the closest distance for any other particle’s surface contact to the surface of particle i . We apply the Lennard-Jones force field to control and fix the closest

ionic distances possible.

Regarding the values of hard core and soft core radius, the main purpose of selecting these values is not related to the electronic configuration, electron density or van der Waals and London interactions between ion-ion or ion-solvent pairs. This is because the Lennard-Jones force-field used in this simulation is for the purely repulsive part and the attractive part (due to London forces) is omitted by truncation of the Lennard-Jones potential (Fig. 2.3). Thus the value of the soft core radii of all ions are chosen to provide a smooth repulsive displacement when two particles meet in contact. From Fig. 2.3 we notice that the steep Lennard-Jones energy increase (implying a large repulsive force) about the closest ionic contact ($r_{\text{off}} + \sigma$) prevents two particles from getting closer than the sum of their radii.

2.5 System Parameters

The system parameters applied for this research are given in the Table 2.2. The simulation

Parameter	value
Temperature (K)	300
Temperature (reduced unit)	1
Bjerrum length (\AA)	7.13
DNA bond length (\AA)	20.4
Thermostat	Langevin
Ensemble	NVT

Table 2.2: The general parameter of the simulation system

uses reduced units. The detail derivation of the reduced units used in this research is given in Appendix A. The NVT ensemble implies that the simulations are conducted in fixed number of particles, volume and temperature. DNA bond length 20.4 \AA corresponds to the DNA axial distance comprising six base pairs (i.e. 6×3.4).

2.6 Force Fields

2.6.1 Intramolecular Force Fields

Harmonic Stretching Bond

The monomer-monomer bonding interaction is governed by the harmonic oscillator potential:

$$U_h(r) = k_h(r - r_h)^2, \quad (2.1)$$

where k_h is the harmonic bond constant and r_h is the equilibrium harmonic distance. Figure 2.1 depicts the harmonic stretching phenomenon. To determine k_h , we equilibrate the

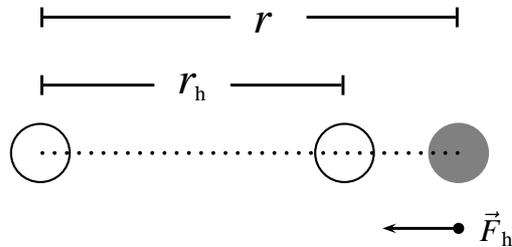


Figure 2.1: An illustration of a monomer-monomer displacements (grey circle) from its equilibrium position (white circles).

force between two bonded monomers. The detail calculation of k_h and r_h will be given at the end of this Chapter.

Bending Angle

The bending angle potential of three consecutive monomers is defined by the following equation:

$$U_\theta(\theta) = k_\theta(\theta - \theta_0)^2, \quad (2.2)$$

where k_θ is the bending constant. θ_0 is the equilibrium bending angle, which equals to zero for our DNA polymer model [see Fig. 2.2]. To obtain the k_θ , we relate it with the experimental DNA persistence length. The details of DNA persistence length are described in Chapter 3.

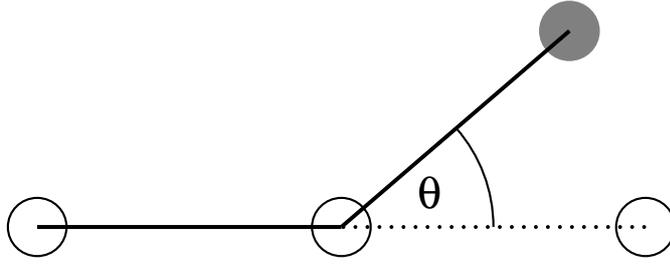


Figure 2.2: An illustration of a bending angle atomic displacement (grey circle) from its equilibrium bending angle $\theta = \theta_0 = 0$ (white circles).

2.6.2 Intermolecular Force Fields

Lennard-Jones Interaction

To model the short range interaction, we use the Lennard-Jones (LJ) potential. LJ potential equation is [14]:

$$U_{\text{LJ}}(r) = 4\epsilon_{\text{LJ}} \left\{ \left(\frac{\sigma}{r - r_{\text{off}}} \right)^{12} - \left(\frac{\sigma}{r - r_{\text{off}}} \right)^6 + c_{\text{shift}} \right\}, \quad (2.3)$$

$$F_{\text{LJ}}(r) = 4\epsilon_{\text{LJ}} \left(\frac{12\sigma^{12}}{(r - r_{\text{off}})^{13}} - \frac{6\sigma^6}{(r - r_{\text{off}})^7} \right), \quad (2.4)$$

where $\epsilon_{\text{LJ}} = k_{\text{B}}T = 4.14195 \times 10^{-21}$ J is the Lennard-Jones energy unit, r is the interparticle distance, σ is the sum of the soft core radius, r_{off} is the sum of the hard core radius and c_{shift} is a constant such that $U_{\text{LJ}} = 0$ at distance r_{cut} . r_{cut} is a distance parameter defined where at a distance larger than the r_{cut} the LJ interaction vanishes and $U_{\text{LJ}} = 0$. The sixth power (r^{-6}) term of the LJ potential (Eq. 2.3) represents the attractive van der Waals interaction due to electron correlations [11]. The twelfth power term (r^{-12}) models qualitatively the strongly repulsive interaction based on the Pauli exclusion principle. There are no strong argument concerning the exponent of the term r^{-12} in Eq. 2.3. The r^{-12} factor is computationally convenient because its value is the square of the r^{-6} term. In this research $r_{\text{cut}} = 2^{1/6}\sigma$, implying that the purely repulsive Lennard-Jones potential is used.

Coulombic Interaction

The Coulombic potential represents a long range interaction. ESPResSo uses the particle-particle-particle mesh (P3M) method to calculate the Coulombic interaction [15–17]. The

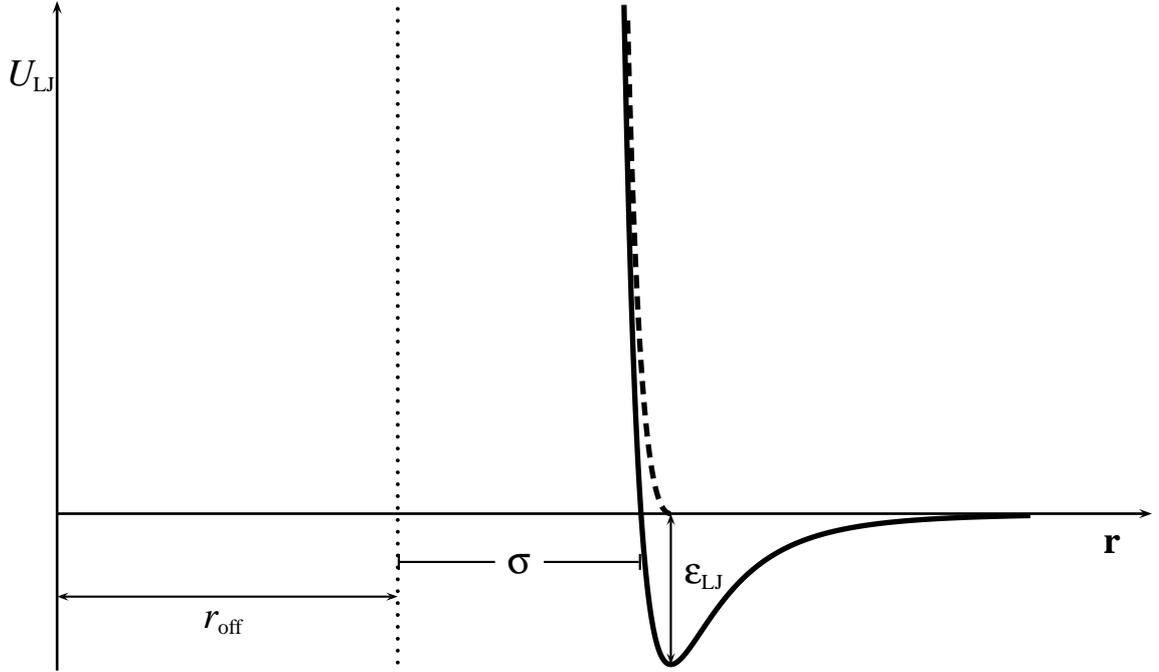


Figure 2.3: The energy profile of Lennard-Jones (LJ) potential (Eq. 2.3). The solid line is the original LJ model. The dashed line represents the shifted and truncated LJ potential at $r_{\text{cut}} = 1.1225\sigma$. The vertical dotted line represents the sum of hard core radii of interacting particles.

Coulombic-P3M potential is given by [14]:

$$U_{\text{C-P3M}}(r) = l_{\text{B}} k_{\text{B}} T \frac{q_1 q_2}{r}, \quad (2.5)$$

where q_i is the charge of particle i , r is the inter-particle distance, T is the temperature and k_{B} is the Boltzmann constant. The Bjerrum length, l_{B} , is defined by:

$$l_{\text{B}} = e_0^2 / 4\pi\epsilon k_{\text{B}} T, \quad (2.6)$$

where e_0 is the elementary proton charge 1.602×10^{-19} C and ϵ is the dielectric constant of the medium.

2.7 Particle Amounts in the Simulation

The system set chosen is a fixed number of DNA monomers and NCP particles in a cubic box of known volume implying a fixed DNA monomer and NCP concentration. We then add salt ions into this system. Below are the number of DNA, NCP, counterion and salt

particles that must be present to correspond to the stated concentration in our simulation cubic box. The set described here is given for purposes of illustration only. Only a subset of these system size were simulated because our program algorithms and computers could not cope with larger particle numbers.

2.7.1 DNA–Salt System

DNA polymer = 1

DNA monomer amount / polymer = 360

NCPs amount=0

Na⁺ counterion amount=4320

DNA Concentration (mg/ml)	Box Length (Å)	Salt Concentration (mM)	Na ⁺ + Cl ⁻ from Salt	Total Particle Number
0.005	7612.0052	0.001	530	5210
0.005	7612.0052	0.01	5310	9990
0.005	7612.0052	0.1	53102	57782
0.005	7612.0052	0.25	132758	137438
2.0	1033.1287	0.01	12	4692
2.0	1033.1287	1.0	1326	6006
2.0	1033.1287	10.0	13276	17956
2.0	1033.1287	50.0	66382	71062
2.0	1033.1287	100.0	132766	137446

Table 2.3: Particle amounts in the DNA-Salt system (without NCP in simulation).

2.7.2 DNA–NCP–Salt System

The following data incorporates NCP particles into the simulation. The numbers correspond to the number of particle in the simulation cell (box).

DNA polymer = 1

DNA monomer amount / polymer = 360

NCPs amount=12

Na⁺ counterion amount=2520

DNA Concentration (mg/ml)	Box Length (Å)	Salt Concentration (mM)	Na ⁺ + Cl ⁻ from Salt	Total Particle Number
0.005	7612.0052	0.001	530	3422
0.005	7612.0052	0.01	5310	8202
0.005	7612.0052	0.1	53102	55994
0.005	7612.0052	0.25	132758	135650
2.0	1033.1287	0.01	12	2904
2.0	1033.1287	1.0	1326	4218
2.0	1033.1287	10.0	13276	16168
2.0	1033.1287	50.0	66382	69274
2.0	1033.1287	100.0	132766	135658

Table 2.4: Particle amounts in the DNA–NCP–Salt system.

2.8 Determining the Harmonic Bonding Constant k_h and the Equilibrium Harmonic Distance r_h

We determine the parameter k_h and r_h in Eq. 2.1 for what follows. The defined bond length, b , between two monomers is 20.4 Å. At equilibrium, we assume that the total force between two bonded monomers is zero when the monomer-monomer distance equals the bond length b .

$$\vec{F}_{\text{total}}(b, \Omega) = \vec{F}_{\text{harmonic}}(b, \Omega) + \vec{F}_{\text{bending}}(b, \Omega) + \vec{F}_{\text{LJ}}(b, \Omega) + \vec{F}_{\text{Coulomb}}(b, \Omega) = 0, \quad (2.7)$$

where Ω are any variables involved in the respective force calculations. We regard the charge repulsions between non-adjacent monomers gives secondary effects compared to the adjacent monomers repulsion. In addition, the repulsive forces between a monomer and other monomers from opposite directions will decrease the nett force. Computing the additive forces from non-adjacent monomers can be complicated since we need to

account the position changes due to bending. Then we calculate $\vec{F}_{\text{total}}(b, \Omega)$ as the force between two adjacent monomers only. Because this force only involves two contiguous monomers, $\vec{F}_{\text{bending}} = 0$. We calculate $F_{\text{LJ}}(b, \Omega)$ and $F_{\text{Coulomb}}(b, \Omega)$ with the parameters stated earlier for our simulations.

$$\vec{F}_{\text{LJ}}(b) = 4\epsilon_{\text{LJ}} \left(\frac{12\sigma^{12}}{(b - r_{\text{off}})^{13}} - \frac{6\sigma^6}{(b - r_{\text{off}})^7} \right), \quad (2.8)$$

where $\epsilon_{\text{LJ}} = 4.142 \times 10^{-21}$, $\sigma = 4 \times 10^{-10}$ m, $b = 20.4 \times 10^{-10}$ m, and $r_{\text{off}} = 16 \times 10^{-10}$ m. Thus:

$$\begin{aligned} \vec{F}_{\text{LJ}}(b) &= 4 \times 4.142 \times 10^{-21} \left(\frac{12(4 \times 10^{-10})^{12}}{(4.4 \times 10^{-10})^{13}} - \frac{6(4 \times 10^{-10})^6}{(4.4 \times 10^{-10})^7} \right) \\ &= 16.568 \times 10^{-21} \left(\frac{2}{(1.1)^{13}} - \frac{1}{(1.1)^7} \right) \times 10^{10} \times 1.5 \\ &= 1.64 \times 10^{-11} = 0.164 \times 10^{-10} \text{ N} \end{aligned}$$

$$\vec{F}_{\text{Coulomb}}(b) = l_{\text{B}} k_{\text{B}} T \frac{q_1 q_2}{b^2}, \quad (2.9)$$

with $l_{\text{B}} = 7.13 \times 10^{-10}$ m, $k_{\text{B}} = 1.38 \times 10^{-23}$, $q_1 = q_2 = -12$, and $b = 20.4 \text{ \AA} = 20.4 \times 10^{-10}$ m.

$$\begin{aligned} \vec{F}_{\text{Coulomb}}(b) &= \frac{7.13 \times 10^{-10} \times 1.38 \times 10^{-23} \times 300 \times -12 \times -12}{(20.4 \times 10^{-10})^2} \\ &= 1.021 \times 10^{-10} \text{ N} . \end{aligned}$$

Inserting $\vec{F}_{\text{LJ}}(b)$ and $\vec{F}_{\text{Coulomb}}(b)$ in Eq. 2.7 ($\vec{F}_{\text{bending}} = 0$). We obtain:

$$\vec{F}_{\text{total}}(b) = \vec{F}_{\text{harmonic}}(b) + \vec{F}_{\text{LJ}}(b) + \vec{F}_{\text{Coulomb}}(b) = 0 \quad (2.10)$$

$$\vec{F}_{\text{harmonic}}(b) = -k_{\text{h}}(b - r_{\text{h}}) = -\vec{F}_{\text{LJ}}(b) - \vec{F}_{\text{Coulomb}}(b)$$

$$k_{\text{h}}(b - r_{\text{h}}) = (0.164 + 1.021) \times 10^{-10}$$

$$k_{\text{h}}(20.4 \times 10^{-10} - r_{\text{h}}) = 1.185 \times 10^{-10} \quad . \quad (2.11)$$

The unit of r_h in Eq. 2.11 is the meter. For simplification, we omit the factor 10^{-10} to obtain the r_h in Angstroms. Thus:

$$r_h = 20.4 - \frac{1.185}{k_h} . \quad (2.12)$$

We use reduce units for the harmonic constant k_h . We had calculated the relationship between k_h and its reduced unit k_h^* for our simulation, where $k_h^* = k_h \times 122.794$. See Appendix A for the details of the $k_h - k_h^*$ calculation. Then

$$r_h = 20.4 - \frac{1.185}{k_h^*/122.794} \quad (2.13)$$

$$= 20.4 - 145.511/k_h^* . \quad (2.14)$$

The reason for replacing k_h with k_h^* in Eq. 2.13 is because we wish to obtain the relationship between r_h and k_h directly using the parameter k_h^* , which is parameter used in our simulation. The value of k_h^* in Eq. 2.14 will be converted to its unreduced form in the Boltzmann integration that follows.

The Boltzmann probability factor is

$$P(r) = \frac{e^{-U(r)/k_B T}}{\int e^{-U(r)/k_B T} dr} , \quad (2.15)$$

assuming the degeneracy is the same over all possible r . Then

$$\bar{r} = \int rP(r) dr = \frac{\int r e^{-U(r)/k_B T} dr}{\int e^{-U(r)/k_B T} dr} , \quad (2.16)$$

where \bar{r} is the average equilibrium distance between bonded monomers. It equals the DNA bond length 20.4 \AA . The integration of Eq. 2.16 is performed from zero to infinity. The potential energy $U(r)$ contains the Lennard Jones, Coulombic and harmonic stretching potential energies.

$$\bar{r} = \frac{\int_0^\infty r e^{-(U_{LJ}(r)+U_{Coulomb}(r)+U_{harmonic}(r))/k_B T} dr}{\int_0^\infty e^{-(U_{LJ}(r)+U_{Coulomb}(r)+U_{harmonic}(r))/k_B T} dr} . \quad (2.17)$$

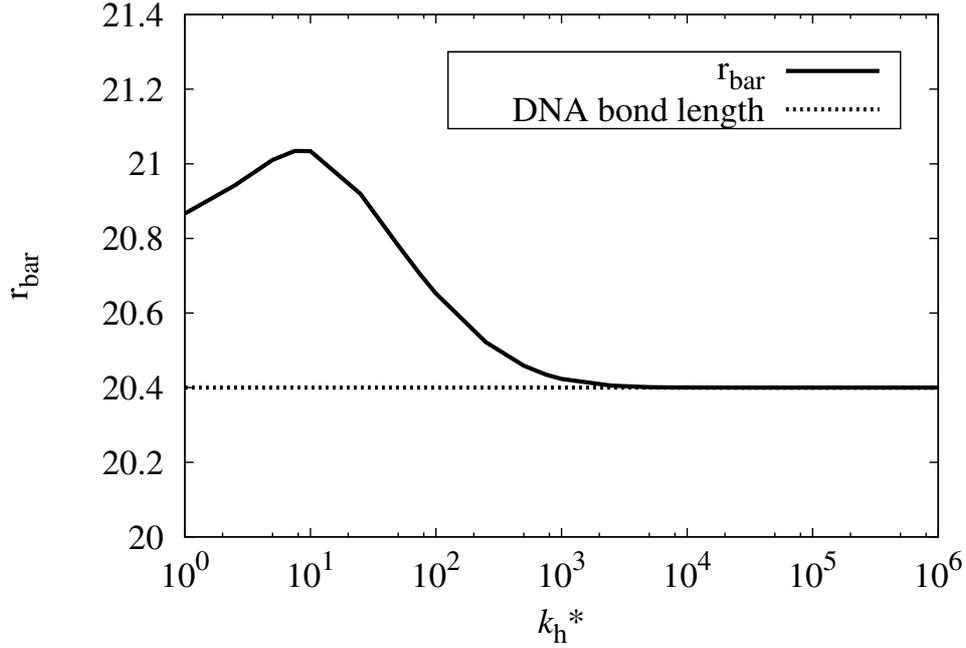


Figure 2.4: k_h^* and \bar{r} (written r_{bar} in the graph) relationship obtained from the integration of Eq. 2.17.

The integration of the Eq. 2.17 involved a complicated gamma function, where numerical integration was used. By choosing a value of k_h^* , we obtain r_h from Eq. 2.14 and k_h ($k_h = k_h^*/122.794$). Then we calculate the U_{LJ} , U_{Coulomb} and U_{harmonic} as functions of r and use them as inputs for the integration of Eq. 2.17. In this numerical integration we apply the composite Simpson [23] rule with r ranging from 0 – 15.000 Å and the grid width, Δr , 0.02 Å. In our observation, increasing the range of r or reducing the grid width, Δr , does not change the \bar{r} resulted significantly.

Table 2.5 and Fig. 2.4 give the relationship between k_h^* , r_h , and \bar{r} as the result of the integration of Eq. 2.17. We intend to obtain appropriate parameters for the equilibrium monomer-monomer distance 20.4 Å. Table 2.5 shows that using $k_h^* \geq 2500$ all yield the average monomer distance 20.4 Å. Thus in our simulation we choose the minimum $k_h^* = 2500$ and the corresponding $r_h = 20.341796$ Å (from Eq. 2.14). The minimum value is chosen so as to avoid computational singularities and overflows that would result from higher values. Energy non-conservation would also result since we are numerically integrating the dynamical equations. To confirm the correctness of the chosen k_h^* and r_h , in Fig. 2.5 we give the average contour length (Fig. 2.5.a) and bond length (Fig. 2.5.b) of polyelectrolyte chain from some simulations. The contour length of a polymer chain is

k_h^*	r_h	\bar{r}
0.5	-270.622	20.834
1.0	-125.111	20.867
5.0	-8.702	21.010
10.0	5.849	21.034
50.0	17.49	20.781
100.0	18.945	20.652
250.0	19.818	20.522
500.0	20.109	20.459
750.0	20.206	20.4352
1000.0	20.254	20.423
2500.0	20.342	20.405
5000.0	20.371	20.401
7500.0	20.381	20.40
10000.0	20.385	20.40
25000.0	20.394	20.40
50000.0	20.397	20.40
75000.0	20.398	20.40
100000.0	20.399	20.40

Table 2.5: k_h^* , r_h and \bar{r} relationship obtained from integration of Eq. 2.17.

the length at when the chain is set linearly without imposing a strain force on the system. Thus contour length is the sum of all monomer-monomer bond distances or the length of a straight DNA. We compare the contour length from simulations with the theoretical contour length $359 \times 20.4 = 7323.6 \text{ \AA}$. From Fig. 2.5, the average contour length and the bond length data from our simulations are in good agreement to the theoretical values. The error estimations of the average values from simulations are below 0.1 % from the theoretical expectation. Thus the chosen values for k_h^* and r_h are reasonable with respect to experiment. In the simulation box, the extra salt screen the electrostatic interaction leading to a slight departures to the predicted contour length.

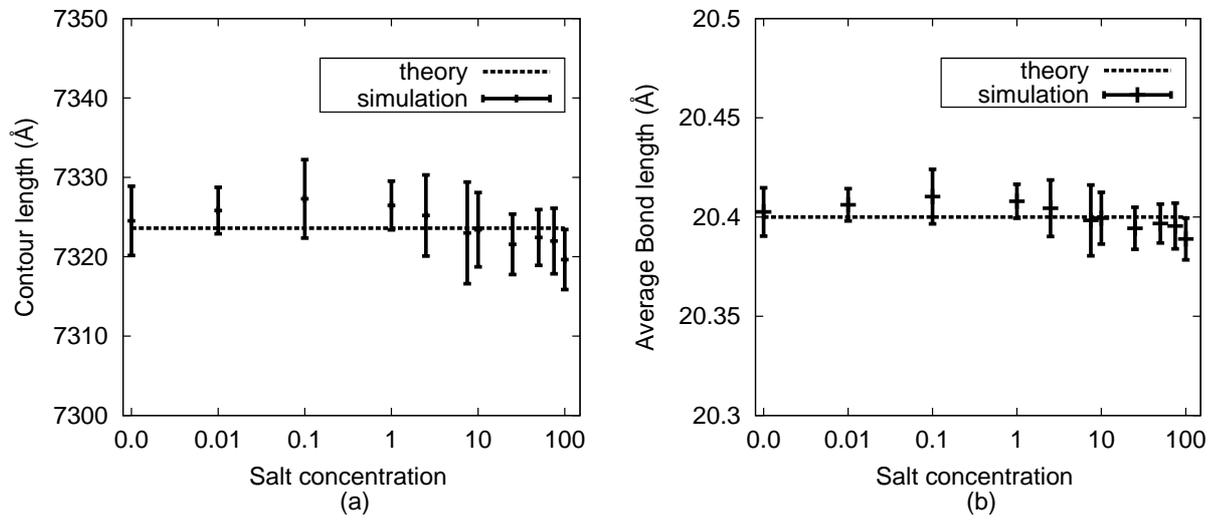


Figure 2.5: Average contour length (a) and bond length (b), both with two standard deviation error bars, from simulations using the harmonic constant 2500 (in reduced unit) and equilibrium bond length set at 20.3418 Å. Simulation results are compared to the theoretical values.

Chapter 3

DNA FLEXIBILITY AND PERSISTENCE LENGTH

3.1 Definition and Calculation of Persistence Length L_p

Persistence length is the variable to describe the stiffness of a polymer chain. The persistence length is the average projection of the end-to-end tangent vectors of polymer chain at a distance s from the initial vector for a chain of an arbitrarily length (sometimes defined in the limit of infinite chain length). Evans and Wennerstrom [24] defined the persistence length as the length over which two parts of the chain keep their orientational correlation and Cifra [25] referred the persistence length as the distance over which the direction of the chain persists.

3.1.1 Calculating Persistence Length with Cosine Correlation Function

The persistence length L_p can be determined from the following equation:

$$\langle \cos \theta(s) \rangle = e^{-s/L_p}, \quad (3.1)$$

where s is the segment length of the chain as depicted in Fig. 3.1. The segment points can be taken at any point along the chain. The angle θ is the result of the orientational difference between two end points of any segments with length s . To find the persistence

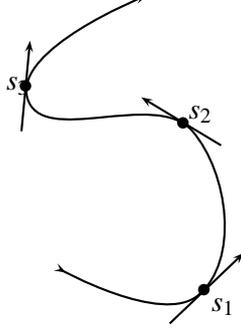


Figure 3.1: A continuum chain showing some points that determine chain segments. The arrows indicate the orientation at each point segment.

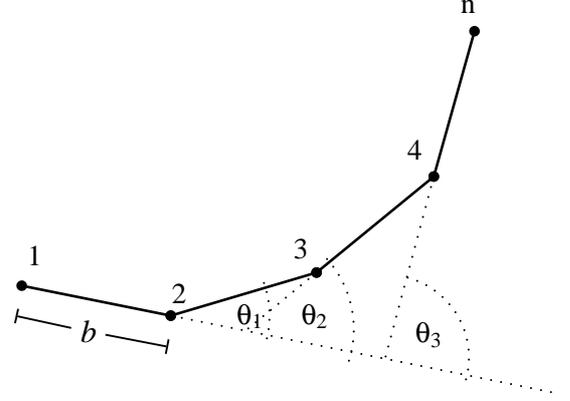


Figure 3.2: A discrete chain. The angle θ_1 , θ_2 and θ_3 are formed by two ends of segments with lengths b , $2b$ and $3b$.

length of a discrete polymer model with bond length b , Eq. 3.1 can be modified as follows:

$$\langle \cos \theta(i, i+k) \rangle = e^{-kb/L_P}, \quad (3.2)$$

where θ is formed by the orientation difference of i^{th} and the $(i+k)^{th}$ monomer. Fig. 3.2 explains how Eq. 3.2 is used for a polymer with n total monomers. The segment length s is equal to kb . For $k = 1, 2$ and 3 with respect to monomer 1, the segment length increase as $b, 2b$ and $3b$. Each segment length s will generate a certain θ . To get the average $\cos \theta$ from Eq. 3.2, the $\cos \theta$ for a particular segment length s is used for all possible segments for the entire polymer. To give a practical example, Eq. 3.2 can be written as:

$$|\cos \theta(i, i+k)| = |\cos \theta(s)| = \left| \frac{\vec{\mathbf{u}}_i \cdot \vec{\mathbf{u}}_{i+k}}{|\mathbf{u}_i| |\mathbf{u}_{i+k}|} \right| = \left| \frac{\vec{\mathbf{u}}_i \cdot \vec{\mathbf{u}}_{i+k}}{s^2} \right| = e^{-s/L_P} \quad (3.3)$$

or

$$\ln |\langle \cos \theta(s) \rangle| = -\frac{1}{L_P} s. \quad (3.4)$$

The reason for putting the absolute marks in Eq. 3.3 because the tangent of two end points of segment is modeled by a positive exponential form (Eq. 3.1). To obtain the persistence length, the $\cos \theta(s)$ is averaged from the 1st monomer to the $(n - \frac{s}{b} + 1)^{th}$ monomer. This average is plotted versus s . From Eq. 3.4, the slope of the $\ln |\langle \cos \theta(s) \rangle|$ vs. s is equal to $-1/L_P$.

3.2 The Relationship between Persistence Length, Bending Modulus (B) and Angle Force Constant (k_θ)

3.2.1 Neutral Polymer Persistence Length

This subsection will give the relationship between persistence length L_P and bending stiffness B as described by [26]. Recall for a single chain with total length L , any segment of length s will have the two end points, and the tangent vectors will make an angle θ ; we shall ensemble average θ by considering all possible length segment s . Then for $s \ll L_P$, θ would be exceedingly small, Then the following:

$$\langle \cos \theta(s) \rangle \approx 1 - s/L_P, \quad (3.5)$$

and for $\theta \rightarrow 0$, $\cos \theta(s) \approx 1 - \theta^2(s)/2$, which from Eq. 3.5 leads to

$$\langle \theta^2(s) \rangle \approx 2s/L_P. \quad (3.6)$$

Since Eq. 3.6 is obtained from the assumption that θ is close to zero, this relationship holds for only stiff chain such as DNA. The total elastic bending energy of a segment with length s is:

$$\Delta E = \frac{1}{2} s B \rho^2, \quad (3.7)$$

where ρ is the curvature which is the inverse of the radius of curvature. B is the bending modulus/stiffness of the polymer. The energy written in Eq. 3.7 is a local variable, where $\Delta E/s = 1/2B\rho^2$ is the energy per unit length of a segment with length s and radius of curvature $1/\rho$. Since $\rho = \theta/s$, then

$$\Delta E = \frac{1}{2}sB \left(\frac{\theta}{s}\right)^2 = \frac{1}{2s}B\theta^2. \quad (3.8)$$

The angle θ is formed by the intersection of the tangents of the ends of the curve with length s . Then put the elastic bending energy in the Boltzmann integration to obtain the average curve. In order to achieve the standard result, Grosberg and Kholkhlov [26] used the standard Boltzmann averaging, but introduced a factor of 2 which does not appear in the denominator partition function because "2 independent planes" (page 8, [26]). Their derivation reads:

$$\langle \theta^2(s) \rangle = \frac{2 \int_0^\pi e^{-\frac{\Delta E}{k_B T}} \theta^2 d\theta}{\int_0^\pi e^{-\frac{\Delta E}{k_B T}} d\theta} = \frac{2 \int_0^\pi e^{-\frac{B\theta^2/2s}{k_B T}} \theta^2 d\theta}{\int_0^\pi e^{-\frac{B\theta^2/2s}{k_B T}} d\theta} = \frac{2sk_B T}{B} \quad (3.9)$$

for $s \ll B/k_B T$. Eq. 3.6 and 3.9 results in the following:

$$\langle \theta^2(s) \rangle = \frac{2s}{L_P} = \frac{2sk_B T}{B},$$

$$L_P = \frac{B}{k_B T} \text{ (standard result)}. \quad (3.10)$$

The Eq. 3.10 is the well-known equation in the worm-like chain (WLC) theory relating the persistence length and bending stiffness [27]. The $L_P - B$ relationship given by Eq. 3.10 suggests that the value of L_P is only a function of B for fixed temperature.

To test the reliability of Eq. 3.10, we have done a simulation for a single neutral polymer having bending stiffness B (Section 3.6). There is no salt and the polymer chain is uncharged. The parameter B is included into the parameter k_θ in the bending energy force fields (see Chapter 2, Eq. 2.2) in the following manner.

$$\Delta E = \frac{B}{2s}\theta^2 = \frac{1}{2}k_\theta\theta^2, \quad (3.11)$$

$$k_\theta = \frac{B}{s}. \quad (3.12)$$

Because k_θ is a parameter for the monomer-monomer bond, the bending potential energy from Eq. 3.11 is a local variable. Then the segment length s approximately equals the monomer-monomer distance b ($s \approx b$) if θ close to zero. Assume θ is very small for stiff chain, then by utilizing Eq. 3.10 we obtain

$$k_\theta = \frac{B}{b} = \frac{L_P}{b}k_B T. \quad (3.13)$$

If we expect a DNA chain has a certain persistence length (L_P), we employ Eq. 3.13 to obtain the k_θ parameter for simulation.

3.2.2 Polyelectrolyte Chain Persistence Length

The $L_P - B$ relationship derived for a neutral polymer cannot in general be used for polyelectrolyte chains without modification. The uniform charge along the polyelectrolyte chain induce repulsive forces among the charged point monomers which results in a larger persistence length. In this section we review some polyelectrolyte persistence length theories before presenting our own ideas.

3.2.2.1 Odijk-Skolnick-Fixman (OSF) Theory

The idea behind OSF theory is to determine the electrostatic persistence length by calculating the energy difference between circular and rodlike conformation. This is accomplished by adding this difference to the bending energy contribution (Odijk [28] and Skolnick and Fixman [29] in [30]). Eq. 3.14 represents the electrostatic energy difference of two configurations.

$$\frac{\Delta U_{\text{electrostatic}}(\theta)}{k_B T} = l_B q^2 \sum_{n=1}^{\infty} \left(\frac{e^{-\kappa r(n)}}{r(n)} - \frac{e^{-\kappa bn}}{bn} \right) \approx \frac{l_B q^2}{8\kappa^2 b^3} \theta^2, \quad (3.14)$$

where l_B is the Bjerrum length, q is the monomers valence, κ is the inverse of the Debye screening length and $r(n)$ is the straight distance between 2 monomers center separated by n bonds in the circular conformation ($r(n) \approx bn(1 - n^2\theta^2/24)$ for $\theta \ll 1$). We can derive the total energy change ΔU_{bond} as

$$\frac{\Delta U_{bond}}{k_B T} \approx \frac{\Delta U_{bending}(\theta)}{k_B T} + \frac{\Delta U_{electrostatic}(\theta)}{k_B T} \quad (3.15)$$

$$= \frac{B\theta^2}{2b} + \frac{l_B q^2 \theta^2}{8\kappa^2 b^3} = \frac{1}{2} \left(B + \frac{l_B q^2}{4\kappa^2 b^2} \right) \frac{\theta^2}{b}. \quad (3.16)$$

In OSF theory, it is assumed that the persistence length L_P can be identified with k_θ (i.e. $L_P = \frac{k_\theta b}{k_B T}$). By comparing Eq. 3.13 and 3.14 we deduce that $k_\theta = \left(B + \frac{l_B q^2}{4\kappa^2 b^2} \right) \frac{1}{b}$. Hence from 3.13 we derive

$$L_P = L_P^0 + L_{OSF} = \frac{B + B_{OSF}}{k_B T} \approx \left(B + \frac{l_B q^2}{4\kappa^2 b} \right) / k_B T, \quad (3.17)$$

where $L_{OSF} = \frac{l_B q^2}{4\kappa^2 b^2} \times \frac{1}{k_B T}$.

The term L_P^0 is defined as the persistence length of the uncharged polymer and L_{OSF} is the electrostatic persistence length. Clearly, OSF theory predicts a linear contribution to the persistence length for charge interactions.

3.2.2.2 Dobrynin Theory

Experimental determination of persistence length showed a quadratic dependence to r_D at relatively low salt concentration (when r_D is large) and a linear dependence at high salt concentration (when r_D is small). On the other hand, Eq. 3.17 shows that it is in agreement with experiment only for low salt concentrations. For higher salt concentration, Dobrynin [30] has attempted to modify the original OSF theory by including torsional terms in the chain deformation energy. In the original OSF theory, the chain deformation energy is a function of the bending angle θ with no torsion angle contribution. The derivation of [30] results in the κ^{-1} dependence of the electrostatic persistence length (L_{WLC}), where

$$L_P = L_P^0 + L_{WLC} = \frac{B + B_{WLC}}{k_B T} = \left(B + \frac{0.32 l_B q^2}{\kappa b} \right) / k_B T \quad (3.18)$$

3.2.2.3 Manning Theory

Manning [31] proposed a method to calculate DNA persistence length by relating the DNA persistence length (L_P) and the persistence length of null isomer DNA (L_P^*), a hypothetical structure where DNA phosphate groups are not ionized. The result of this theory is in contradiction to the additive relationship (Eq. 3.17) of the OSF theory. The relationship between L_P and L_P^* from Manning's theory is given by:

$$L_P = \left(\frac{\pi}{2}\right)^{2/3} R^{4/3} (L_P^*)^{2/3} Z^{-2} l_B^{-1} \left[(2Z\xi - 1) \frac{\kappa b e^{-\kappa b}}{1 - e^{-\kappa b}} - 1 - \ln(1 - e^{-\kappa b}) \right] \quad (3.19)$$

where b is the charge spacing of DNA, R the radius of DNA (assumed cylinder), ξ the charge density ($\xi = l_b/b$), z the counterion charge and $1/\kappa$ Debye screening length which characterizes the electrostatic strength of the salt solution. Generally the parameters stated above is adapted to the typical structure of B-DNA [31] in salt-water solution ($b = 1.7 \text{ \AA}$; $R = 20 \text{ \AA}$; $l_B = 7.13 \text{ \AA}$; $\xi = 7.13/1.7 = 4.2$; $Z = +1$ for Na^+). Manning [31] proposed the value of L_P^* is equal to 74 \AA based on the experimental persistence length data at 0.1 M NaCl (550 \AA).

3.3 A New Derivation of Polyelectrolyte Persistence Length

For the following we propose a different scheme for polyelectrolyte persistence length calculation. Before we proceed to our derivation, we first discuss the previous three theories.

Eq. 3.9 is the "foundation" for equations relating the persistence length L_P to the bending modulus B (Eq. 3.10). Any L_P derivation starting with an inappropriate application of Eq. 3.9 results in unrealistic values for the persistence length. In the following, we emphasize three rules that should be obeyed when using Eq. 3.9.

- (A) Since the energy difference ΔE in the Boltzmann probability denotes the total energy difference, the electrostatic energy changes must be included for a charged polymer.
- (B) The average $\langle \theta^2(s) \rangle$ is a quantity for a continuous segment, not a discretely bonded monomer or a single point charge within a segment. This can be seen from Eq. 3.7

which utilizes the curvature implying a continuous line.

(C) Eq. 3.9 cannot become Eq. 3.10 if the criteria $s \ll B/k_B T$ is not obeyed. Fig. 3.3 shows the value of L_P calculated by Eq. 3.10 and 3.9 at different ratios of $(B/k_B T)$ over s . The figure shows deviation of the equality $L_P = B/k_B T$ (Eq. 3.10) when $B/k_B T \leq s$.

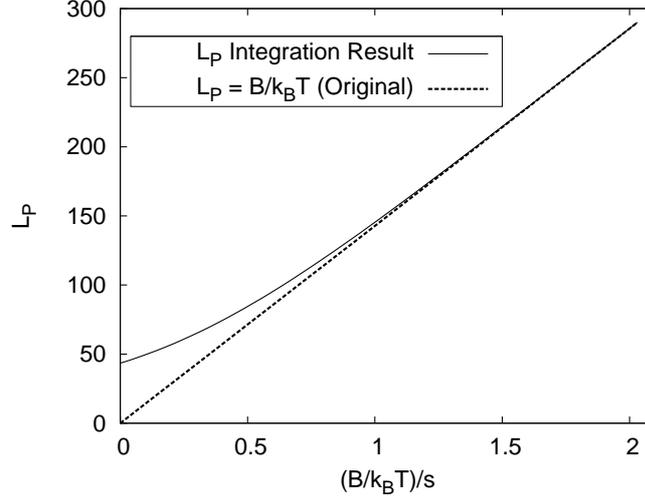


Figure 3.3: L_P obtained by Eq. 3.10 (dashed line) and 3.9 (solid line) at different ratios of $(B/k_B T)$ over s ; s is fixed at $7 \times 20.4 = 142.8 \text{ \AA}$

In the following, some remarks on how previous theories applied Eq. 3.9 and their conformity to the three rules above are given. Then the differences to our derivation are pointed.

OSF theory :

1. There is only a single energy summation in Eq. 3.14, because OSF theory calculates the delta conformational energy per monomer bond. We are apprehensive at this expression since they neglected the importance of averaging in term of the segment conformation. This apprehension arises because the foundational model of persistence length (Eq. 3.1 and 3.9) requires the ensemble averaging over different segment lengths. Since the OSF theory only calculates the energy per monomer bond (ΔE_{bond}), it implies that the energy changes per segment ($\Delta E_{\text{segment}}$) (where the segment has $n + 1$ monomers)

$$\Delta E_{\text{segment}} = \Delta E_{\text{bond}} \times (n + 1) . \quad (3.20)$$

This expression is incorrect because it does double counting of the electrostatic energy calculation (cf. Eq. 3.21). Thus we infer that the OSF energy expression does not represent a characteristic segment which breaks rule (B).

2. The summation limit in Eq. 3.14 for n infinity. Since the segment length $s = nb$, then s has limit infinity. This summation automatically breaks rule (C) where $s \ll B/k_B T$.

Dobrynin theory : Its basis is the OSF theory, thus it retains the characteristics of the OSF theory. They add additional feature which is the torsional term. We comment that they use the torsional angle ϕ to model the torsional degeneracy at a certain bending angle θ . It is probably another alternative if we constructed a degeneracy based on the continuous segment conformation (which is not developed here).

Manning theory : Manning used a different approach compared to the previous two. In Manning, he retained Eq. 3.1 and 3.9 for a neutral polymer. Then he associated the electrostatic extension force between charged monomers by using the force defined from the counterion condensation theory. He defined the "null isomer" as the neutralized polyelectrolyte, where the extension force is balanced by a compressive force. He used elasticity theories to model the compression force.

In our derivation : We start from Eq. 3.21 and 3.24 to satisfy rule (A) and use Eq. 3.22 to conform to rule (B). Our derivation complies at every stage to rule (C).

By including the Coulomb energy in the total chain energy, we can rewrite the chain deformation energy for a single polyelectrolyte chain, where the salt effect in the polyelectrolyte system is included in the κ value, where $\kappa = 1/r_D$ and r_D is the Debye length [32].

$$\Delta E = \frac{B\theta^2}{2s} + \sum_{i=1}^n \sum_{j=i+1}^{n+1} \frac{l_B k_B T q_i q_j \exp(-\kappa r_{ij})}{r_{ij}}. \quad (3.21)$$

By assuming the polyelectrolyte segment bends in circular conformation (Fig. 3.4), we obtain

$$r_{ij} = \frac{s}{\theta} \sqrt{2 \left(1 - \cos \left[\left(\frac{\theta}{n} \right) (j-i) \right] \right)} = \frac{nb}{\theta} \sqrt{2 \left(1 - \cos \left[\left(\frac{\theta}{n} \right) (j-i) \right] \right)}. \quad (3.22)$$

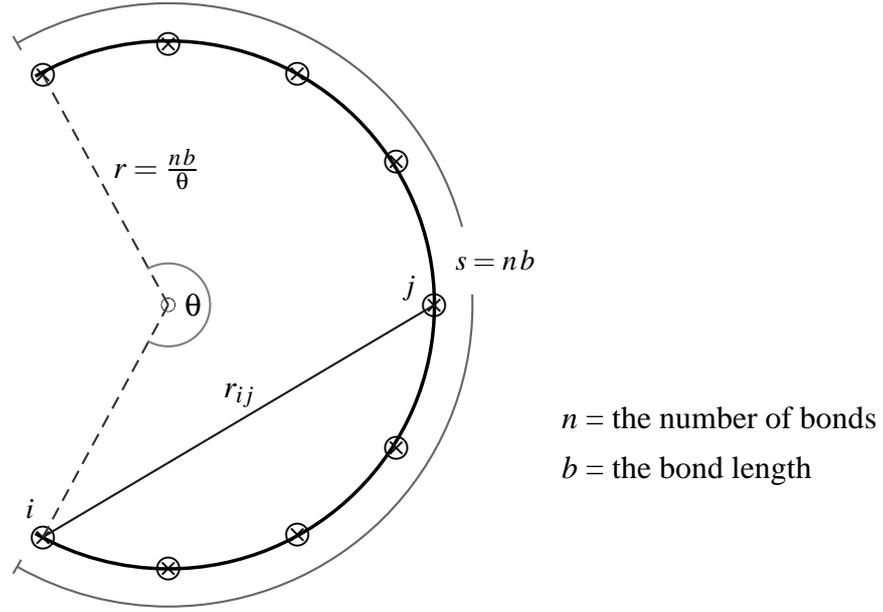


Figure 3.4: Illustration of a polyelectrolyte segment with length s which bends in a circular form. The r_{ij} distance can be determined by the Cosines law.

Define $r_{ij} = nbG$, with

$$G(n, \theta, i, j) = \frac{1}{\theta} \sqrt{2 \left(1 - \cos \left[\left(\frac{\theta}{n} \right) (j - i) \right] \right)}, \quad (3.23)$$

where both i and j denote two individual point charges (monomers) within the polyelectrolyte with charge q_i and q_j respectively separated by a distance $|r_i - r_j|$. The B and s are the chain bending modulus and the segment length respectively. The n is the number of bonds within the segment length s . For the calculation above, the short range repulsive term (such as modeled by the Lennard Jones potential) need not be included in the chain deformation energy. This is because the distance between pairs are larger than the repulsive short range interaction distance. Further, the probability for end to end chain interactions are infinitely small. We rewrite the Boltzmann integration of Eq. 3.9 by

including the coulomb energy term in the Grosberg averaging method [26].

$$\langle \theta^2 \rangle = \frac{2 \int_0^{2\pi} e^{-\frac{B\theta^2/2s}{k_B T} - \frac{\sum_{i=1}^n \sum_{j=i+1}^{n+1} \frac{l_B k_B T q_i q_j \exp(-\kappa n b G)}{s G}}{k_B T}} \theta^2 d\theta}{\int_0^{2\pi} e^{-\frac{B\theta^2/2s}{k_B T} - \frac{\sum_{i=1}^n \sum_{j=i+1}^{n+1} \frac{l_B k_B T q_i q_j \exp(-\kappa n b G)}{s G}}{k_B T}} d\theta} \quad (3.24)$$

$$\langle \theta^2 \rangle = \frac{2 \int_0^{2\pi} e^{-\frac{\left(B + \sum_{i=1}^n \sum_{j=i+1}^{n+1} \frac{2l_B k_B T q_i q_j \exp(-\kappa n b G)}{G\theta^2} \right) \theta^2}{2s k_B T}} \theta^2 d\theta}{\int_0^{2\pi} e^{-\frac{\left(B + \sum_{i=1}^n \sum_{j=i+1}^{n+1} \frac{2l_B k_B T q_i q_j \exp(-\kappa n b G)}{G\theta^2} \right) \theta^2}{2s k_B T}} d\theta}. \quad (3.25)$$

Although not developed here, it seems very likely that other non-equivalent averaging methods are possible other than the Grosberg form. We defer such investigations here and use the Grosberg form as the first approximation. Define B_{new} as:

$$B_{\text{new}} = B + \sum_{i=1}^n \sum_{j=i+1}^{n+1} \frac{2l_B k_B T q_i q_j \exp(-\kappa n b G)}{G\theta^2}. \quad (3.26)$$

Thus analogous to the integration of Eq. 3.9 and the derivation of Eq. 3.10, we can write

$$\langle \theta^2 \rangle = \frac{2 \int_0^{2\pi} e^{-\frac{B_{\text{new}}\theta^2}{2s k_B T}} \theta^2 d\theta}{\int_0^{2\pi} e^{-\frac{B_{\text{new}}\theta^2}{2s k_B T}} d\theta} = \frac{2s k_B T}{B_{\text{new}}},$$

With the substitution of Eq. 3.6, $\langle \theta^2(s) \rangle \approx 2s/L_P$, we derive

$$L_P = \frac{B_{\text{new}}}{k_B T}. \quad (3.27)$$

We propose that the persistence length L_P is the sum of the non-electrostatic persistence length L_P^0 and the additional electrostatic persistence length L_P^{el} . By combining Eq. 3.26 and 3.27, we obtain

$$L_P = \frac{B + B_{\text{el}}}{k_B T} = L_P^0 + L_P^{\text{el}}, \quad (3.28)$$

with

$$L_P^0 = \frac{B}{k_B T} \text{ and} \quad (3.29)$$

$$L_P^{\text{el}} = \frac{B_{\text{el}}}{k_B T} = \sum_{i=1}^n \sum_{j=i+1}^{n+1} \frac{2l_B k_B T q_i q_j \exp(-\kappa n b G)}{G \theta^2} \times \frac{1}{k_B T} . \quad (3.30)$$

Returning to the function $G(n, \theta, i, j)$, for any polyelectrolyte with uniform monomer charge, we can rewrite Eq. 3.30:

$$L_P^{\text{el}} = 2l_B q^2 \sum_{i=1}^n \frac{v_i e^{-\kappa n b G(n, \theta, i)}}{G(n, \theta, i) \theta^2} , \quad (3.31)$$

with $v_i = n - i + 1$.

We have eliminated the independent variable j from G . We have also add a new variable $v_i = v_i(i, n)$ which denotes the Coulomb interaction factor between two point charges with distance ib within a chain segment of length nb . Then the function $G(n, \theta, i)$ is simplified as follows:

$$G(n, \theta, i) = \frac{1}{\theta} \sqrt{2 \left(1 - \cos \left[\left(\frac{\theta}{n} \right) i \right] \right)} .$$

To simplify the $G(n, \theta, i)$ we use the assumption that θ is small ($\theta < 1$) so that $(\theta i/n) \ll 1$. This assumption is reasonable for a stiff polymer where θ is small, and this approximation also accords with the expression in Eq. 3.24 where for large θ , the probability is negligible compared to small θ . Thus with $x = (\frac{\theta i}{n})$, $\cos x \approx 1 - \frac{x^2}{2}$, we have

$$G(n, \theta, i) = \frac{1}{\theta} \sqrt{2 \left(1 - \left[1 - \frac{(\frac{\theta i}{n})^2}{2} \right] \right)} = \frac{1}{\theta} \sqrt{\left(\frac{\theta}{n} i \right)^2}$$

$$G(n, i) = \frac{i}{n} .$$

Eq. 3.31 become:

$$L_P^{\text{el}} = \frac{2l_B q^2}{\theta^2} \sum_{i=1}^n \frac{v_i e^{\kappa b n \frac{i}{n}}}{i/n}$$

$$L_P^{\text{el}} = \frac{2l_B q^2 n}{\theta^2} \sum_{i=1}^n \frac{(n-i+1)}{i} e^{-\kappa b i} . \quad (3.32)$$

It can be shown ([33], symbolic algebra toolbox) that

$$\sum_i^n \frac{n-i+1}{i} e^{-Ai} = e^{-A} \left(\frac{1}{e^{-A}-1} - (n+1)e^A \ln(1-e^{-A}) \right) + \frac{e^{-A(n+2)}}{n+2}$$

$$\left(\frac{(n+2)(-n-1+e^{-A})e^{2A}}{n(e^{-A}-1)} - (n+2)(n+1)e^{2A} S_A(e^{-A}, 1, n) \right) , \quad (3.33)$$

with $S_A(e^{-A}, 1, n) = \sum_{k=0}^{\infty} \frac{e^{-Ak}}{(n+k)}$ [34] and $A = \kappa b$.

From the result (Eq.3.33), the integration of Eq. 3.32 would still yield κb factor in exponential form and the parameters κ and b will not be separated from their multiplication κb form. The variable i (Eq. 3.32) vanishes and a new variable k is introduced in the $S_A(e^{-A}, 1, n)$ formula (Eq. 3.33). However, the variable k in the $S_A(e^{-A}, 1, n)$ has fixed values (from 0 to ∞) and independents from any other variables. Thus we assume it is the simplest and most reliable way to simplify the scale of electrostatic persistence length L_P^{el} by the following equation:

$$L_P^{\text{el}} \sim 2l_B q^2 n \langle \langle f_1(n) \rangle \rangle_b e^{-\kappa b \langle f_2(n) \rangle_b} \theta . \quad (3.34)$$

Notice that we keep the κb multiplication in the exponential factor. In the following we concentrate in obtaining the $f_1(n)$ and $f_2(n)$ by curve fitting procedures. We note that Eq. 3.34 is only an approximation to Eq. 3.32. In this approximation, f_1 and f_2 are fitting parameters to yield the best approximation to Eq. 3.34. We use non-linear least squares to derive the results. The data for fitting are obtained by numerical integration of Eq. 3.24. First we perform the numerical integration to test the relationship between L_P^0 and L_P as expressed in Eq. 3.28. Fig 3.5 depicts the numerical integration data, which proves the linear dependence of L_P^0 to L_P . The electrostatic persistence length L_P^{el} is calculated over

different Debye length.

Fig 3.6 (a) and (b) are the numerical integration data showing the linear dependence of

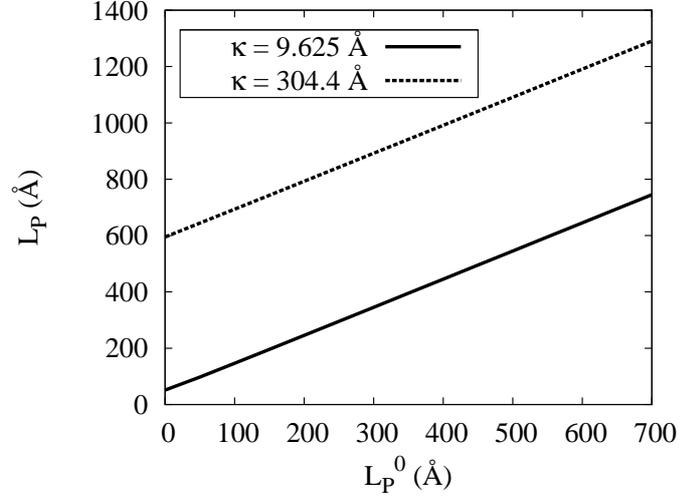


Figure 3.5: The L_P and L_P^0 relationship (in two different Debye length κ^{-1}) obtained from numerical integration of Eq. 3.24. This figure confirms the $L_P - L_P^0$ relationship in Eq. 3.28 ($L_P = L_P^0 + L_P^{\text{el}}$). Each κ value correspond to a definite L_P^{el} . For all κ values, a linear dependency is observed but only two values are depicted.

the electrostatic persistence length (L_P^{el}) to l_B and the quadratic dependence of L_P^{el} on q . This data are shown to confirm the dependence of (L_P^{el}) to l_B and q in Eq.3.32 at any fixed Debye length.

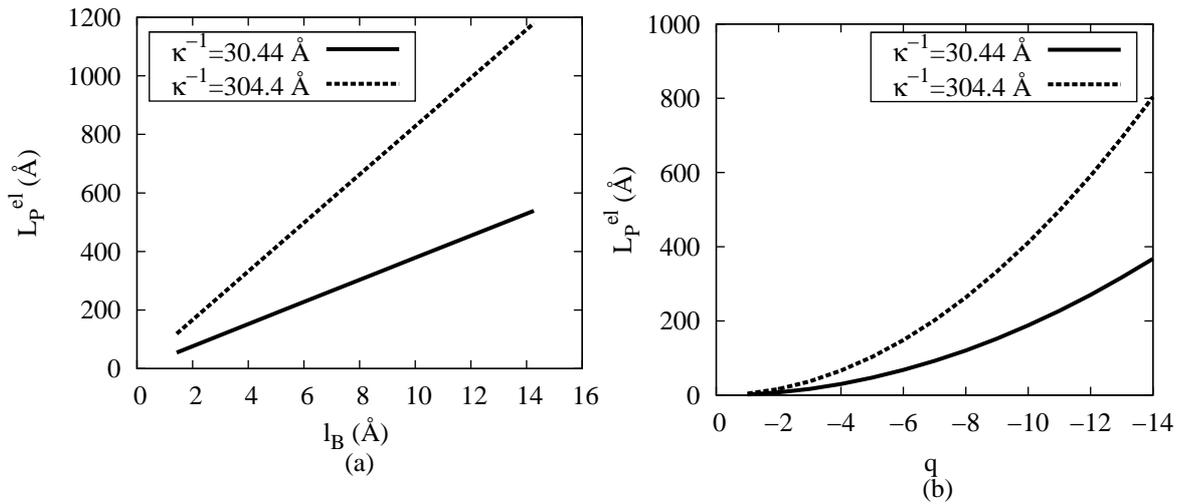


Figure 3.6: The L_P^{el} linear relationship to the Bjerrum length l_B (Fig. a) and quadratic relationship to the monomer charge q (Fig. b) obtained from numerical integration of Eq. 3.24 (for different Debye lengths κ^{-1}). These figures confirm the $L_P^{\text{el}} - l_B$ and $L_P^{\text{el}} - q$ relationship in Eq. 3.34

3.3.1 Details of the Fitting Procedures

The curve fitting for obtaining $f_1(n)$ and $f_2(n)$ are performed by using the *LAB FIT*¹ software with *Levenberg-Marquardt* algorithm (LMA)[35, 36]. The LMA has become a standard method for fitting nonlinear least-squares functions[37, 38].

We simplify the expression in Eq. 3.34 for the fitting procedure. Because q and l_B are independent of the $f_1(n)$ and $f_2(n)$ function, we may choose arbitrarily q and l_B . We choose $q=1$ and $l_B = 1/2n$. Thus Eq. 3.34 becomes

$$L_P^{\text{el}}(q = 1, l_B = 1/2n, \kappa, b, n) \sim \langle \langle f_1(n) \rangle_b e^{-\kappa b \langle f_2(n) \rangle_b} \rangle_\theta . \quad (3.35)$$

First we perform numerical integration of Eq. 3.24 to obtain $\langle \theta^2 \rangle$ as a function of κ with range within $1/5 \leq \kappa \leq 1/2450$, which corresponds to salt concentration range 400 mM - 0.0015 mM. Then we use Eq. 3.6 to obtain L_P and Eq. 3.28 to get L_P^{el} ($L_P^{\text{el}} = L_P - L_P^0$). $L_P^0 = B/k_B T$ is a chosen parameter in numerical integration of Eq. 3.24, where we notice as long $L_P^0 \ll b$, the final L_P^{el} obtained does not change. In our observation, extending the κ range did not give any significant change to the final result. The κ range was divided into 490 intervals which corresponds to the maximum number of points that can be fitted for the software used. This integration is repeated for different n ($1 \leq n \leq 25$). At this point, we graph L_P^{el} vs. κb at different n and fit these graphs to the function

$$L_P^{\text{el}} = A_i \exp(\kappa b B_i). \quad (3.36)$$

We then obtain the parameters A_i and B_i for each n . The sets of A_i and B_i are used to obtain $f_1(n)$ and $f_2(n)$ (Eq. 3.35).

Finally, each integration with the fixed n is repeated for different bond lengths b . The bond length b values include 1.7, 3.4, 5.1, 10.2, 15.3, and 20.4 Å. With the integration at different b , we can determine $\langle f_2(n) \rangle_b$ by analyzing the $f_2(n) - b$ relationship. Figure 3.7 gives a sample of the numerical integration (Eq. 3.24) and fitting result (Eq. 3.36).

¹<http://www.angelfire.com/rnb/labfit/>

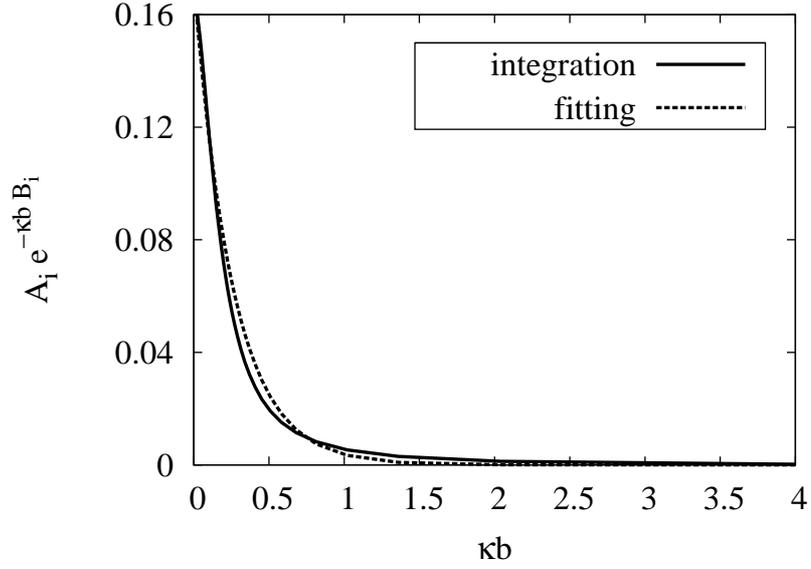


Figure 3.7: Numerical integration of Eq. 3.35 and its fitting 3.36, with parameters $n=20$, $b= 20.4 \text{ \AA}$; κ is varied between $1/5$ and $1/2450 \text{ \AA}^{-1}$.

3.3.2 Obtaining $f_1(n)$

In order to test the dependency, we first determine the $f_1(n)$ function. This function is not the same as $\langle f_1(n, b) \rangle$, but related to it. We wish to derive $f_1(n)$ as a function *not* dependent on b ; $f_1(n)$ is defined in 3.34. In order to confirm non-dependency on the b factor, we plot A_i (3.36) where A_i is obtained from the fitting of Eq. 3.24 to the function L_p^{el} in Eq. 3.36. (In other words, for each n and fixed b , we can determine L_p^{el} from which A_i can be obtained from curve-fitting). Then we repeat this plotting for different bond lengths b . Eq. 3.36 is a parametric form for Eq. 3.35. It would be convenient if A_i were not strongly dependent on b . In the parametric form, $A_i \equiv f_1(n)$.

Fig. 3.8 shows the $f_1(n) - n$ relationship at different bond lengths b . From the figure, we conclude that the most reliable form for $f_1(n)$ is $f_1(n) = An + B$ with the A and B parameters obtained by curve fitting. Table 3.1 gives the fitting result of $f_1(n) = An + B$. From Fig. 3.8 and Table 3.1, we observe that the change in parameter A and B over different b is not significant. The standard deviation for the parameter A is 3.46% from its mean and the standard deviation of the parameter B is 3.49%. We attributed the slight change to the approximation used in the fitting algorithm. To determine the final function

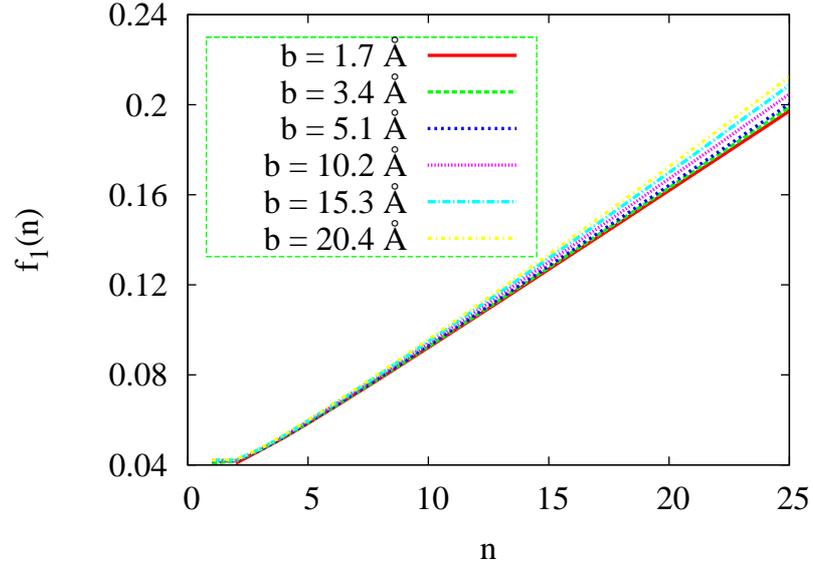


Figure 3.8: $f_1(n)$ (from Eq. 3.35) versus n at different bond length b . These curves will be used to determine the best form of $f_1(n)$.

b (Å)	$f_1(n) = An + B$ (dimensionless)
1.7	$0.0069 n + 0.024$
3.4	$0.0069 n + 0.024$
5.1	$0.0070 n + 0.024$
10.2	$0.0072 n + 0.023$
15.3	$0.0074 n + 0.023$
20.4	$0.0075 n + 0.022$

Table 3.1: Numerical result of fitting the function $f_1(n) = An + B$ in Fig. 3.8. The mean of parameter A is 0.00714, with standard deviation 0.000248. The mean of parameter B is 0.0234 with standard deviation 0.000819

$f_1(n) = An + B$, we take the A and B parameters from their averages resulting in:

$$f_1(n) = 0.0071n + 0.0234 . \quad (3.37)$$

3.3.3 Obtaining $f_2(n)$

The general idea to obtain $f_2(n)$ is the same as with $f_1(n)$. Instead of using A_i parameter in the parametric function Eq. 3.36, the B_i parameter is considered equivalent to the $f_2(n)$ factor in Eq. 3.35, i.e. $B_i \equiv f_2(n)$. We first determine B_i from fitting Eq. 3.24 to Eq.

3.36 at some fixed n . We then plot this set of B_i vs. n . We repeat the above procedure by varying the bond length parameter b . Based on the derivation of Eq. 3.34 which is a fitting model, we expect the $B_i - n (f_2(n) - n)$ relationship to be also independent of b . Figure 3.9 shows the relationship between $f_2(n)$ and n for different b 's. It is found

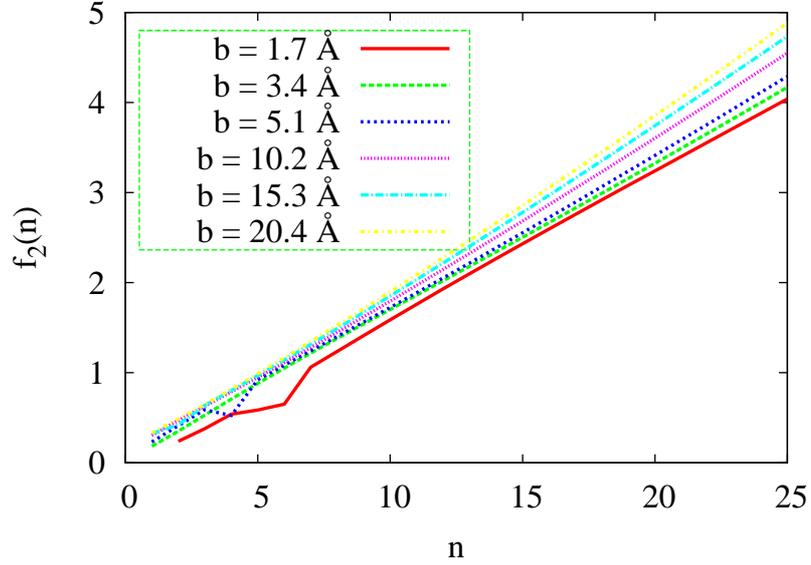


Figure 3.9: $f_2(n)$ (from Eq. 3.35) versus n at different bond lengths b . These curves will be used to determine the best form of $f_2(n)$.

through many fittings that $f_2(n)$ is (i) a linear function of n and (ii) that the gradient of this function exhibits b dependency. On the other hand from the original L_p^{el} result (Eq. 3.32 and 3.33), the b parameter always appears in the κb multiplication term only and not in the $f_2(n)$ term. Thus we attribute the effect of bond length parameter b on a dimensionless modifying function f_2^* to account for the observations. As such each graph may be plotted as a straight line from the origin $(0, 0)$ where we infer that the $f_2(n)$ function is separable and can be written in the form:

$$f_2(n, b) = f_2^*(b) n. \quad (3.38)$$

In the following, we proceed to obtain $f_2^*(b)$. The slope of each line in figure 3.9 (numerically given in table 3.2) represents the dimensionless $f_2^*(b)$. It is shown that the value of b will slightly affect the gradient f_2^* in the exponential scaling $f_2(n, b)$, where the standard deviation of the gradient $f_2^*(b)$ is 6.93% of its mean.

We emphasize that the parameter $f_2^*(b)$ pertains only to scaling b in Å unit. To guarantee

b (Å)	$f_2(n) = f_2^*(b) n$ (dimensionless)
1.7	0.161 n
3.4	0.167 n
5.1	0.171 n
10.2	0.181 n
15.3	0.187 n
20.4	0.193 n

Table 3.2: Numerical result of fitting the function $f_2(n, b) = f_2^* n$ in Fig. 3.9. The mean of the f_b^* parameter is 0.177, with standard deviation 0.0123.

that $f_2^*(b)$ is dimensionless for all length units, we can rewrite $f_2^*(b)$ as

$$f_2^*(b) = f_2^*(b/l_0)$$

where $l_0 = 1$ Å.

We plot the graph $f_2^*(b)$ vs b in figure 3.10 to obtain the scaling of $f_2^*(b)$ over different b , which ranges from 1.7 Å – 20.4 Å. The dots in Fig. 3.10 represent the slope of the

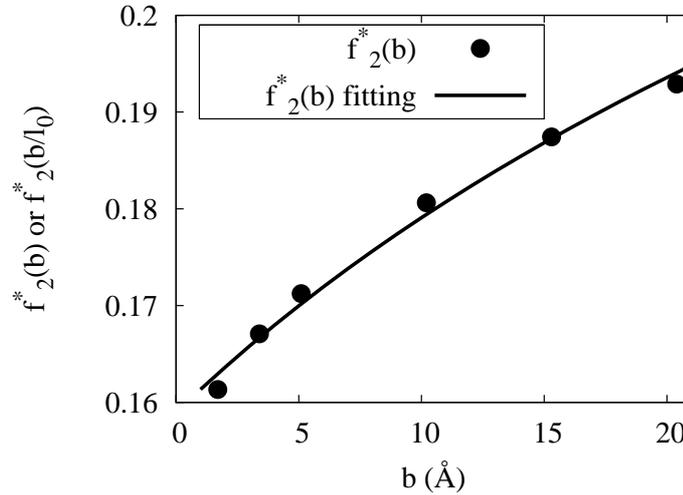


Figure 3.10: The $f_2^*(b)$ and b relationship (dots) and the fitting function (curve)

lines in Fig. 3.9 (and numerically in Table 3.2). The continuous curve in Fig. 3.10 is the best fit according to the L-M algorithm. Concerning the fitted curve of Fig. 3.10, we choose an arbitrary function with the minimum number of parameters; an appropriate form was found to be $f_2^*(b/l_0) = A \ln(b/l_0 + B)$, $A = 0.0521$ and $B = 21.17$ to yield the

final expression as

$$f_2^*(b, l_0 = 1\text{\AA}) = 0.0521 \ln(b/l_0 + 21.17) . \quad (3.39)$$

3.3.4 Final Result of the Fitting Procedure

Returning to the initial expression of the electrostatic persistence length, L_P^{el} , in Eq. 3.34,

$$L_P^{\text{el}} \sim 2l_B q^2 n \langle \langle f_1(n) \rangle_b e^{-\kappa b \langle f_2(n) \rangle_b} \rangle_\theta$$

and the $f_1(n)$ and $f_2(n)$ functions from Eq. 3.37, 3.38 and 3.39 into L_P^{el} , we arrive at the final expression:

$$\begin{aligned} L_P^{\text{el}} &\sim 2l_B q^2 n (0.0071n + 0.0234) e^{-(\kappa b n) 0.052 \ln(b/l_0 + 21.17)} , \\ &= 2l_B q^2 n (0.0071n + 0.0234) e^{\ln(b/l_0 + 21.17) - 0.052 \kappa b n} , \\ &= 2l_B q^2 n (0.0071n + 0.0234) (b/l_0 + 21.17)^{-0.052 \kappa b n} . \end{aligned} \quad (3.40)$$

Since $L_P = L_P^0 + L_P^{\text{el}}$, then

$$L_P = L_P^0 + 2l_B q^2 n (0.0071n + 0.0234) (b/l_0 + 21.17)^{-0.052 \kappa b n} , \quad (3.41)$$

where the $l_0 = 1\text{\AA}$ is the parameter to make b/l_0 dimensionless.

3.4 Discussion on The New Derived Persistence Length

We compare the L_P result obtained from numerical integration (Eq. 3.24) and fitting (Eq. 3.41) in Fig. 3.11. From Fig. 3.11, the fitting function is in good agreement with the numerical integration data. We attribute the slight discrepancy in the fitted-data to the fitting approximations that we made and which was described in detail above.

The bending modulus $B = EI$ is a function of Young's modulus E and the area moment of inertia I , therefore B is not a function of the segment length $s = nb$. Since $L_P = B/k_B T$ holds for small s ($s \ll B/k_B T$ (Eq. 3.5 and 3.9)), thus the variable L_P is independent of

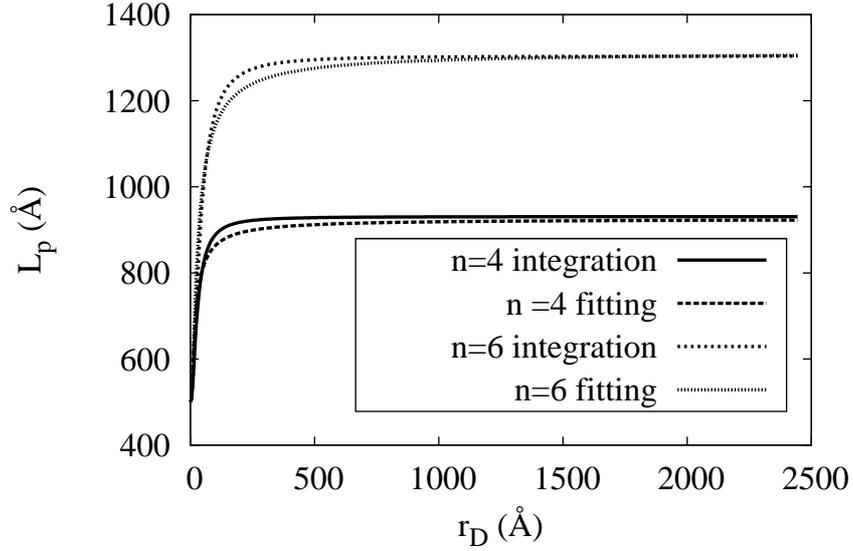


Figure 3.11: Persistence length (L_P) versus Debye screening length $r_D = 1/\kappa$ from numerical integration (Eq. 3.24), compared to the fitting equation (Eq. 3.41). For these curves, $L_P^0 = 500 \text{ \AA}$, $l_B = 7.13 \text{ \AA}$, $q = -12$, $b = 20.4 \text{ \AA}$, and $l_0 = 1 \text{ \AA}$.

the segment length s for small s . The new persistence length equation (3.41) contains the parameter bond number n ($n = s/b$). Thus at small n , L_P should not be a function of n .

Equation 3.6, $\langle \theta^2 \rangle \approx 2nb/L_P$ can be used to determine L_P function not dependent on n .

We can calculate $\langle \theta^2 \rangle$ from

$$\langle \theta^2(n) \rangle \approx 2nb/L_P(n) ,$$

where L_P is obtained from Eq. 3.41. The $\langle \theta^2(n) \rangle$ data are plotted over n . The gradient (m) of the $\langle \theta^2 \rangle - n$ plots equals $2b/L_P$. Thus

$$L_P = 2b/m . \tag{3.42}$$

Our theory was based on the $s = nb$ length being "small", in accordance with the assumption used in framing the conventional polymer equations (see subsection 3.2.1, Eq. 3.5 and 3.9). It is observed that at small nb (small s), we indeed derive a linear curve (Fig. 3.12) and hence it is safe to conclude that we have provided a formulation that includes an electrostatic contribution directly to the $\langle \theta^2 \rangle$ calculation in Eq. 3.9 that other theories did not include or neglect. We caution that this theoretical foundation is based on the Gros-

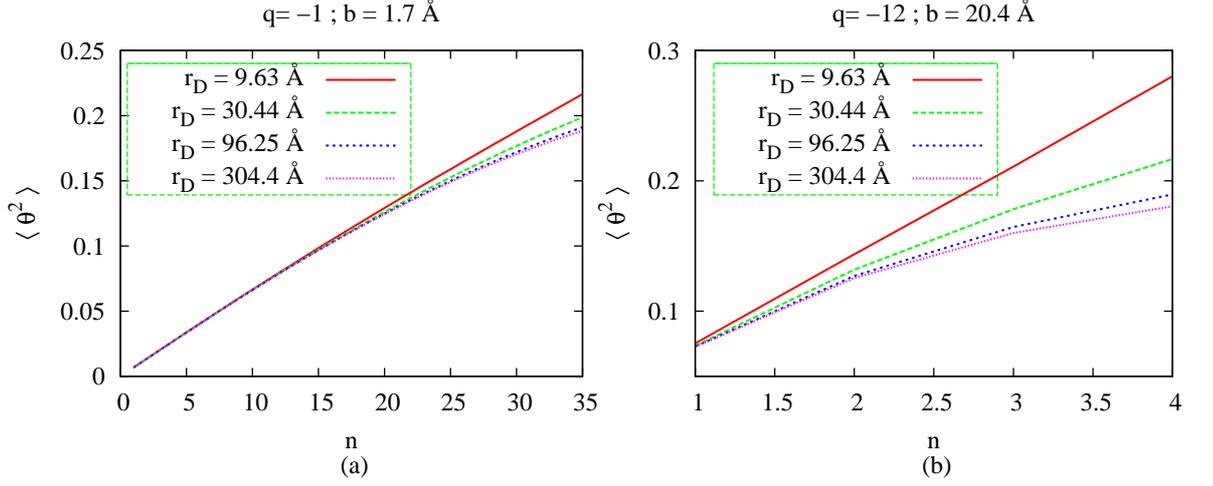


Figure 3.12: $\langle \theta^2 \rangle - n$ relationship based on the Eq. 3.6 ($\langle \theta^2 \rangle \approx 2nb/L_P(n)$). The $L_P(n)$ is calculated by Eq. 3.41. The q and b parameters for Eq. 3.41 are given above each figure. $L_P^0 = 500 \text{ \AA}$, $l_B = 7.13 \text{ \AA}$, and $l_0 = 1 \text{ \AA}$ for all curves.

berg partition function and integration [26]. Another expression (see Scipioni et al. [39] who provides a different interpretation for the virtual segment s) insignificantly modifies our numerical results above where the same conclusion would obtain as for the Grosberg scheme which we shall use in what follows. We intend to pursue our own alternative formulation not based on Grosberg's methodology in the future.

In the experimental work of Scipioni et al. [39], the measurable experimental segment length is half of our s (Fig. 3.4). Define s_{sc} and θ_{sc} as the virtual segment length and angle they defined, where

$$s_{sc} = \frac{1}{2}s \quad \text{and} \quad \theta_{sc} = \frac{1}{2}\theta, \quad (3.43)$$

and where s and θ are quantities in our work. Eq. 3.24 becomes

$$\langle \theta_{sc}^2 \rangle = \frac{2 \int_0^{2\pi} e^{-\frac{B\theta_{sc}^2/2s_{sc}}{k_B T} - \frac{\frac{1}{2} \sum_{i=1}^n \sum_{j=i+1}^{n+1} \frac{l_B k_B T q_i q_j \exp(-\kappa n b G(\theta))}{s G(\theta)}}{k_B T}} \theta_{sc}^2 d\theta}{\int_0^{2\pi} e^{-\frac{B\theta_{sc}^2/2s_{sc}}{k_B T} - \frac{\frac{1}{2} \sum_{i=1}^n \sum_{j=i+1}^{n+1} \frac{l_B k_B T q_i q_j \exp(-\kappa n b G(\theta))}{s G(\theta)}}{k_B T}} d\theta} \quad (3.44)$$

Notice the ensemble averaging is done for the $\langle \theta_{sc}^2 \rangle$ although the integration is done over θ . The calculation of the electrostatic energy term is the same with Eq. 3.24 since there is an additional segment s_{sc} appended at the tail of the central segment s_{sc} . The $1/2$ factor preceding the electrostatic energy term appears because we only calculate the energy of

and (c) for meaningful variables such as the persistence length which is reported in Table 3.3, using equation (3.41).

Table 3.3 shows that models with higher monomer charge, despite the same bond

$r_D /$ $c_{\text{salt}} \text{ mM}$	$b = 1.7 \text{ \AA}; q = -1$			$b = 3.4 \text{ \AA}; q = -2$			$b = 20.4 \text{ \AA}; q = -12$		
	$L_P (\text{\AA})$	$s (\text{\AA})$	r^2	$L_P (\text{\AA})$	$s (\text{\AA})$	r^2	$L_P (\text{\AA})$	$s (\text{\AA})$	r^2
9.63/100.0	554.10	61.2	0.999	559.24	61.2	0.999	601.91	61.2	1.0
30.44/10.0	603.95	61.2	0.995	617.34	61.2	0.995	778.752	61.2	0.996
96.25/1.0	628.03	61.2	0.992	646.05	61.2	0.991	888.91	61.2	0.989
304.4/0.1	636.78	61.2	0.990	656.55	61.2	0.989	933.51	61.2	0.986

Table 3.3: Persistence length determined by Eq. 3.41 at different salt concentrations for three models in Fig. 3.13 where $L_P^0 = 500 \text{ \AA}$, $l_B = 7.13 \text{ \AA}$, and $l_0 = 1 \text{ \AA}$ for models (a), (b) and (c). The $\langle \theta^2(n) \rangle - n$ lines are plotted from $n = b$ to $n = s$ (see Sec. 3.4); r^2 is the correlation coefficient for the plots.

length/charge ratio, have larger persistence lengths. Thus from a theoretical standpoint, our polyelectrolyte model is not accurate enough to represent the real DNA in terms of chain flexibility. However our model is retained for convenience. In the following we justify the convenience of our DNA model.

The closest axial distance between two negatively charged phosphate groups in experimental DNA chain is 1.7 \AA . The DNA chain has diameter 20 \AA (Fig. 1.1). In many simulations, a single particle or ion is usually modeled by a spherical volume. If we want to create a spherical monomer which cover the whole DNA diameter, it means we must create a monomer with radius 10 \AA . This 10 \AA radius monomer of DNA approximately consists 12 phosphate groups (since DNA chain has 12 phosphate groups per 20.4 \AA chain segment length). In the following, other models of the DNA chain are offered, where the pros and cons between possibility and convenience are given.

The particle radius in our simulation is defined by their short-range interaction parameters (i.e. hard-soft sphere radii in the LJ interaction). We assume we can model the DNA monomer with a discrete charge -1 with an adjacent charge distance of 1.7 \AA (Fig. 3.13). Then the sum of hard and soft core radii of the DNA monomer that can interact with mobile ions equals 10 \AA so we can model the experimental DNA chain diameter. Up to now we do not encounter problems. Problems occur when we think about the short-range

interaction between the DNA monomers. If we assign 10 \AA as the sum of DNA hard-soft radii for DNA-DNA LJ interactions, the monomer bonds will automatically break due to force overflow since the monomer bond distance is 1.7 \AA . Otherwise, if we set the sum of hard-soft sphere monomer radii to 0.85 \AA for DNA-DNA LJ interactions to comply with the monomer bond length, it means that we allow non-adjacent monomers to get closer to each other until the closest distance is 1.7 \AA . This close distance is not possible since we have a 20 \AA DNA chain diameter.

The solution is to create a DNA chain model with monomer charge -1 for each 1.7 \AA axial distance by applying a non-spherical LJ potential (i.e. Gay-Berne potential for disk shape) to regulate the short range interaction among the DNA monomers. We have tested this type of potential in the ESPResSo MD simulation package which cannot run this potential for a very long chained polymer, unfortunately. Even with a short chain (i.e. 5 monomers) the simulation required a very small time-step (i.e. $\pm 1 \times 10^{-12}$ in reduced units) which is simply not reliable for equilibration. A long chain model in simulation is preferred to obtain good averaged properties and minimize the non-isotropic effects at the two ends of chain, and this is the reason why our model is convenient.

3.6 Simulation Results

In the following the DNA persistence length data from MD simulation are given. In Fig. 3.14, the data from simulations of neutral polymer are given. These data are used to examine the theoretical prediction of the neutral polymer persistence length. The theoretical persistence length is given by the WLC theory of Eq. 3.10, $L_P = B/k_B T$, where L_P is the persistence length and $B/k_B T$ is chosen to define the bending angle constant parameter (k_θ) in our model [see Eq. 3.13 and Eq. A.17 in the Appendix A]. We simulate a single neutral polymer with the same dimension as our DNA model. We perform simulations in two kinds of time-step (1×10^{-3} and 1×10^{-5} in reduced unit) to confirm that the systems are in equilibrium. In the simulations, the polymer did not experience any type of non-bonding intermolecular interactions (e.g. Lennard-Jones or Coulombic). The harmonic bonding potential preserved the equilibrium bond distance and the bending angle poten-

tial governs the polymer flexibility. The bending modulus B range is $[50-1500] \text{ \AA} k_B T$. The straight lines in Figures 3.14 (a,b) are the expected persistence length from the WLC theory. The simulation L_P are calculated by using Eqs. 3.3 and 3.4 which represent the

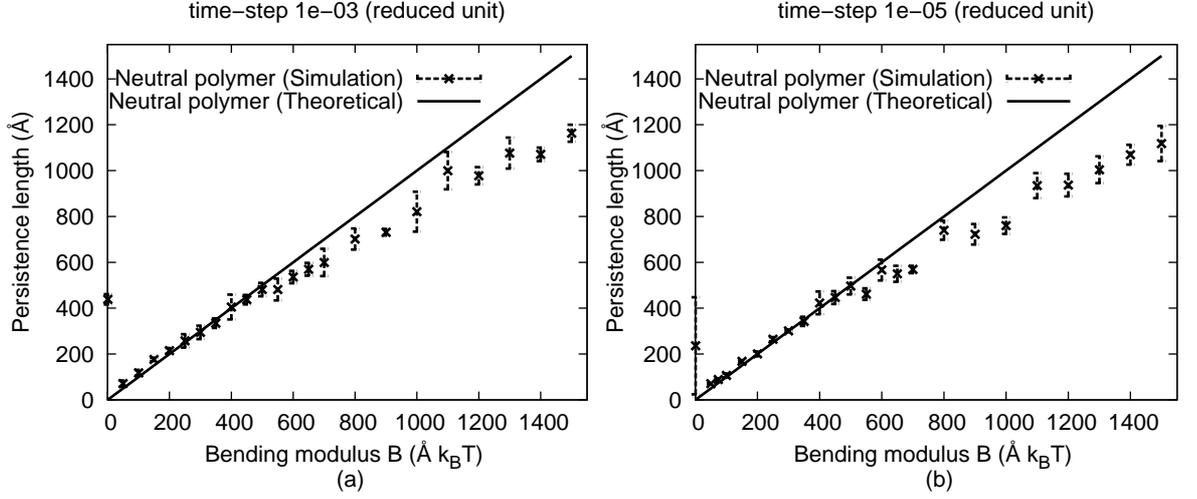


Figure 3.14: Persistence length (L_P) of the neutral polymer possessing the same dimension with the polyelectrolyte DNA models, with 2 standard deviation error bars, at different bending modulus (B) parameters. The straight lines are the theoretical prediction. The simulations are performed with time-step 1×10^{-3} (Fig. (a)) and 1×10^{-5} (Fig. (b)).

natural definition of persistence length. For all persistence length calculations which require a plot between any function $f(s)$ and s , where s is the segment length, we choose the maximum segment length $s = 61.2 \text{ \AA}$. The reason is because from observation, the correlation coefficient r^2 ($0 \leq r^2 \leq 1$) from any gradient determination (Eq. 3.4 for the DNA monomer coordinate in the simulation, verification of Eqs. 3.6 and 3.42 for theoretical calculation) will be less than 0.98 for $s > 61.2 \text{ \AA}$. Another reason is that the theoretical equations for persistence length are derived from the assumption $s \ll L_P$ (Eq. 3.5 and 3.9), and the bond length 20.4 \AA in our models already contributes a relatively large distance to the segment length s .

From the data in Fig. 3.14, the simulation L_P agree with the theoretical L_P when the bending modulus B is within the range $[50-550] \text{ \AA} k_B T$. For $B \geq 600 \text{ \AA} k_B T$, the simulation persistence length deviates from the theoretical predictions. The deviation becomes larger as the bending modulus increases. As far as we know, the maximum B value for which the theoretical L_P is reliable has never been discussed. In what follows, we provide suggestions with regard to this departure. We conjecture that the theoretical WLC equa-

tion fails at large B because in its derivation (Eq. 3.9) the energy contributions from each segment s to the Boltzmann factor are considered independent. In other words, we divide a very long chain into many small segments and assume that the topology of each segment does influence the topology of neighboring segments. We argue that despite polymer stiffness, dynamics caused by a segment will affect neighbors leading to additional bending. That added bending occurs is clearly observed for stiff enough chains.

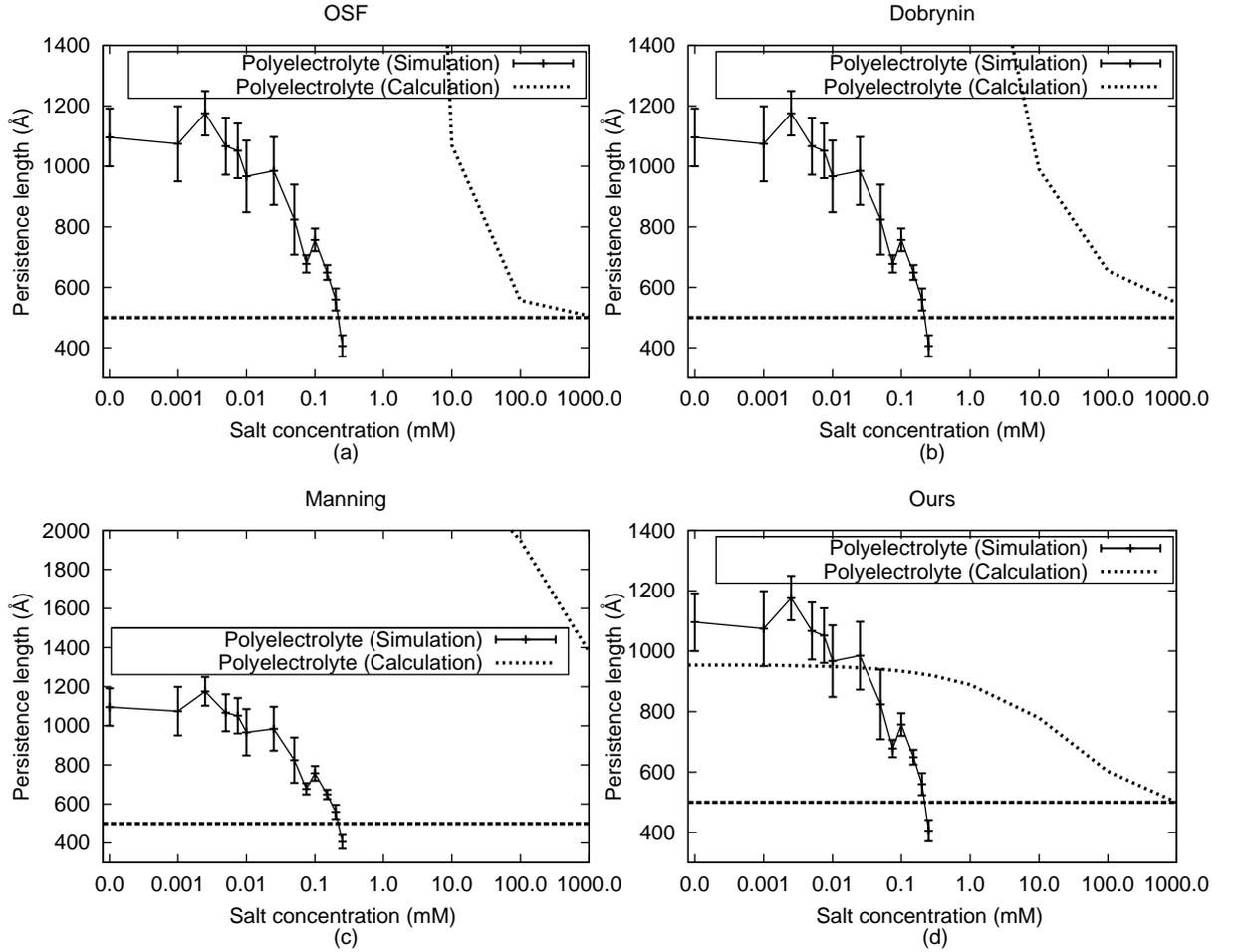


Figure 3.15: Persistence length (L_P) of our DNA polyelectrolyte simulations, with 2 standard deviation error bars, at salt concentration 0.0–0.25 mM. The simulation results are compared to the L_P prediction from OSF, Dobrynin and Manning theories (a, b, c). Comparison with our derivation is given in (d). The uncharged DNA bending modulus $B_0 = 500 \text{ \AA} k_B T$ throughout. The horizontal dashed line is the persistence length of the neutral uncharged DNA ($L_P^0 = 500 \text{ \AA}$) [see Eq. 3.28].

We present the persistence length of the DNA polyelectrolytes at different salt concentrations in Figures 3.15. The highest salt concentration 0.25 mM corresponds to the maximum capability of our computer resources and the MD package used. Before we continue

discussing about persistence length, we want to illustrate the importance of choosing the DNA concentration. The DNA concentrations in our simulations is 0.005 mg/ml which defines the cubic simulation box length 7612.005 Å. Table 3.4 presents the end-to-end distance (R_e) and radii of gyration (R_g) of the DNA in different salt concentrations. R_e is the straight distance between two ends chain. R_g is defined as the mean square distance away from the center of gravity (\mathbf{r}_{cg}), where center of gravity is the average location of the weight of the polymer. R_g can be determined by the following equation

$$R_g^2 = \frac{1}{N} \sum_{k=1}^N (\mathbf{r}_k - \mathbf{r}_{cg})^2, \quad (3.46)$$

where k is the monomer index and N is the number of monomers. Thus R_g is a measure of the size of the chain. Table 3.4 shows that the values of R_e and R_g in equilibrium are much smaller than the cubic box length. Thus we infer that it is safe to neglect the DNA-DNA interactions between the cell neighbors when determining the DNA topology. From sim-

Salt Concentration (mM)	R_e (Å)	δ_{R_e} (Å)	R_g (Å)	δ_{R_g} (Å)
0.001	5731.06	8.81	1604.60	1.78
0.025	5393.09	26.38	1487.54	6.46
0.1	5100.30	21.81	1383.59	4.72
0.15	4825.28	20.39	1322.53	1.22
0.25	4465.05	12.38	1284.93	1.58

Table 3.4: End-to-end distance (R_e) and radii of gyration (R_g) of DNA models at different salt concentrations. The δ_{R_e} and δ_{R_g} are the standard deviations of R_e and R_g respectively.

ulation data in Fig. 3.15, the DNA persistence length decreases as the salt concentration increases. At low salt concentration, the Debye screening length ($r_D = \kappa^{-1}$) is very large (i.e. $r_D \approx 3044.0$ Å at 0.001 mM 1:1 salt). It causes the Coulombic repulsive interaction between monomers to be almost unscreened. This strong monomer-monomer repulsion maximizes the persistence length. From Fig. 3.15, at relatively low salt concentrations 0.0-0.05 mM, the average L_p from simulations does not alter. At this concentration range, the L_p is about 1100 Å. As the salt concentration increases, the Debye length becomes smaller, resulting in a weaker DNA-DNA repulsive potential leading to smaller persis-

tence lengths. From Fig. 3.15, the DNA persistence length generally decreases at salt concentration 0.05 mM and above.

Fig. 3.16 shows some DNA snapshots at different salt concentrations. At salt concentra-

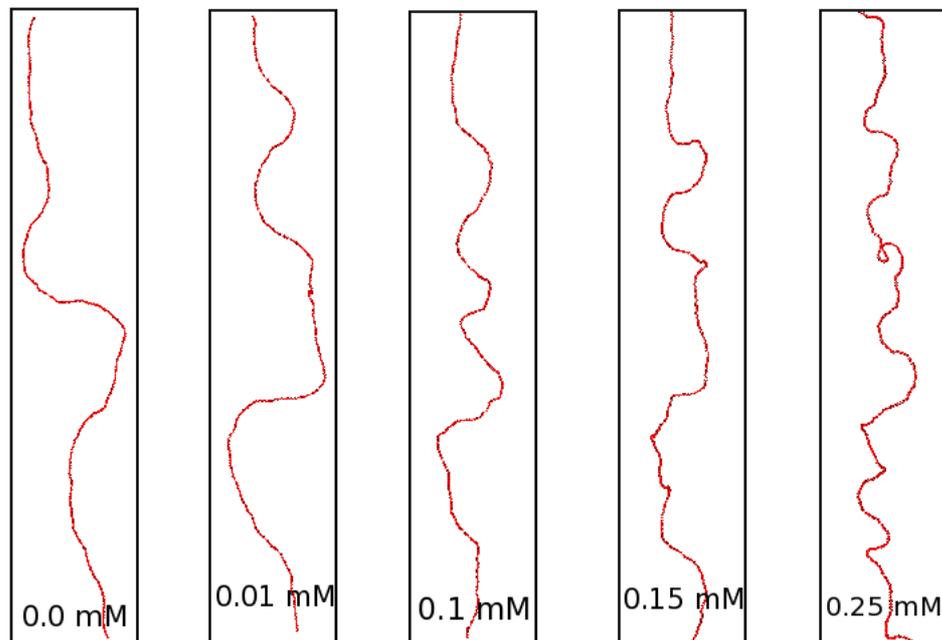


Figure 3.16: Snapshots of DNA chains in different salt concentrations. Salt concentrations are written at the bottom of the pictures.

tions below 0.01 mM, the DNA structures do not change significantly. At salt concentration above 0.01 mM, the DNA chains form a more curved structure. The curved structures are formed because the salt ions screen the repulsive Coulombic potential between DNA monomers. It is also clear that the weakening of the monomer-monomer repulsions by screening ions reduces the dimension of the overall DNA which is indicated by the data of end-to-end distance and radii of gyration (Table 3.4).

Figure 3.17 shows the DNA conformations at different time intervals (in reduced unit). The initial configuration of the DNA chain was created randomly inside the cubic simulation box. As the time elapsed, the chain equilibrates its conformation within the system. After some time the system energy and the chain conformation stabilized. These are sufficient indications of an equilibrium chain conformation.

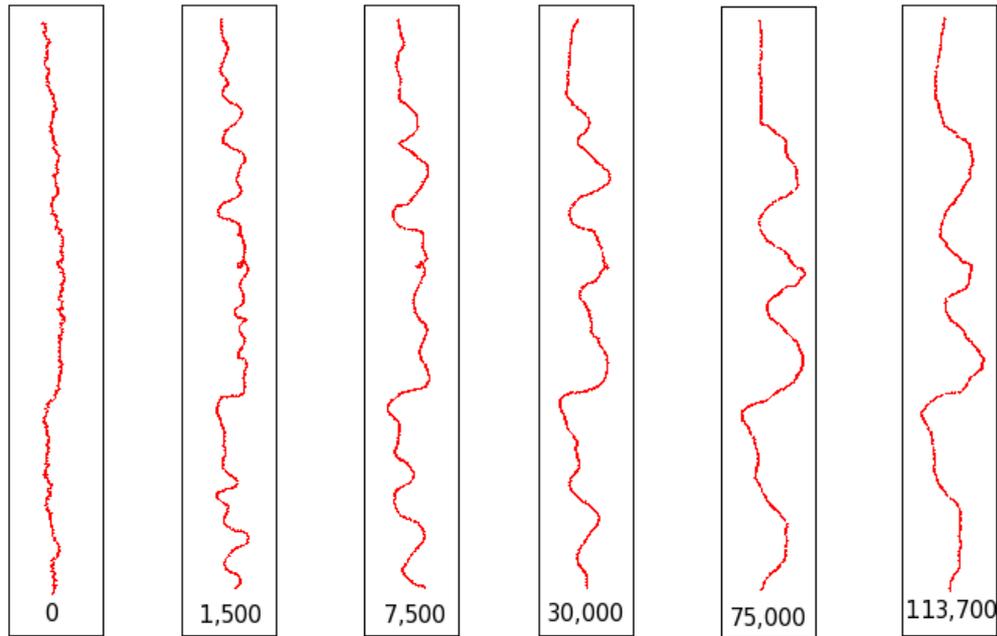


Figure 3.17: Snapshots of DNA conformation at different time intervals (reduced unit). DNA concentration 0.005 mg/ml and salt concentration 0.1 mM.

3.7 Experiment vs. Theory

In the following the comparison between experimental and theoretical persistence length is discussed. Theoretically, our model in simulation is not accurate enough to represent the experimental DNA flexibility, so we compare our theoretical predictions with real DNA data from experiments. We notice that the experimental data are based on different physical variables that arise from measurements that include light scattering, sedimentation velocity, electro-optics, ligase-catalized cyclization, gel electrophoresis, circularization kinetics, electric dichroism and scanning force microscopy (SFM) visualization [40]. These variables are then interpreted according to an appropriate theory that yields the persistence length. Indeed, there is inconsistency in the results. The above techniques have limitations, because the persistence length is not a measurable property, thus we have to convert the measurable property (e.g. diffusion coefficient, average radius of gyration, end-to-end distance, force of extension) to persistence length through model-dependent theories. Thus it is impossible to discuss the experimental persistence length in detail without considering each experimental method used. Another issue is the flexibility of the DNA is dependent on its base pair sequence [39].

In Fig. 3.18, the DNA persistence length from theoretical calculation is compared to

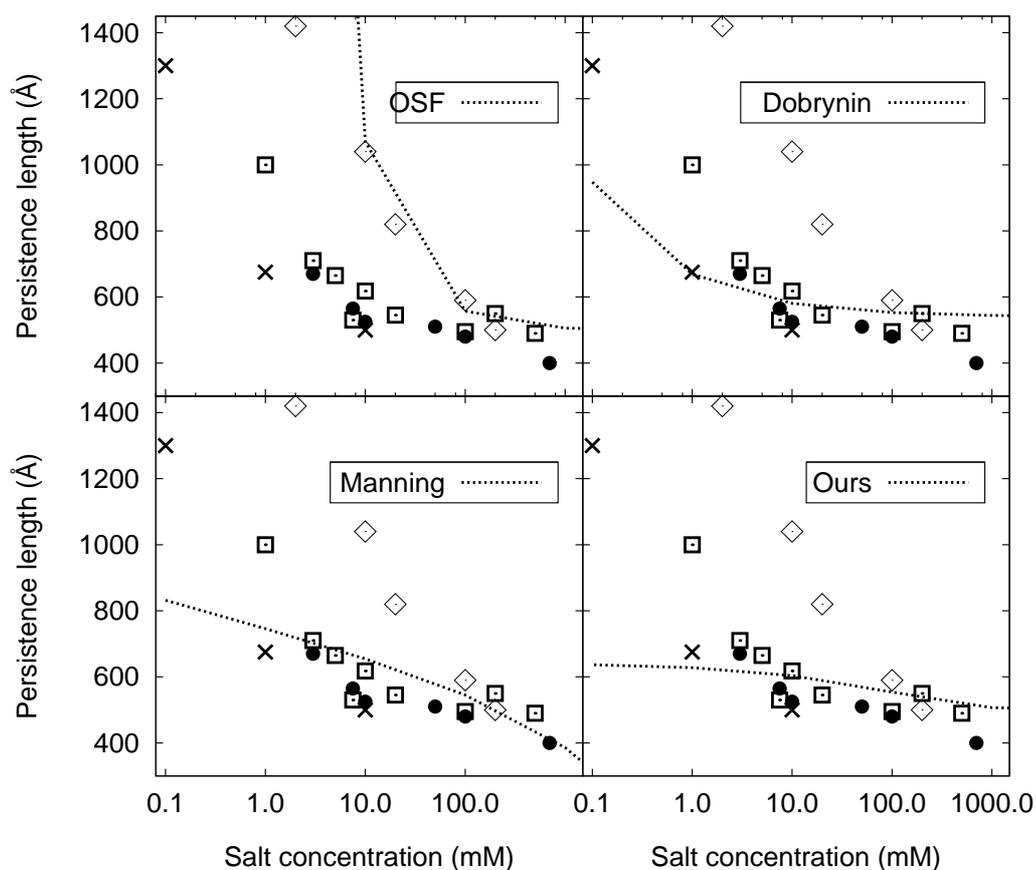


Figure 3.18: Experimental data and theoretical calculations of the DNA persistence length at different salt concentrations. The experimental data are taken from Baumann et al. [42] (rectangle); Nordmeier [41] (diamond); Rizzo and Schellman [43] (closed circle); Smith et al. [40] (cross).

some experimental data. It seems in general that the trends of the data for the different determinations are not consistent. It suggests the need of a thorough understanding of the experimental procedure before comparing experiment and theory. Due to the lack of our DNA experimental experience at the moment, we choose to focus on simulation and theories because both rely on universal mechanical laws.

3.8 Simulation vs. Theory

In Fig. 3.15, we compare the experimental persistence length to the theoretical prediction from the OSF (a), Dobrynin (b), Manning (c) theories and from our derivation (d). Specific to the Manning equation (Eq. 3.19), the monomer charge (q) variable is not

explicitly present. From his derivation [31], he use the dimensionless DNA line charge density parameter $\xi = l_B/b = 4.2$ to model the DNA charge properties where the DNA has a negative charge -1 for each 1.7 \AA distance. To apply Manning's calculation for our DNA model, we use a related line charge density parameter, $\xi = \frac{l_B}{b/|q|}$ instead of $\xi' = \frac{l_B}{b}$ where $b/|q|$ denotes the segment length per unit charge. Thus the $\xi = 4.2$ which is the same as $\xi' = 4.2$ used by the Manning model.

The equations of those theories are based on mechanics where the stiff rod model is used. Thus the usage of these theories are not limited to DNA only (i.e. DNA double helix with phosphate groups charge -1 and charge spacing 1.7 \AA). These theories should be applicable for any stiff polymer, including our DNA simulation models (DNA with monomer charge -12 and bond length 20.4 \AA). From Fig. 3.15, it is clear that there is no theoretical L_P calculation that can match the simulation results. The OSF, Dobrynin and Manning L_P go to infinity at very low salt concentrations. Our model retains the L_P as constant at low salt concentration, but as the salt concentration increases, the simulation L_P diminishes much faster than our L_P theoretical model which is still qualitatively accurate. However, the basic ideas of our derivation can be used to develop a chain stiffness concept, since it prevents the persistence length from going to infinity at very low salt concentration. We attribute the premature decrease of the simulation persistence length to the "ionic bridging" phenomenon which will be introduced in the Section 3.10.

3.9 $L_P^{\text{polyelectrolyte}} < L_P^{\text{neutral polymer}}$. Is It Possible?

In figure 3.19, the DNA persistence length derived by the OSF, Dobrynin and Manning theories and ours are presented. The parameters given are for the real DNA (charge $q=-1$, bond length $b=1.7 \text{ \AA}$). The L_P^0 in each model are defined to obtain the DNA $L_P=550 \text{ \AA}$, as the consensus persistence length of experimental DNA in 0.1 M NaCl concentration. The L_P^0 is $500, 540, 74$ and 500 \AA for the OSF, Dobrynin, Manning and our theory respectively. From the simulation data in Fig. 3.15, at salt concentration 0.25 mM , the simulation L_P (polyelectrolyte) decrease below the L_P^0 (neutral polymer). The condition $L_P^{\text{polyelectrolyte}} < L_P^{\text{neutral polymer}}$ is impossible for the OSF and Dobrynin theories and also our theoretical model because these theoretical calculations follow the basic assumption

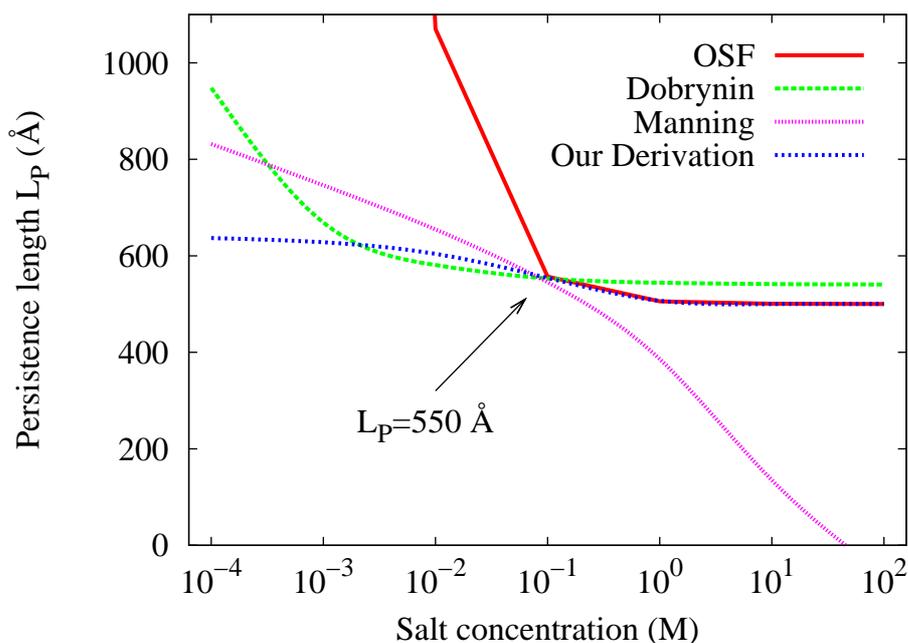


Figure 3.19: Theoretical persistence length (L_P) calculated by the OSF, Dobrynin and Manning theory, and our derivation.

$L_P=L_P^0+L_P^{el}$ (Eq. 3.28) with positive L_P^{el} . The L_P^{el} is the additional persistence length due to the monomer-monomer repulsion. The L_P equation given by Manning [31] (Eq. 3.19) is the only calculation which does not follow the expression $L_P=L_P^0+L_P^{el}$. The theoretical L_P given by Manning offers the possibility of a charged polymer to have persistence length lower than its uncharged polymer. But a quite disturbing fact from the calculation derived by Manning is that the persistence length value can even be negative at high salt concentration (Fig. 3.19) despite Manning's [[31], p.3613] statement "*the persistence length of DNA is many times larger than the persistence length of its uncharged isomer*". This implies that the DNA persistence length cannot be smaller than the uncharged DNA. Due to this contradiction, we may doubt the applicability of these theories to predict L_P over the entire range of salt concentrations. In the next section we introduce the "ionic bridging" phenomenon, which may lead to the inequality $L_P^{\text{polyelectrolyte}} < L_P^{\text{neutral polymer}}$ at salt concentrations 0.25 mM (Fig. 3.15). Thus we conclude that polyelectrolytes with smaller persistence length than its uncharged polymer is **possible** if the "ionic bridging" phenomena exists.

3.10 Ionic Bridging: a New Postulate?

We suggest that the premature persistence length decrease in our simulations is due to the unbalanced counterion condensation around the DNA surface. We use Fig. 3.20 to illustrate this concept.

Figure 3.20.(a) is the hypothetical condition, where we expect the counterion distributions in the regions α and β to be the same. This is the condition expected by many classical theories in charged systems, where charge distribution around a central macroion is independent of the orientation of the macroion. The bending angle 12° ($6^\circ + 6^\circ$) in Fig. 3.20.(a) gives an example of the polyelectrolyte equilibrium bending angle in the hypothetical state. We suggest that when the bending monomers orientate to the β region, the DNA negative charge density in the β region (narrow angle) is larger than in the α region. Due to the larger negative charge density, Na^+ ions would tend to be attracted in the β region. The accumulation of Na^+ ions in this region increase the bending angle of the DNA segment because of the larger magnitude of DNA– Na^+ attraction [Fig. 3.20.(b)]. The additional bending angle (e.g. of magnitude 8° in Fig. (b) which comes from $20^\circ - 12^\circ$), will again increase the negative charge density in the β region side. This positive feedback mechanism leads to a further DNA– Na^+ attraction in the β region by attraction of more adjacent DNA monomers which is propagated along the chain in the β region (Fig.(b),region β). The DNA– Na^+ attraction eventually produces a smaller preferred electrostatic energy. We call the change of the hypothetical (Fig. 3.20.(a)) to the real state (Fig. 3.20.(b)) as the "ionic bridging" or "counterion bridging" effect, because a large amount of Na^+ ions inside the narrow angle of region β acts as "bridge" between DNA monomers (from the figure point of view). Normally, the adjacent m_1 , m_2 , and m_3 (see Fig. 3.20) are mutually repulsive. The "bridging" is used to refer to the positioning charged salt ions that ameliorates the repulsive tendencies, allowing for an added curvature within β region. This ionic bridging effect is balanced by the polymer bending potential, DNA monomer-monomer repulsion and the system entropy. Figure 3.21 presents two snapshots from our simulations to substantiate the occurrence of the ionic bridging effect. The red spheres are the DNA monomers and the small purple spheres are the Na^+ ions. It is clear that when Na^+ ions accumulate at one side of DNA surface,

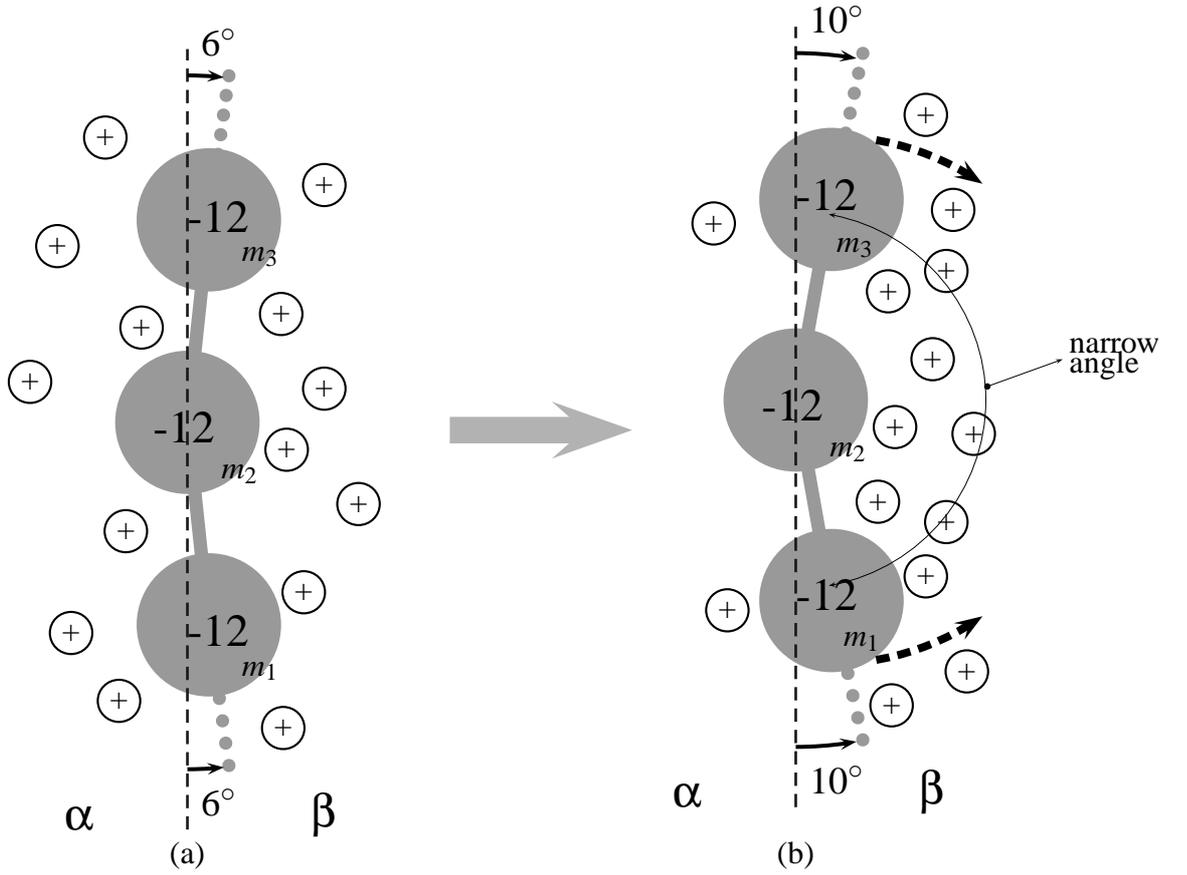


Figure 3.20: Figure (a) is the hypothetical condition. Fig. (b) is the real condition. The vertical dashed line is an imaginary boundary dividing the polymer environment at the equilibrium bending plane.

the DNA bending segments orientate to the side of Na^+ accumulation. We denote A_1 and A_2 in each figure as the regions where ionic bridging occurs. We consider the ionic bridging in the A_1 and A_2 regions as the local in nature because the additional bending occurs amongst the closest monomer neighbors. We denote by \mathbf{A} in both figures as the global ionic bridging region because in this region the Na^+ ions can "bridge" the DNA monomers along larger distances (e.g. amongst the oppositely facing monomers indicated by the double arrow line in (a)).

In the following we attempt to include the ionic bridging effect in the expression of the persistence length calculation. For clarity we collect and rewrite some equations

$$\langle \theta^2 \rangle = \frac{2 \int_0^{2\pi} e^{-\frac{B\theta^2/2s}{k_B T} - \frac{\sum_{i=1}^n \sum_{j=i+1}^{n+1} \frac{l_B k_B T q_i q_j \exp(-\kappa n b G)}{sG}}{k_B T}} \theta^2 d\theta}{\int_0^{2\pi} e^{-\frac{B\theta^2/2s}{k_B T} - \frac{\sum_{i=1}^n \sum_{j=i+1}^{n+1} \frac{l_B k_B T q_i q_j \exp(-\kappa n b G)}{sG}}{k_B T}} d\theta} \quad 3.24$$

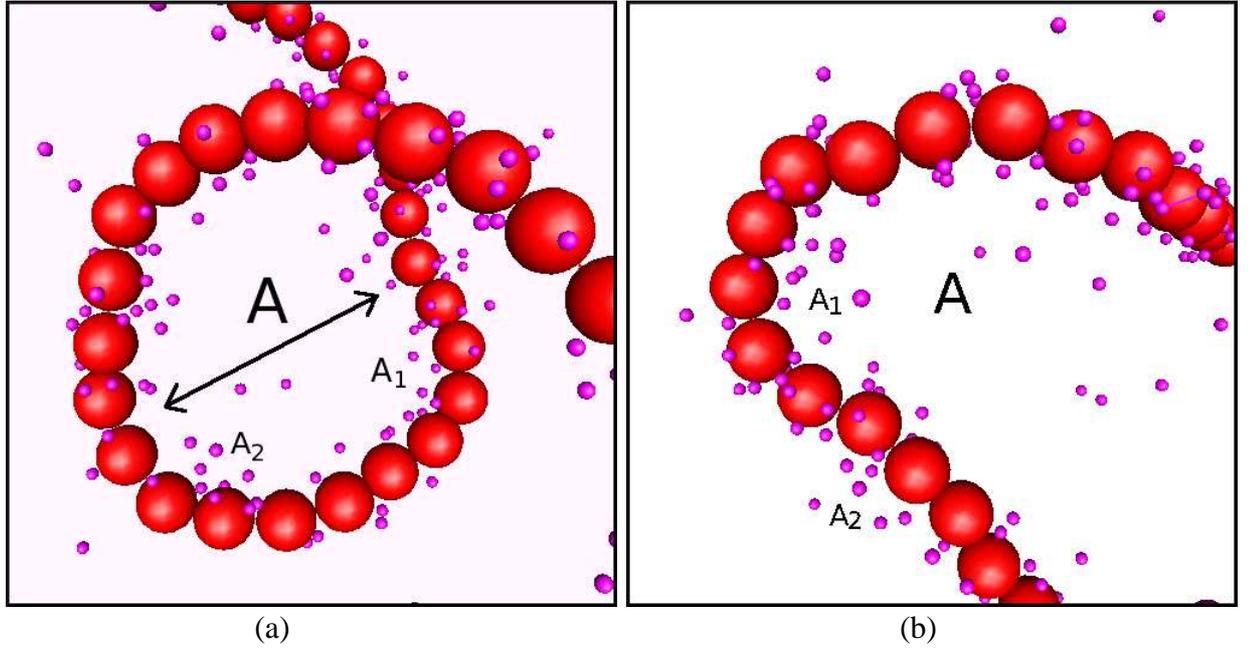


Figure 3.21: Snapshots from the DNA chain simulations in salt concentration 0.25 mM (a), and 0.30 mM (b). Area \mathbf{A} gives the global ionic bridging, and region A_1 and A_2 denote the local ionic bridging. We run the simulation at 0.30 mM (which took one month) only to focus on the ionic bridging without sampling because this would too much time.

$$B_{\text{new}} = B + \sum_{i=1}^n \sum_{j=i+1}^{n+1} \frac{2l_B k_B T q_i q_j \exp(-\kappa n b G)}{G \theta^2} \quad 3.26$$

$$L_P = \frac{B + B_{el}}{k_B T} = L_P^0 + L_P^{el} \quad 3.28$$

If we just rely on the above equations to determine L_P , where $L_P < L_P^0$, we will end up with negative L_P^{el} . The negative L_P^{el} is impossible because the second term of Eq. 3.26 which supplies the value for L_P^{el} is always positive for $q_i = q_j$.

Recall that the goal is not to have negative L_P^{el} , but to include the ionic bridging effect to produce "small" L_P . In the following, there are two possible ways for the ionic bridging inclusion to provide smaller L_P

1. The exponential term in Eq. 3.26 comes from the Debye-Huckel potential theory, where the monomer charge repulsion reduces due to the screening of ions. The Debye-Huckel potential (exponential term) appears from models which have high symmetry (planar, spherical or cylindrical), implying an isotropic screening about the symmetrical axis. Since the ionic bridging produces non-isotropic ionic distri-

butions between both sides of Fig. 3.20.b, the exponential term in Eq. 3.26 needs to be modified completely (i.e. by using the Poisson-Boltzmann equation). The exponential term cannot be retained to produce small L_P because it always supplies a positive value to the second term of Eq. 3.26. The only possible way to have smaller L_P in this way is when the screening ions lead to changes in the the potential sign of the monomer charge such that the monomer-monomer repulsive force becomes attractive. Then the second term in Eq. 3.26 becomes negative.

2. For the second possibility, consider the chain conformation in Fig. 3.20.a at a certain state where the availability of the mobile ions do not affect that particular state. If ionic bridging exists, it implies the ions can change the chain state of Fig. 3.20.a into another "preferred" state such as shown in Fig. 3.20.b. Hence the chain state of Fig. 3.20.b has a larger degeneracy than the state in Fig. 3.20.a. Then there should be another factor in Eq. 3.24 to denote the degeneracy due to ionic bridging, i.e.

$$\langle \theta^2 \rangle = \frac{2 \int_0^{2\pi} G(\theta) e^{-\frac{B\theta^2/2s}{k_B T} - \frac{\sum_{i=1}^n \sum_{j=i+1}^{n+1} \frac{l_B k_B T q_i q_j \exp(-\kappa n b G)}{s G}}{\theta^2} d\theta}{\int_0^{2\pi} G(\theta) e^{-\frac{B\theta^2/2s}{k_B T} - \frac{\sum_{i=1}^n \sum_{j=i+1}^{n+1} \frac{l_B k_B T q_i q_j \exp(-\kappa n b G)}{s G}} d\theta},$$

where $G(\theta)$ is the degeneracy factor. Thus if ionic bridging exists, it could produce a smaller L_P since the $G(\theta)$ is relatively large for larger θ .

To examine the existence of the ionic bridging effect, we simulate a short DNA chain in 20 mM salt concentration. The number of monomer equals 21 monomers. Short DNA chain is chosen to provide a much smaller space thus we can focus in observing the particles displacement. We fix the position of the center monomer (the 11st monomer) to minimize the chain translational movement so that we can focus on the chain bending conformation. Figure 3.22 gives the snapshot of the short DNA chain conformation at different time intervals. The snapshots were taken randomly from the particle's trajectory. We used a small time-step ($1 - 2 \times 10^{-5}$ in reduced unit) to minimize the amplitude of the displacement of mobile ions. Even with such small time step, we observe that the fluctuation of the ions in space is still quite large (i.e. the variance of the ion concentration at a specified site is very high during the sampling over time). This is a qualitative statement and a whole new quantitative theory must be constructed to verify the simulation data. From

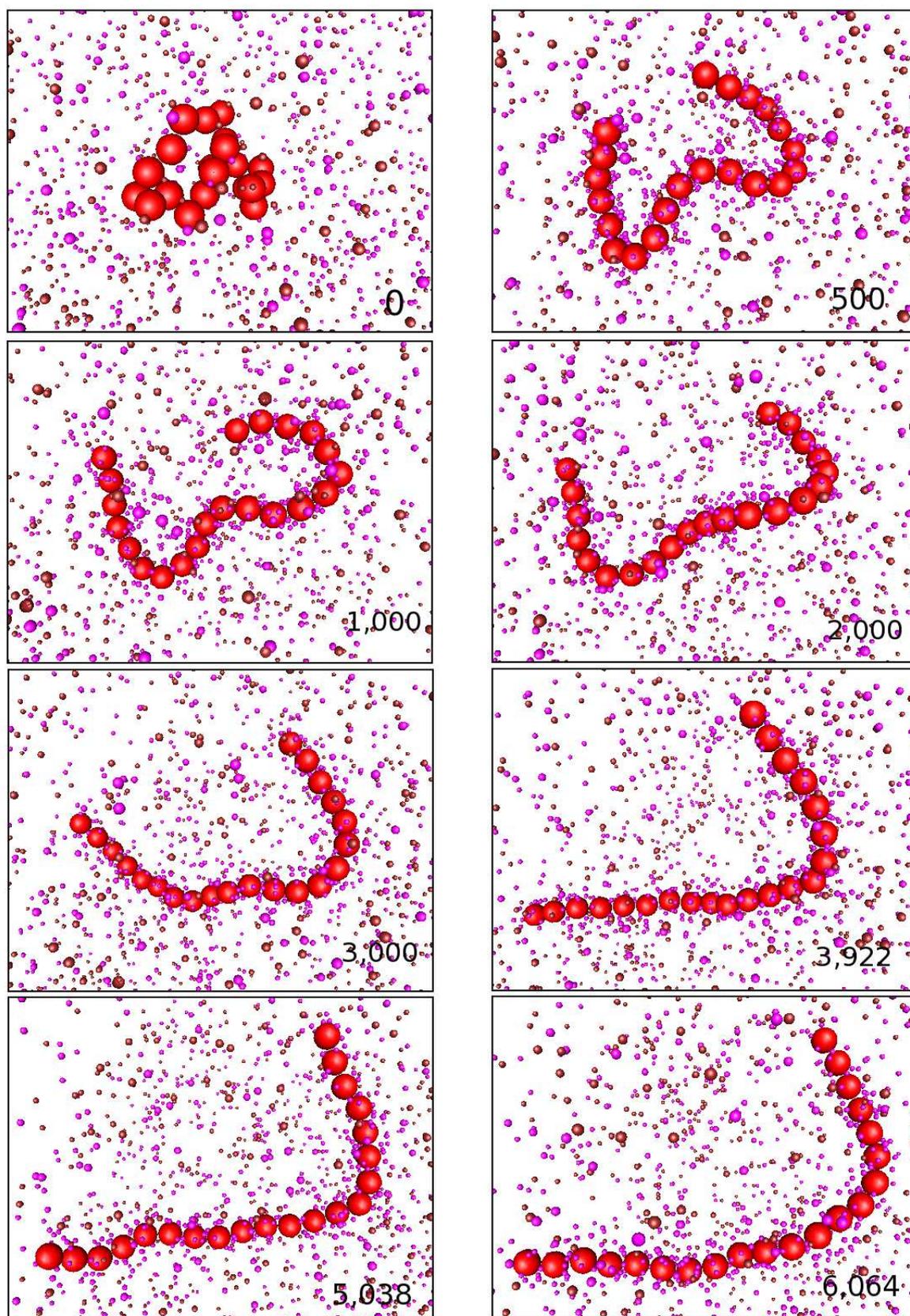


Figure 3.22: A short DNA chain conformation in different time intervals. The number of monomer equals 21. NaCl salt concentration is 20 mM. This simulation was run to examine the availability of ionic bridging effect in different time intervals. The Na⁺ and Cl⁻ particles are modeled by the purple and brown spheres respectively.

Fig. 3.22, the accumulation of the Na^+ particles (purple spheres) at the narrow angle is more pronounced at the beginning of simulation (time interval $\sim 0 - 1000$). It is shown that when the chain is in bent conformation, many Na^+ ions positioned themselves within the narrow angle. This Na^+ positioning supports the proposal of ionic bridging effect. At larger time interval ($\sim 3922 - 6064$) where the system has reached equilibrium, there is only one bending site in the chain line. More linked monomers were present in straight conformation due to the strong monomer-monomer repulsion and large chain stiffness. In larger time intervals, the indication of the ionic bridging effect is not clearly visible. We attribute the low indication at large time interval to the minimum bent conformation and the large fluctuation of mobile ions in space. We suggest that the fluctuation of ions in the narrow chain angle occurs because of the balance of ionic bridging and Na^+ ions repulsion. To precisely determine the existence of ionic bridging, one possible method is measuring the local density around a charged polymer. This measurement is not included in this work because separate research with specially designed simulation and algorithms are needed to pursue this topic.

As far as is known, this "ionic bridging" phenomenon is never accounted for in calculating the DNA persistence length. We conclude that studies of the mutual effects of chain orientations and ionic distributions are essential for determining the topological properties of polyelectrolytes.

Chapter 4

IONIC DISTRIBUTION IN A DNA SYSTEM

4.1 Theoretical Review of Ionic Distribution

The radial distribution function (symbolized $g(r)$) describes how the density of surrounding matter varies with the distance function. It also denotes the probability of finding a particle at a distance r from a central particle. Consider a spherical shell with inner radius r and outer radius $r + \delta r$. If we specify r as the distance from any point within the shell with thickness δr , we can express $g(r)$ as:

$$g(r) = \frac{n_{\text{real}}}{n_{\text{bulk}}}$$

where n_{real} is the total number of particle between r and $r + \delta r$, and n_{bulk} is the number of particles if the density of the system is homogeneous everywhere.

At infinitely low density, where the effect of the third particle to a pair interaction can be neglected, the radial distribution function (RDF) between central particle M and diffused particle type i can be determined by a Boltzmann-like distribution [44]:

$$g(r) = \frac{c_{iM}}{c_i} = \exp\left(\frac{-z_i e \Psi(R_{iM})}{k_B T}\right) \quad (4.1)$$

where c_{iM} is the concentration of the ion i at a distance R_{iM} (from the surface of particle M to the center of particle i), c_i is bulk the concentration of ion i , z_i is the valence of ion i , and $\psi(R_{iM})$ is the potential "felt" by particle i at position R_{iM} from the surface of particle M . In this thesis, the central particle M will be used to denote a macroion or DNA monomer, while the term particle i will point to any of the ionic species Na^+ , Cl^- or Nucleosome Core Particle (NCP). The Debye-Huckel approximation (DHA) can be used to calculate the value of $\psi(R_{iM})$, where it is an approximate solution of the Poisson-Boltzmann equation [45].

Debye-Huckel Approximation

Debye-Huckel approximation (DHA) is derived from the Poisson-Boltzmann equation with the assumption the system is low in potential, or $\left| \frac{z_i e \psi(R_{iM})}{k_B T} \right| < 1$. In a univalent electrolyte system, DHA obtains at potential $\psi(R_{iM}) < 25.7$ mV. The potential at a distance R_{iM} from a central ion M (DNA monomer) surface is obtained from DHA [46] as:

$$\psi(R_{iM}) = \psi_M \frac{r_M}{R_{iM} + r_M} \exp(-\kappa R_{iM}), \quad (4.2)$$

where

$$\psi_M = \frac{\sigma_M}{\epsilon} \frac{r_M}{1 + \kappa r_M} \quad (4.3)$$

$$\kappa = \left(\frac{1000 e^2 N_A}{\epsilon k_B T} \sum_i z_i^2 c_i \right); \quad (4.4)$$

ψ_M is the surface potential of the DNA monomer with radius r_M , κ is the inverse Debye length, ϵ is the dielectric of the medium, k_B is the Boltzmann constant, e is the elementary proton charge 1.602×10^{-19} C, T is temperature and N_A is the Avogadro number.

The distance R_{iM} and surface charge density is defined by:

$$R_{iM} = r_{iM} - r_M \quad (4.5)$$

$$\sigma_M = \frac{z_M e}{4\pi r_M^2}, \quad (4.6)$$

where r_{iM} is the distance of the center of the ion i from the center of DNA monomer and z_M is the DNA monomer charge. Some extensions of DHA has been proposed, for example the Far-Field Approximation (FFA). According to Sader [47], the FFA has a wider range of applicability than DHA.

4.2 Clarifying the Ambiguity of the Screening Parameter

κ

In some references, we find that the ionic strength parameter κ (Eq. 4.4) for calculating any electrostatic properties varies in term of ions involved in the screening effect. . The parameter κ itself characterizes the magnitude of ionic screening in the system because its value is the inverse of the Debye screening length. Morisada et al. [46] mentioned that we only need to use the z_i and c_i from salt particles, while Schmitz [48] mentioned there are conditions where we should include the counterion and macroion as well. Thus the specification regarding which ions should be included in determining the κ parameter is still unclear.

In this section, we examine this ambiguity. We will be able to obtain an unambiguous κ in analysis which will be used to compute potentials throughout this chapter. We simulate free macroions having the same dimension and properties as the DNA monomer. We will call these macroions as particle-like DNA monomer (PLDNA). These monomer macroions, with the concentration equal to the number of monomers in the DNA chain, which itself has a definite concentration in our simulation, are freely dispersed in salt solution. Then the radial distribution function of PLDNA obtained from simulation will be compared to the RDF calculated theoretically by PB the model. There are three types of κ used in RDF calculation from the model that utilizes the DHA in conjunction with Eq. 4.1, and this concept will be referred to as PB hereafter. The first uses only salt ions with charge z_i and counterion c_i (κ_{salt}) as the only independent particle variables ($\kappa_{\text{salt}} = \kappa_{\text{salt}}(c_i, z_i, \Omega)$) where Ω are the other thermodynamical variables. The second is a function of salt and counterion charge and concentration ($\kappa_{\text{salt-counterion}}$) and the third is a function of DNA, salt and counterion charge and concentration ($\kappa_{\text{all-ion}}$). The RDF calcu-

lation results that most agree with the simulation results will be taken as the calculation having the proper κ .

In Fig. 4.1 and 4.2, the simulation RDF of two different PLDNA concentrations are

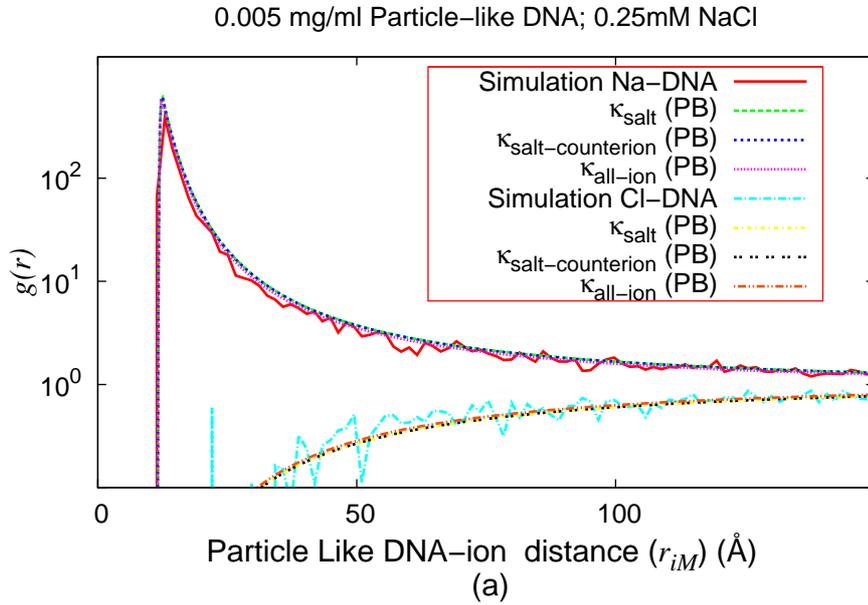


Figure 4.1: PLDNA–Na⁺ and PLDNA–Cl[−] RDF obtained from simulation and PB model calculation. The PB model is calculated for three kinds of ionic screening parameters κ . PLDNA concentration 0.005 mg/ml and salt concentration 0.25 mM

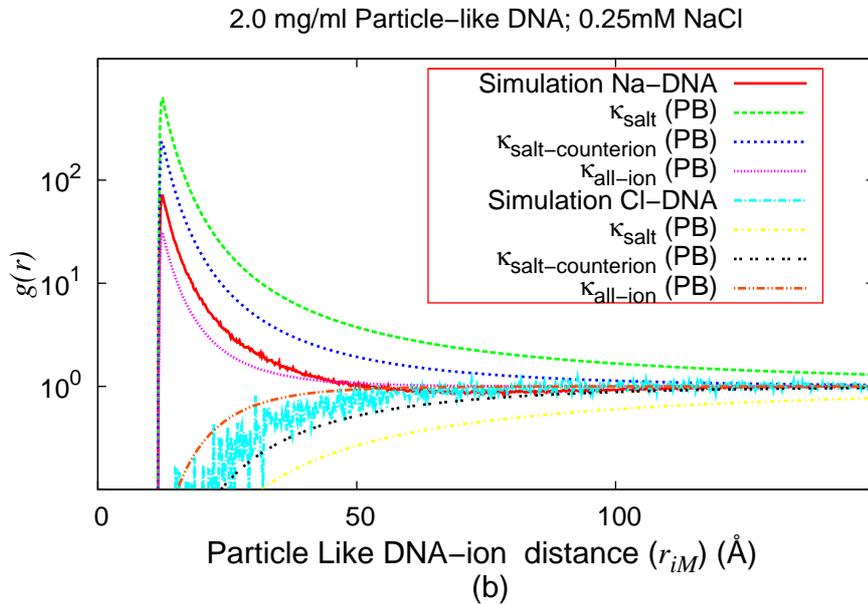


Figure 4.2: PLDNA–Na⁺ and PLDNA–Cl[−] RDF obtained from simulation and PB model calculation. The PB model is calculated for three kinds of ionic screening parameters κ . PLDNA concentration 2.0 mg/ml and salt concentration 0.25 mM. The only system difference from Fig. 4.1 is the PLDNA concentration

compared with the RDF results from PB calculations [46]. At PLDNA concentration

0.005 mg/ml, all three calculations with different κ match the simulation result and are indistinguishable. At particle-like DNA concentration 2.0 mg/ml, the RDF calculations are distinguishable but there is no match with the simulation results. Recall that the DHA is an approximation for low potentials, not for low macroion concentration. Thus the simulation-PB disagreement at DNA concentration 2.0 mg/mg is not due to the DHA. It implies that the simulation-calculation RDF disagreement occurs because the Boltzmann-like expression (Eq.4.1) is inapplicable at DNA concentration 2.0 mg/ml (or higher). The assumption in Boltzmann-like equation (Eq.4.1) is that the interaction and distance between two particles is random and independent of the interaction with other particles. At higher DNA concentration, the PLDNA–Na⁺ attraction will be reduced due to strong attractive interaction between the Na⁺ ion and other close PLDNA ion. At very low DNA concentrations, the Boltzmann-like equation is applicable because the distance between PLDNA in the system are far apart. Thus the effect of the third DNA particle in the DNA–Na⁺ interaction is negligible.

To overcome the RDF disagreement at high concentration of DNA, we apply the Ornstein-Zernike (OZ) relationship instead of the Boltzmann-like expression, where the effect of the third particle is accounted for in any pair interaction. For details of the OZ method, refer to the following references for example [48–51]. The OZ relationship is expressed by the calculation that can also be iteratively utilized:

$$h_{ij}(\mathbf{r}) = c_{ij}(\mathbf{r}) + \sum_{k=1}^S \rho_k \int c_{ik}(\mathbf{r} - \mathbf{r}') h_{kj}(\mathbf{r}') d\mathbf{r}' \quad (4.7)$$

with

$$h_{ij}(\mathbf{r}) = g_{ij}(\mathbf{r}) - 1 \quad (4.8)$$

where $h_{ij}(r)$ is defined as the *total correlation function*, $c_{ij}(r)$ is the *direct correlation function* and ρ_k is the density of particle type k . We can obtain another $h_{ij} - c_{ij}$ relationship by using the Hypernetted-Chain Approximation (HNC) and Percus-Yevick Approximation (PYA). The HNC and PYA expressions are:

$$c_{ij}^{\text{HNC}}(r) = -\beta V_{ij}(r) + h_{ij}(r) - \ln[h_{ij}(r) + 1] \quad (4.9)$$

$$c_{ij}^{\text{PYA}}(r) = (1 - \exp[\beta V_{ij}(r)]) (h_{ij}(r) + 1). \quad (4.10)$$

In order to use the OZ equation, the initial value of $h_{ij}(r)$ is guessed. Then the $c_{ij}(r)$ counterpart is calculated by either PYA or HNC. The initial $h_{ij}(r)$ and $c_{ij}(r)$ values are used as the input for the OZ equation to produce a new $h_{ij}(r)$ in Eq. 4.7. This process is iterated until the value of $h_{ij}(r)$ and $c_{ij}(r)$ converge. For our system consisting of three different kind of particles, we need to calculate nine correlations, DNA–DNA, DNA–Na⁺, Na⁺–DNA, DNA–Cl⁻, Cl⁻–DNA, Na⁺–Na⁺, Na⁺–Cl⁻, Cl⁻–Na⁻ and Cl⁻–Cl⁻. The integration in the OZ equation is in 3-dimensional space, consequently this calculation is highly time consuming. To reduce the cpu time, we have to parallelize the C++ code that was specifically written for this purpose. As an illustration, for one run using 8 processors, we need 3-5 days for the iteration to converge, with speeds about 6-7 times faster compared to serial run [52–56]. Applying the OZ computation is not straightforward. We need to make a reasonable guess such that during the interaction the fluctuation in the intermediate steps do not become too large so as to cause divergences. Fig. 4.3 shows the results for HNC and PYA. These graphs show that the RDF calculation using $\kappa_{\text{salt-counterion}}$ are in close to the simulation. The above calculation is limited to the effect of the third particle. More accurate calculations can be performed if we apply the OZ equation at higher orders, i.e. count the effect of the 3rd, 4th, \dots , n^{th} particles simultaneously. Since the closest RDF is found when we use $\kappa_{\text{salt-counterion}}$, we will use this κ for the rest of our calculations. From our observation in performing the OZ equation, some tempting questions appear. Such as (i) is the equation for calculating κ is system specific? and (ii) are the screening effects encountered by each particle type in a system different?. In Fig. 4.3 we can notice an anomaly at one of the graph from the PYA calculation, specifically at the screening parameter κ equals κ_{salt} . For that anomalous graph, we infer

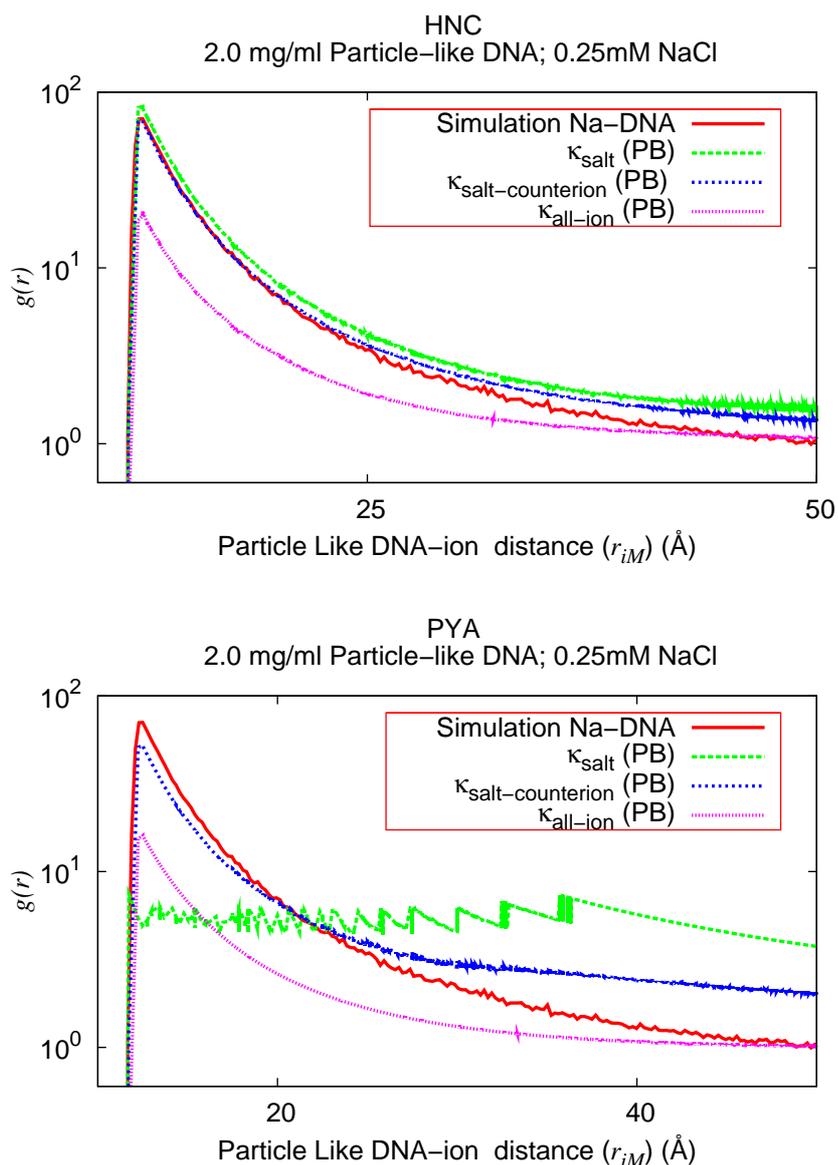


Figure 4.3: PLDNA–Na⁺ RDF obtained from simulation, HNC and PYA. The HNC and PYA calculation are done for three kinds of ionic screening parameter κ .

that the iteration of the PYA calculation did not converge. In the numerical iteration, we observe that all HNC and PYA calculations will experience an "oscillating phase" before finally converging to a certain smooth graph without significant changes in further iterations. We have attempted methods to break the anomaly in the graph in Fig. 4.3 (i.e. by using different grid widths or different initial condition), but the result still persists. It could reflect the limits of applicability on the PYA application because the other graphs depict the expected forms.

4.3 Coordination number in various salt concentration

The first variable we wish to measure is the coordination number of ions about any DNA monomer. We concentrate on the Na^+ ions here, since Cl^- is repulsive and there are only 12 NCP particles. As the DNA monomer has a nett charge -12 , Na^+ ions will accumulate at the DNA surface. The expression to calculate the first coordination number [57] is:

$$n_{\text{DNA}}^{\text{Na}} = 4\pi\rho_{\text{Na}}^0 \int_0^{r_s} g_{\text{DNA-Na}}(r)r^2 dr . \quad (4.11)$$

where $g_{\text{DNA-Na}}(r)$ is the RDF of DNA– Na^+ pair interaction, ρ_{Na}^0 is the Na^+ bulk density and r_s is the position of the first valley of $g_{\text{DNA-Na}}(r)$.

Table 4.1 and 4.2 gives the coordination number of Na^+ ions at DNA concentration

Salt Concentration (mM)	$n_{\text{DNA}}^{\text{Na}}$
0.0	13.87
0.001	14.00
0.0025	14.22
0.005	13.67
0.0075	13.62
0.01	13.49
0.025	12.75
0.075	12.96
0.1	13.41
0.15	12.58
0.25	14.45

Salt Concentration (mM)	$n_{\text{DNA}}^{\text{Na}}$
0.0	15.67
0.01	15.82
0.1	15.57
1.0	15.86
2.5	16.34
7.5	16.69
10.0	16.96
25.0	18.25
50.0	19.11
75.0	20.33
100.0	21.66

Table 4.1: Coordination number of Na^+ ions about DNA. DNA concentration 0.005 mg/ml. First valley distance of DNA–Na RDF= 26.6420 Å

Table 4.2: Coordination number of Na^+ ions about DNA. DNA concentration 2.0 mg/ml. First valley distance of DNA–Na RDF= 26.4739 Å

0.005 mg/ml and 2.0 mg/ml respectively. At DNA concentration 0.005 mg/ml, where the salt concentration varies between 0–0.25 mM, we do not observe a trend in coordination number. We surmise that at DNA concentration 0.005 mg/ml, the lack of a trend is because the salt concentration range is too narrow.

At DNA concentration 2.0 mg/ml, the coordination number is relatively constant at low salt concentration. At a certain threshold ($\sim 10\text{mM}$ NaCl at this study), the coordination number increases as the salt amount increases. This statement supports the previous argument when the DNA concentration was 0.005 mg/ml, where the coordination number does not change at low salt concentration. We attribute the increase of coordination number at high salt concentration to the higher possibility of DNA–Na⁺ contact, leading to a greater propensity for DNA–Na⁺ binding.

4.4 Determining the Electrostatic Free Energy of DNA–NCP interactions

Currently models are required to estimate the DNA–ion interactions. We review here the rather primitive first attempts where an extended cylindrical shell (representing the environmental charge) interacts with an extended rod (representing the negatively charged DNA polymer chain) inside the cylindrical shell.

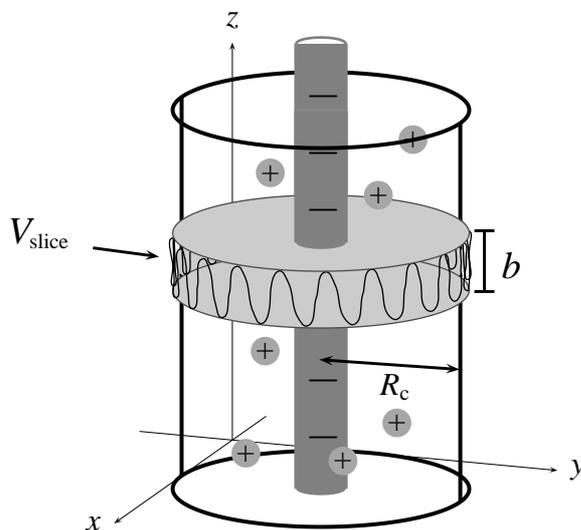


Figure 4.4: The cylinder model of the charged system. The inner cylinder is an infinite charged rod with charge spacing b . R_c is the cylinder radius defined by the charged rod concentration. V_{slice} is a cylinder volume that is perpendicular to the rod with height b and radius R_c .

4.4.1 The Poisson-Boltzmann Cylindrical Cell Model (PBCCM)

Changes to the electrostatic energy and entropy occurs when the DNA polyelectrolyte is introduced into the salt system. The original expression of the additional electrostatic energy E_{el} per unit charged group is (i.e see [58–60]) :

$$E_{el}/k_B T = \frac{1}{2l_B} \xi \phi_M + \frac{1}{2} \int_{r_M}^{R_c} 2\pi r \sum_{\alpha} (z_{\alpha} \rho_{\alpha}(r)) \phi(r) dr, \quad (4.12)$$

where the $\phi_M = e\psi_M/kT$ is the reduced electrostatic surface potential, $\phi(r) = e\psi(R_{iM})/kT$ is the reduced electrostatic potential with $r = R_{iM} + r_M$, $\xi = l_B/b$ is the line charge density of the polyelectrolyte, and $\rho_{\alpha}(r)$ is the charge density of α -species at a distance r from the DNA polyelectrolyte surface which is given as $\rho_{\alpha}(r) = g_{\text{DNA-}\alpha}(r) \cdot \rho_{\alpha}^0$. The $g_{\text{DNA-}\alpha}(r)$ is the RDF of species α relative to the DNA ions and ρ_{α}^0 is the bulk charge density of species α . According to Korolev et al. [58], R_c is the radius of the cylindrical cell defined by the DNA concentration (Fig. 4.4), $c_{\text{DNA}} = \xi/4\pi l_B N_A R_c^2$.

We note that the right the right hand side of Eq. 4.12 has dimension length^{-1} which is dimensionally inconsistent. Nevertheless, equation 4.12 can still be used if interpreted according to that which follows. We require the value of the electrostatic energy per unit bond length. In order to obtain the reduced energy $E_{el}/k_B T$, we have to exclude the term length^{-1} in the calculation. A clearer expression for Eq. 4.12 is:

$$\{E_{el}/k_B T\}/(\text{unit length}) = \left[\frac{1}{2} \phi_M + \frac{1}{2} \int_{r_M}^{R_c} 2\pi r b \sum_{\alpha} (z_{\alpha} \rho_{\alpha}(r)) \phi(r) dr \right] / b, \quad (4.13)$$

where we calculate the value inside the square bracket of Eq. 4.13 as the electrostatic energy between r_M and R_c .

The electrostatic entropy contribution due to the mixing and redistribution of ions around a polyelectrolyte is dimensionally inconsistent given by [58, 60]:

$$S_{el}/k_B = - \int_{r_M}^{R_c} 2\pi r \sum_{\alpha} \rho_{\alpha}(r) \ln \left[\frac{\rho_{\alpha}(r)}{\rho_{\alpha}^0} \right] dr, \quad (4.14)$$

where the explanation about the usage and unit of electrostatic entropy is analogous to the previous electrostatic energy equation. Korolev et al. [59] has calculated the electrostatic

contribution in the formation of DNA–NCP complex. At their work, they used PBCCM calculation to determine the density profile (RDF) of the DNA-ion pair interaction.

Since RDF from simulation $\rho_\alpha(r)$ is known, we check whether the equations given by Stigter Dirk [60] and Korolev et al. [58] is applicable to our system. We perform two simulations, the first involves the DNA polymer and the second involves particle-like DNA monomers (PLDNA), both of which exist within an ionic milieu. Except for the shape of the DNA macroion, both types of simulations are conducted under the same physical and ionic conditions.

In Fig. 4.5, the DNA–Na⁺ and DNA–Cl[−] RDF from the pair PB-model, simulation of DNA and simulation of PLDNA are compared. Fig. 4.5 shows that the PB model RDF

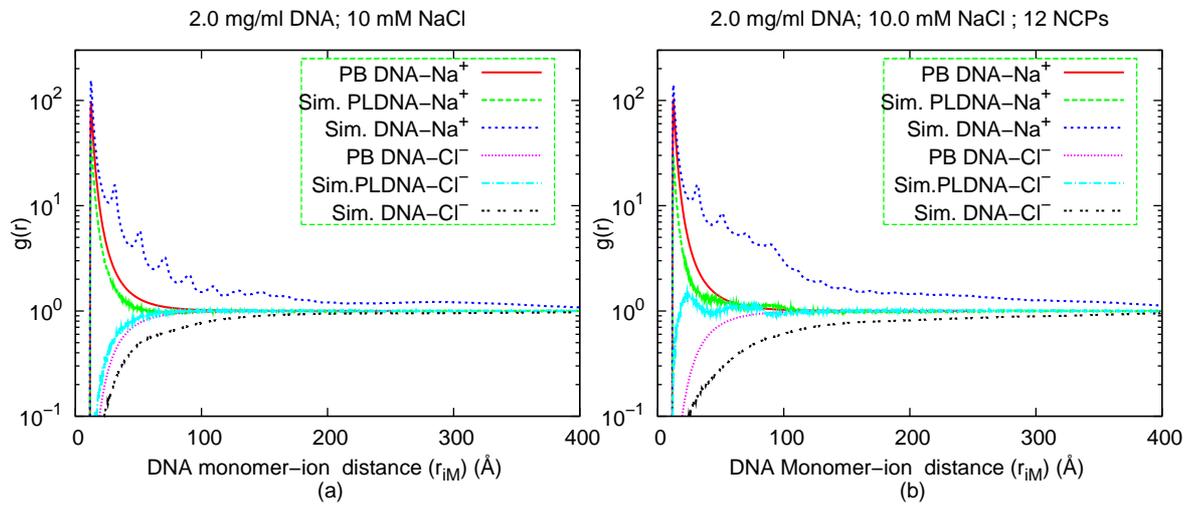


Figure 4.5: DNA–Na⁺ and PLDNA–Na⁺ RDF simulation compared to PB calculation. Simulation are done without (a) and with NCP (b)

is only comparable to the RDF of the Particle-Like DNA (PLDNA) simulation. The PB model cannot explain the fluctuation of the DNA–Na⁺ RDF which occurs in the polyelectrolyte DNA simulation. The fluctuation of DNA–Na⁺ RDF is uniform within the range 17.5–20 Å. We infer that these uniform fluctuations occur due to the Na⁺ accumulation at DNA monomer neighbors because the DNA–DNA monomer distance is 20.4 Å. Fluctuation distances which are smaller than the DNA bond length are attributed to the effect of DNA bending.

Stigter Dirk [60] and Korolev et al. [58] obtained the potentials $(\phi_M, \phi(r))$ for Eq. 4.12 with the PBCCM and they used the Boltzmann-like integral, Eq. 4.1, to derive the particle density profile. Since the RDF obtained from Eq. 4.1 does not account for the availability

of the third particle, we believe that those authors actually only calculated the potential and particle density that obtains for regions that are perpendicular to the axis of the DNA rod, such as the region within V_{slice} (see figure 4.4), with no lateral interactions other than the perpendicular plane. Hence they did not account for the potential and density at the upper and lower position of the monomer. Thus, unless our interpretation of the papers of [60] and [58] is wrong, they principally divided an infinite rod cylindrical rod into a stack of many small cylinders with the height of the cylinders equal to the monomer bond length. By calculating the electrostatic energy and entropy in a small cylinder individually and neglecting the electrostatic contribution of the neighboring cylinders, they obtained the electrostatic contribution per monomer charge (or per bond length). Hence equation 4.12 and 4.14 do not account the electrostatic contributions of cylinder neighbors, whereas the RDF profile of our simulation is the result of all interactions, including of course the lateral ones. This statement is supported by the RDF result from simulation involving PLDNA where the derived PB RDF used by [60] agrees to the RDF of PLDNA, where the DNA–DNA distance effect on the DNA–Na⁺ interaction is negligible

This conclusion casts doubts as to, whether our RDF simulation result can be accounted for by the Eqs. 4.12 and 4.14. One method that can be used to correlate our polymer RDF with that of equation (4.12 and 4.14) is to take the RDF data from our polymer DNA simulation until the first valley, which is the first layer of DNA–Na⁺ interaction. Thereafter, we use the RDF data of either PB or PLDNA simulation for all other distances. This method avoids counting Na⁺ ion aggregation at DNA monomer neighbors, but there is a discontinuity in the DNA–Na⁺ RDF profile because of matching the two different RDF data. This RDF discontinuity resembles the counterion condensation theory suggested by Manning [63] (quoted from [60]). In counterion condensation theory the RDF discontinuity occurs due to the difference in calculating the potential inside and outside the ‘condensation’ volume.

The other method that might be employed to rationalized the experimentally derived RDF’s is to modify the Eqs. 4.12 and 4.14, where instead of calculating the electrostatic contribution per unit charge, both equations are modified to calculate the change of the electrostatic contribution to the thermodynamics (per DNA strand). In our modifica-

tion we allow each DNA monomer to interact with the Na^+ ions that are associated with neighboring monomers units. These "neighbors" extend to the whole DNA chain.

In the following, we discuss modification of Eq. 4.12 and 4.14. The first term in Eq. 4.12 is the surface potential of the monomer. Because the polyelectrolyte model contains same monomer charge and dimension, the total surface potential is the summation of N monomers. The parameter $\xi = l_B/b$ in Eq. 4.12 has to be changed to become l_B/Nb to denote the unit over the polyelectrolyte length. The second term is the electrostatic energy due to the interaction with α -type ions.

Summing up the electrostatic energy of each monomer- α -type particles, we have:

$$\{E_{el}/k_B T\}/(\text{unit length}) = N \frac{1}{2l_B} \frac{l_B}{Nb} \phi_M + \sum_{i=1}^N \frac{1}{2} \int_{r_M}^{R_c} 2\pi r \sum_{\alpha} (z_{\alpha} \rho_{\alpha}^0 g_{\alpha}^i(r)) \phi(r) dr, \quad (4.15)$$

where $\rho_{\alpha}(r) = \rho_{\alpha}^0 g_{\alpha}^i(r)$. The $g_{\alpha}^i(r)$ is the RDF of the α type particle to the monomer with index i . The radial distribution function obtained from simulation is the average RDF of the RDF due to each of the monomers. Thus $g_{\alpha}(r) = \frac{\sum_{i=1}^N g_{\alpha}^i(r)}{N}$. Expanding Eq. 4.13 yields:

$$\frac{\{E_{el}/k_B T\}}{(\text{unit length})} = \frac{N}{2} \frac{1}{Nb} \phi_M + \sum_{i=1}^N \frac{1}{2} \left[\int_{r_M}^{R_c} 2\pi r z_1 \rho_1^0 g_1^i(r) \phi(r) dr \right. \quad (4.16)$$

$$\left. + \int_{r_M}^{R_c} 2\pi r z_2 \rho_2^0 g_2^i(r) \phi(r) dr + \dots \right]$$

$$= \frac{N}{2} \frac{1}{Nb} \phi_M + \frac{1}{2} \left[\int_{r_M}^{R_c} 2\pi r z_1 \rho_1^0 \left(\sum_{i=1}^N g_1^i(r) \right) \phi(r) dr \right. \quad (4.17)$$

$$\left. + \int_{r_M}^{R_c} 2\pi r z_2 \rho_2^0 \left(\sum_{i=1}^N g_2^i(r) \right) \phi(r) dr + \dots \right]$$

$$= \frac{N}{2} \frac{1}{Nb} \phi_M + \frac{1}{2} \left[\int_{r_M}^{R_c} 2\pi r z_1 \rho_1^0 N g_1(r) \phi(r) dr \right. \quad (4.18)$$

$$\left. + \int_{r_M}^{R_c} 2\pi r z_2 \rho_2^0 N g_2(r) \phi(r) dr + \dots \right]$$

$$= \frac{N}{2} \frac{1}{Nb} \phi_M + \frac{N}{2} \int_{r_M}^{R_c} 2\pi r \left(\sum_{\alpha} z_{\alpha} \rho_{\alpha}^0 g_{\alpha}(r) \right) \phi(r) dr . \quad (4.19)$$

In order to get the reduced electrostatic energy (E_{el}/kT) per unit polyelectrolyte length, we need to include the the Nb contour length factor in the denominator.

Thus we arrive at the final expression for the modified electrostatic energy in a cylindrical cell:

$$\frac{\{E_{\text{el}}/k_B T\}}{(\text{unit length})} = \left[\frac{N}{2} \phi_M + \frac{N^2 b}{2} \int_{r_M}^{R_c} 2\pi r \left(\sum_{\alpha} z_{\alpha} \rho_{\alpha}^0 g_{\alpha}(r) \right) \phi(r) dr \right] / Nb . \quad (4.20)$$

To modify the electrostatic entropy, we begin summing up the electrostatic entropy for all monomers along the chain:

$$\frac{\{S_{\text{el}}/k_B\}}{(\text{unit length})} = \sum_{i=1}^N \left[- \int_{r_M}^{R_c} 2\pi r \sum_{\alpha} \rho_{\alpha}^0 g_{\alpha}^i(r) \ln g_{\alpha}^i(r) \right] dr \quad (4.21)$$

$$= - \int_{r_M}^{R_c} 2\pi r \sum_{\alpha} \rho_{\alpha}^0 \left(\sum_{i=1}^N g_{\alpha}^i(r) \ln g_{\alpha}^i(r) \right) dr , \quad (4.22)$$

where $\rho_{\alpha}(r) = g_{\alpha}(r) \rho_{\alpha}^0$. The summation in Eq. 4.22 contains a natural logarithm, which is difficult to solve. Thus we make an assumption that the RDF of individual monomers is the same over the chain. This approximation is reasonable for a very long chain because we can neglect the end effects. In our model which simulates a very long chain containing 360 monomers with contour length 7344 Å, this approximation is reasonable. Thus:

$$\frac{\{S_{\text{el}}/k_B\}}{(\text{unit length})} = - \int_{r_M}^{R_c} 2\pi r N \sum_{\alpha} (\rho_{\alpha}^0 g_{\alpha}(r) \ln g_{\alpha}(r)) dr . \quad (4.23)$$

As with the electrostatic energy, we need to include the contour length Nb in the denominator to get the entropy term ($TS_{\text{el}}/k_B T$) per chain length:

$$\frac{\{S_{\text{el}}/k_B\}}{(\text{unit length})} = \left[- \int_{r_M}^{R_c} 2\pi r N^2 b \sum_{\alpha} (\rho_{\alpha}^0 g_{\alpha}(r) \ln g_{\alpha}(r)) dr \right] / Nb . \quad (4.24)$$

In Eq. 4.20 and 4.24, only the formula inside the square brackets will be used in calculations since we are interested in total energies.

The $\phi(r)$ electrostatic potential is obtained by solving the for PB equation for a cylindrical

cell [60]:

$$\frac{d^2(z\phi)}{dr^2} + \frac{1}{r} \frac{d(z\phi)}{dr} = \kappa^2 \sinh(z\phi), \quad (4.25)$$

where $\phi = e\psi/k_B T$ is the reduced electrostatic potential.

This differential equation and others [Korolev et al. [58], Eq. (10)] must be solved iteratively. In order to calculate the entropy, energy and free energies for our modified equations, we need to numerically compute ϕ_M and $\phi(r)$. This is a separated and dilated study in differential equations which will not be presented here. On the other hand, in order to illustrate computational ease of our equations, we shall use the Debye-Huckel potential which is known in closed form as a function distance from the central ion. Thus instead of using a cylindrical cell, we consider the spherical cell as more appropriate for our present work. Another reason is that the computation of RDF in simulations is based on the density within a spherical volume, which makes the Debye-Huckel and any spherically symmetrical potential convenient. In the future, a comparisonal study of cylindrical and spherical cell would be conducted, where Eq. 4.25 would be solved to allow for the intended calculations.

4.4.2 Modification of the Cylindrical to the Spherical Cell

In order to change the electrostatic contribution equation element from the cylindrical cell to spherical cell, we need to change the integration from over the volume of the annulus, $2\pi r dr L$, to that over the shell volume $4\pi r^2 dr$. Consequently, we will change the electrostatic energy and energy unit from the reduced energy and entropy per unit length (in a cylindrical cell) to become reduced energy and entropy (in a spherical cell). This is because in the spherical cell model we measure the total electrostatic changes in the spherical cell volume.

The formula of the electrostatic energy and entropy of polyelectrolyte-ion a system in

spherical cell are:

$$E_{\text{el}}/k_{\text{B}}T = \frac{N}{2}\phi_M + \frac{N}{2} \int_{r_M}^{R_n} 4\pi r^2 \left(\sum_{\alpha} z_{\alpha} \rho_{\alpha}^0 g_{\alpha}(r) \right) \phi(r) dr \quad (4.26)$$

$$S_{\text{el}}/k_{\text{B}} = - \int_{r_M}^{R_n} 4\pi r^2 N \sum_{\alpha} \rho_{\alpha}^0 g_{\alpha}(r) \ln g_{\alpha}(r) dr . \quad (4.27)$$

Then we will get the Helmholtz electrostatic free energy (F) with the expression:

$$F_{\text{el}} = E_{\text{el}} - TS_{\text{el}} . \quad (4.28)$$

In our work, we change the limits for integration of electrostatic energy and entropy. Instead of modelling the limit due to the distance radius (R_c) of a cylinder or a sphere, where the volume of the cylinder or sphere is defined by the DNA concentration, we limit the integration limit R_n as the distance obtained after we normalize the RDF. The normalization follows the expression:

$$n_{\alpha} = \int_0^{R_n} 4\pi r^2 g_{\alpha}(r) \rho_{\alpha}^0 dr , \quad (4.29)$$

where n_{α} is the total number of particles α .

c_{salt} (mM)	$E_{\text{el}}/(k_{\text{B}}T)$	$TS_{\text{el}}/(k_{\text{B}}T)$	$F_{\text{el}}/(k_{\text{B}}T)$	DNA–Na ⁺ Coord. Number ($n_{\text{DNA}}^{\text{Na}}$)
0.0	-72265.7	-15036.7	-57228.9	9.37
0.001	-69534.7	-14980.1	-54554.6	9.62
0.0025	-72861.1	-15468	-57393.1	9.87
0.005	-69718.7	-15495.5	-54223.1	9.89
0.01	-68420.5	-16271.3	-52149.1	9.96
0.025	-56314.2	-14695.9	-41618.4	10.75
0.05	-38843.0	-12918.2	-25924.8	11.86
0.075	-33044.9	-12357.2	-20687.7	12.35
0.15	-7301.54	-9519.26	2217.71	15.24
0.25	-5690.49	-9348.75	3658.26	15.41

Table 4.3: Electrostatic profile (Spherical Cell) of DNA–NCP interaction and DNA–Na⁺ coordination number. DNA concentration 0.005 mg/ml

c_{salt} (mM)	$E_{\text{el}}/(k_{\text{B}}T)$	$TS_{\text{el}}/(k_{\text{B}}T)$	$F_{\text{el}}/(k_{\text{B}}T)$	DNA–Na ⁺ Coord. Number ($n_{\text{DNA}}^{\text{Na}}$)
0.0	-23608.1	-2559.41	-21048.7	8.86
0.01	-27293.0	-2335.91	-24957.1	9.04
0.1	-25589.8	-1677.85	-23912	8.89
1.0	-20212.3	-669.775	-19542.5	10.27
2.5	-13256.8	15.1975	-13272	11.85
7.5	-5963.7	967.279	-6930.98	14.28
10.0	-6927.31	358.699	-7286.01	13.58
25.0	-2020.57	1292.37	-3312.94	16.72
50.0	-1280.49	1341.04	-2621.53	18.15
75.0	-870.718	1570.49	-2441.2	20.58
100.0	-757.847	1728.26	-2486.11	21.60

Table 4.4: Electrostatic profile (Spherical Cell) of DNA–NCP interaction and DNA–Na⁺ coordination number. DNA concentration 2.0 mg/ml

Table 4.3 and 4.4 give the numerical result of the electrostatic energy, entropy and free energy of DNA–NCP interaction at DNA concentration 0.005 mg/ml and 2.0 mg/ml. Those tables also includes the DNA coordination number of Na⁺ ion to DNA. We graph these tables in Fig 4.6(a,b). The results for both DNA concentrations show that the electrostatic energy of DNA–NCP interaction increases with the addition of salt. It indicates that the coulomb attractive potential between DNA and NCP weakens as the salt amount increases. We attribute the weakening of the DNA–NCP attraction to the stronger screening effect from salt ions. Table 4.3 and 4.4 show that as the salt amount increases, the DNA coordination number to Na⁺ also increases. Thus besides the screening effect from diffused ions, the addition of Na⁺ into the solution will compete with the NCP binding to the DNA negatively charged surface. This Na⁺–NCP competition also plays a role in weakening the DNA–NCP interaction.

From the entropy point of view, addition of salt will slightly increase the electrostatic entropy of DNA–NCP interaction. The entropy escalation implies that the NCP position will be more spread out in the system rather than being bound at the DNA surface. The entropy increase indicates a smaller measure of DNA–NCP wrapping

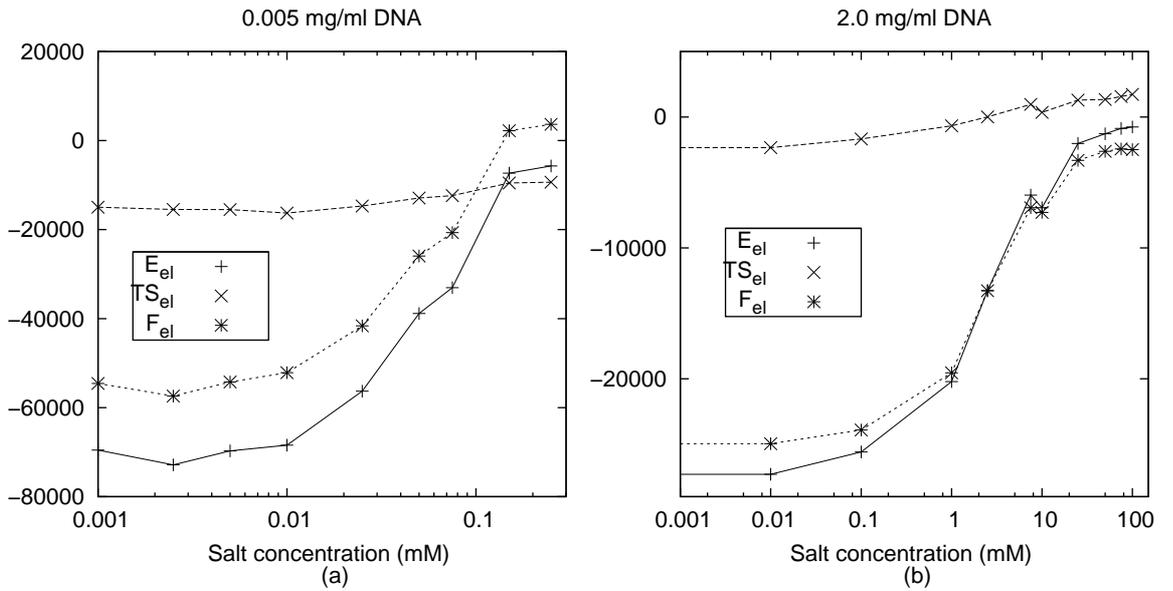


Figure 4.6: Electrostatic profile of DNA–NCP interaction. The unit of E_{el} , TS_{el} and F_{el} is in $k_B T$

In this study the DNA–NCP electrostatic free energy reflects the DNA–NCP interactions at equilibrium. The availability of other salt particles in the solution is responsible for the interaction parameters (condition) for the DNA and NCP particles. Lower free energies indicate the preferred state of DNA–NCP interaction. From the graphs in Fig. 4.6, the electrostatic energy contribution is more significant to the entropy contribution of the free energy. Thus electrostatic energy between DNA and NCP has a more important role than entropy in determining the nature of DNA–NCP interaction.

The comparison of Na^+ ion coordination number to the DNA in simulations with and

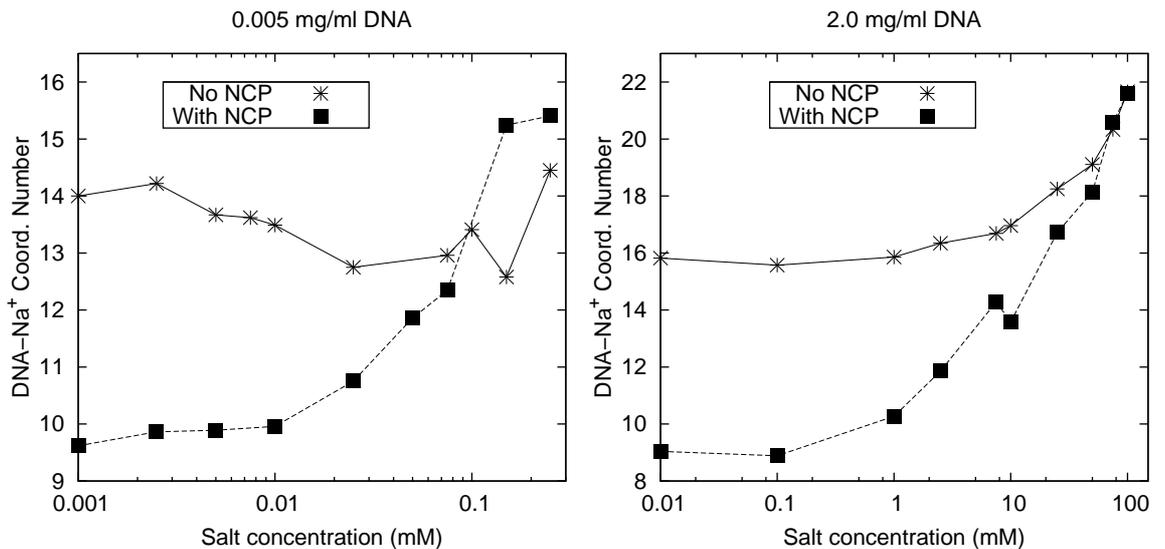


Figure 4.7: Coordination number of Na^+ ions about DNA in simulation with NCP and without NCP available in the system. The error estimation for each points is ± 0.5

without NCP is depicted in Fig. 4.7. From simulations, it is found that the NCP tends to bind to the DNA polyelectrolyte at low salt concentration. Figure 4.7 substantiates this observation in that at low salt concentrations, a smaller amount of Na^+ is bound at the DNA surface. The large positive charge of NCP will repel the Na^+ from the DNA surface. Recall that at low concentration the Debye screening length is very large leading to almost pure Coulombic repulsion between NCP and Na^+ . At higher salt concentrations, the stronger screening effect weakens the DNA–NCP interaction and at a certain concentration, Na^+ wins the competition in cation–DNA binding. Thus at high salt concentrations, the NCP concentration will not affect DNA– Na^+ interaction.

For a comparison to another study pertaining the DNA-NCP conformation, we present

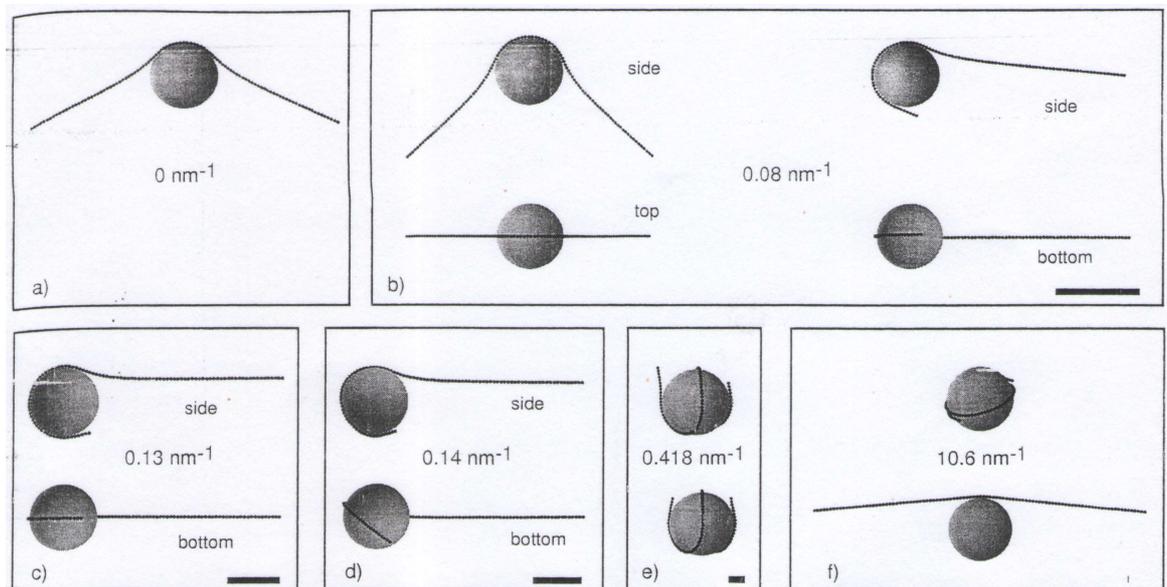


Figure 4.8: Original figure from [62]. DNA-NCP configuration obtained by minimizing the free energy (Eq. 4.30) for a fixed NCP charge $Z = 40$ at increasing salt concentrations. Bars in the lower right are the respective screening lengths. (a) No added salt ($\kappa = 0$). (b) Salt concentration 0.6 mM ($\kappa = 0.08\text{nm}^{-1}$) and 1.6 mM ($\kappa = 0.13\text{nm}^{-1}$). (c) Salt concentration 16.6 mM ($\kappa = 0.418\text{nm}^{-1}$). (d) Salt concentration 10.6 M ($\kappa = 10.6\text{nm}^{-1}$)

an interesting result of the numerical simulation done by [62]. Kunze and Netz [62] determined the equilibrium state of the DNA-NCP complex by calculating the minimum free energy of the DNA-NCP complex. In their work, the free energy expression for a

fixed DNA-NCP configuration is

$$\begin{aligned} \frac{F}{k_B T} = & \frac{L_P}{2} \int_0^L ds \ddot{\mathbf{r}}^2(s) - \frac{l_B Z \tau}{1 + \kappa R} \int_0^L ds \frac{e^{-\kappa(|\mathbf{r}(s)| - R)}}{|\mathbf{r}(s)|} \\ & + l_B \tau^2 \int_0^L ds \int_s^L ds' \frac{e^{-\kappa|\mathbf{r}(s) - \mathbf{r}(s')|}}{|\mathbf{r}(s) - \mathbf{r}(s')|} \end{aligned} \quad (4.30)$$

where they utilized $L_P = 300 \text{ \AA}$ as the bare DNA persistence length, $\mathbf{r}(s)$ is a vector describing the DNA configuration, $\tau = 2/3.4 \text{ \AA}$ is the linear DNA charge density, $L = 500 \text{ \AA}$ is the DNA polymer length, $Z = 40$ is the NCP charge and $R = 50 \text{ \AA}$ is the NCP radius. The salt concentration is included in the Debye-Huckel parameter κ . Since there is no entropic term given in Eq. 4.30, we consider that Eq. 4.30 is more appropriate to represent the energy of the DNA-NCP conformation instead of its free energy. But we keep the free energy notation from the source article. The snapshots of equilibrium DNA-NCP complex from minimizing Eq. 4.30 in different salt concentrations are shown in Fig. 4.8. The minimization seems to refer to defined order parameters for any one chosen conformation.

From Fig. 4.8, there are partial unwrapping of the DNA-NCP complex at zero and low salt concentration which is not observed in our simulation. Kunze and Netz [62] attribute the DNA-NCP unwrapping at zero and low salt concentration to the strong Coulomb repulsion between DNA beads. In contrast, in our simulation the DNA-NCP coiling is more pronounced at low salt. The NCP charge which is explained by [62] in Fig 4.8 is $Z = +40$ whilst we use $Z = +150$. This could be one factor why in [62] the net DNA-DNA repulsive force is larger than the net DNA-NCP coiling force at low salt concentration. However further exploration is needed to justify this interesting phenomenon.

4.5 Snapshots

Fig. 4.9 and 4.10 give some snapshots of DNA and NCP in simulation. In Fig. 4.9, the DNA concentration is 0.005 mg/ml, and in Fig 4.10 it is 2.0 mg/ml. The NCP is represented as green spheres. For clarity, the salt ion images are excluded. These snapshots support the electrostatic free energy data. As depicted in both figures, when the salt concentration increases, the NCPs will tend to be freed from the DNA wrapping due to

greater screening effect and Na^+ competition. From the free energy data (Table 4.3 and 4.4), it is clear that the most wrapped NCP–DNA structure corresponds to that with the lowest free energies, which correspond also to low salt concentrations.

It is evident that salt concentration at which the DNA uncoils from the NCP is a func-

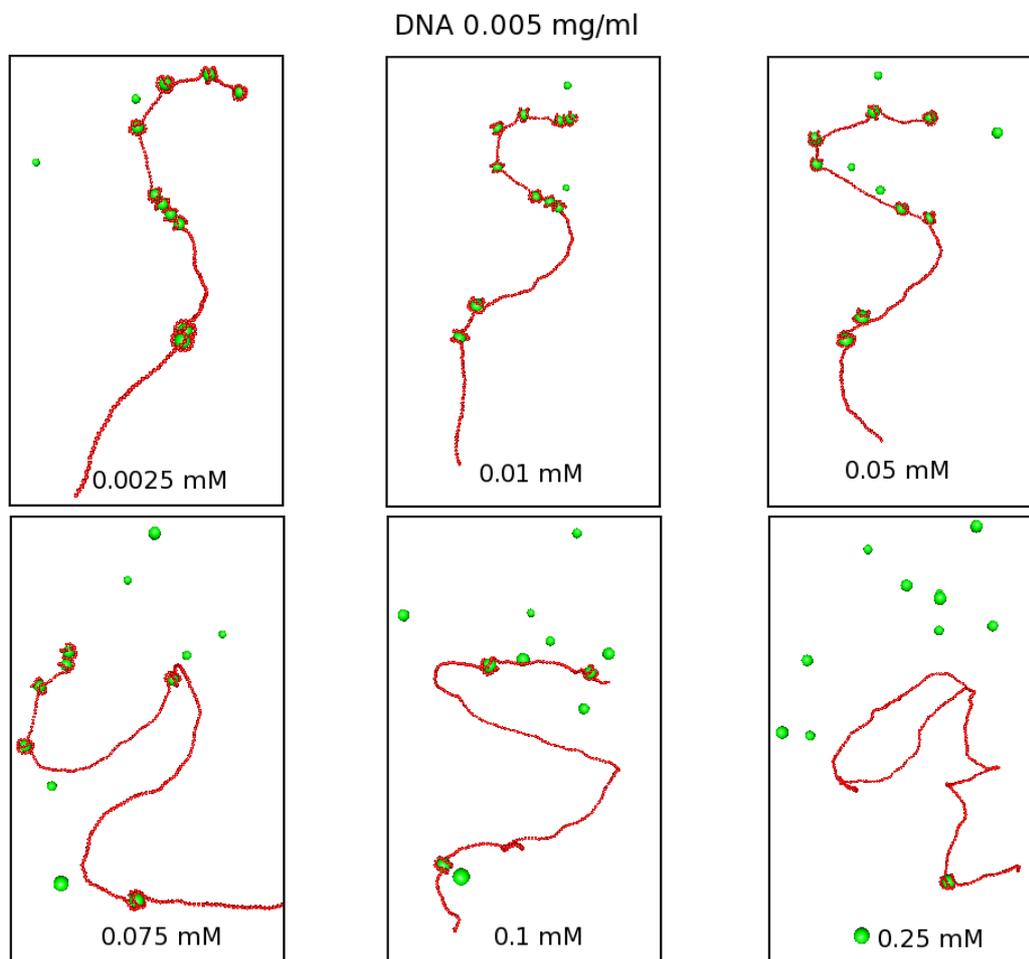


Figure 4.9: Snapshots of DNA–NCP particles at different salt concentrations. DNA concentration 0.005 mg/ml.

tion of DNA–NCP concentration. At 0.005 mg/ml, the DNA–NCP unwrapping starts at salt concentration ~ 0.075 mM. While at DNA concentration 2.0 mg/ml, DNA–NCP unwrapping starts at salt concentration ~ 10 mM. Thus choosing the appropriate DNA and NCP concentration for studying the DNA–NCP coiling mechanism for variable salt concentration is critical in order to determine any type of phase diagram and the associated properties of the phases. Fig. 4.11 gives some snapshots of the DNA–NCP conformations at different time intervals (reduced unit). After some time, the energy and the DNA–NCP conformation in the system are about fixed indicating the system is in equilibrium.

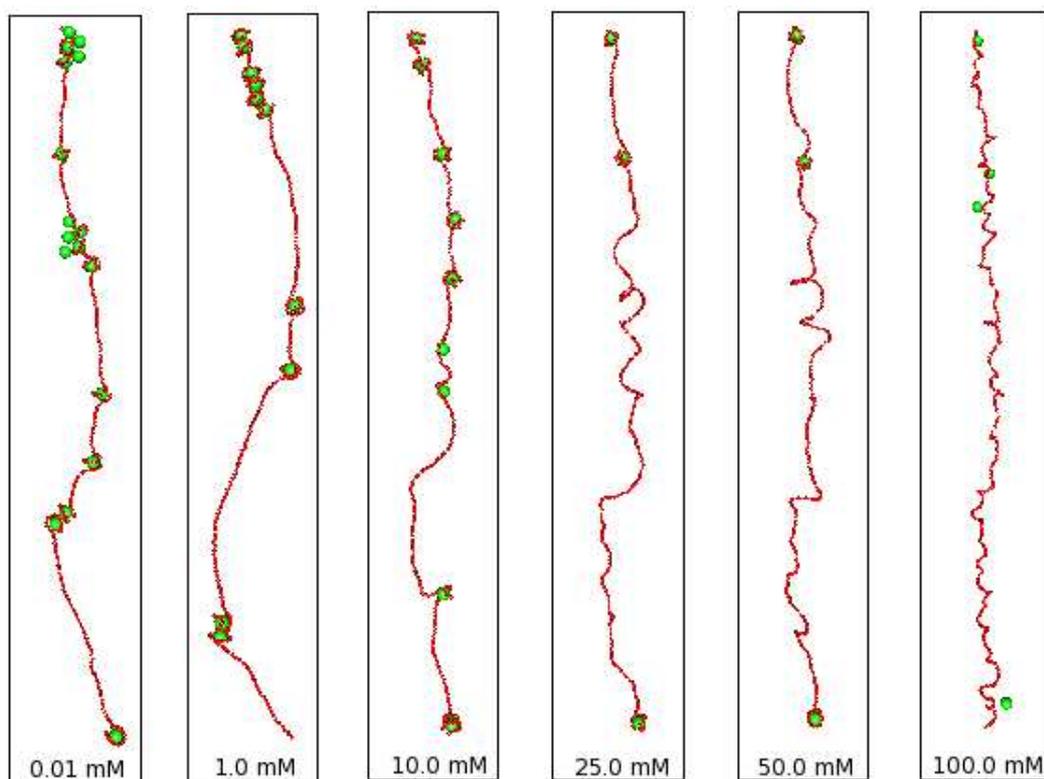


Figure 4.10: Snapshots of DNA–NCP particles at different salt concentrations. DNA concentration 2.0 mg/ml.

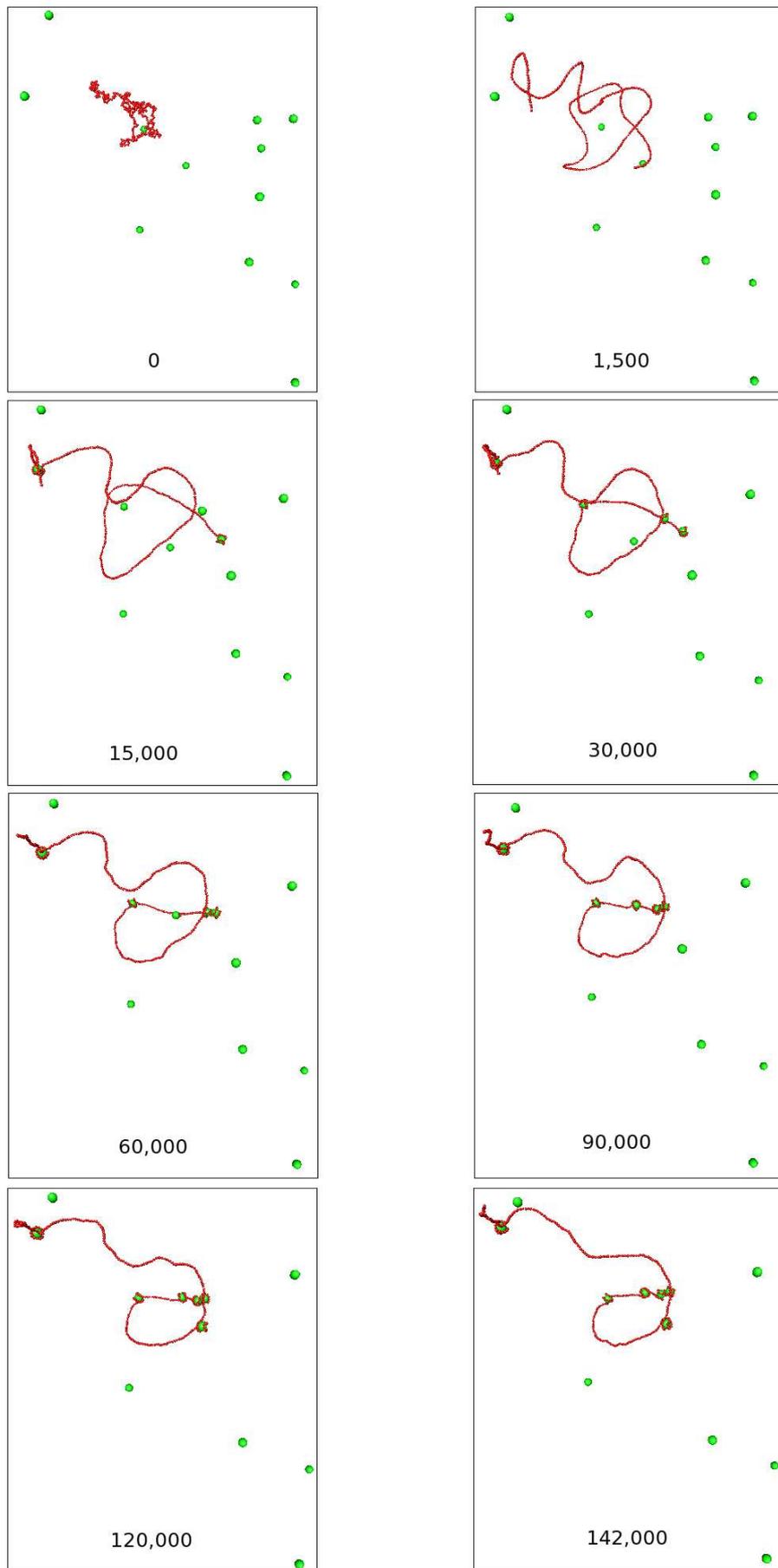


Figure 4.11: DNA-NCP conformations in different time intervals. DNA concentration 0.005 mg/ml, salt concentration 0.075 mM.

Chapter 5

Conclusion

The extent and intensity of the simulation studies was considerable since relatively little known of the conformational dynamics of the DNA–NCP complex in the presence of salt. Hence many unknown variables presented themselves at the outset of the work and we had to survey a very broad terrain of simulation conditions. The above learning experiences allow us to choose the appropriate conditions to verify both qualitatively and quantitatively the following relative to our model.

1. Increasing the salt concentration increases the DNA flexibility indicated by the decrease of the DNA persistence length
2. Our equation for calculating the persistence length of any stiff polyelectrolyte preserves the persistence length as relatively constant at very low salt concentrations whereas in other theories, this length tends to infinity which does not accord with reality
3. Our "ionic bridging effect" allows for the possibility that a charged polyelectrolyte has a smaller persistence length than the uncharged polymer
4. The ionic bridging effect plays an equally important role in reducing the DNA persistence length as the ionic screening effect
5. Molecular simulations are able to directly demonstrate the existence of the ionic bridging phenomenon

6. The persistence length of a stiff neutral chain from WLC theory calculation only holds up to a certain value of the chain bending modulus
7. The ions included for calculating the ionic screening parameter κ are the free mobile ions (salt and counterion) and the macroionic contributions are negligible
8. The original Poisson-Boltzmann cylinder cell model cannot account for the counterion condensation at the DNA monomer neighbors
9. Increasing the NaCl concentration increases the coordination number of Na^+ about the DNA chain
10. Increasing the salt concentration weakens the tendency for DNA–NCP wrapping
11. The electrostatic energy term gives a more significant contribution than the entropic term to the electrostatic free energy change due to the fluctuation in the DNA–NCP wrapping.

It is evident from the list above that this piece of research has opened up at least 11 research direction of topics in critical areas of biophysical as polyelectrolyte research. The investigation of any of the above topics would undoubtedly lead to even greater elucidations connecting atomistic interactions with macroscopic manifestations.

Appendix A

Reduced units

Reduced units are obtained by converting some constant values to a preferred constant (e.g. 1.00) or associating some constant values to other variables. Following the convention, any variables in reduced unit will be appended by the '*' symbol.

A.1 Length (L)

The unit of length in our simulation is the Bjerrum length (l_B) defined as the distance where the coulomb energy equals to the thermal energy $k_B T$. Then

$$l_B = \frac{e^2}{4\pi\epsilon_0\epsilon_r k_B T}, \quad (\text{A.1})$$

where $e = 1.602176 \times 10^{-19}$ C, $\epsilon_0 = 8.854187817 \times 10^{-12}$ C²J⁻¹m⁻¹ is the vacuum permittivity, ϵ_r the relative permittivity or dielectric constant is defined as $\epsilon_r = \epsilon_s/\epsilon_0$. We arrive at the Bjerrum length of water as 7.13 Å at 300 K which is the value that we shall subsequently use for all our simulation runs. Thus:

$$L^* = L/l_B. \quad (\text{A.2})$$

A.2 Temperature

We use the reduced unit temperature $T^* = \epsilon_{LJ}/k_B T$ where ϵ_{LJ} is a fixed energy increment. In our work $T^* = 1$ is equivalent to 300 K.

A.3 Energy

The reduced unit energy (E^*) is equivalent to the Lennard-Jones (LJ) energy scale, ϵ_{LJ} (in 2.3, chapter 2), where $\epsilon_{LJ} = k_B T / T^*$. Thus

$$\epsilon_{LJ} = \frac{1.3807 \times 10^{-23} \times 300}{1} = 4.142 \times 10^{-21} \text{J}. \quad (\text{A.3})$$

Furthermore

$$E^* = E / \epsilon_{LJ}. \quad (\text{A.4})$$

Clearly from the above $\epsilon_{LJ}^* = 1$.

A.4 Mass

To use reduced units for particles mass, the particle with the lightest mass has reduced unit mass, $m^* = 1$. where the mass of this standard is m_s . For any other mass, we have $m^* = m / m_s$. The particles simulated here are DNA monomers (each monomer contains 6 bp), nucleosome core particles (histone octamers) and Na^+ and Cl^- . From the literature ([1][64]), we obtain:

- Mass of Na = 3.819×10^{-26} kg/particle
- Mass of Cl = 5.889×10^{-26} kg/particle
- Mass of DNA bead (average) = 61262×10^{-26} kg/monomer
- Mass of NCP = 18.02668×10^{-23} kg (per histone octamers).

Because Na^+ has the smallest mass, this mass will be used as the standard (m_s). Thus:

- $m_{\text{Na}^+}^* = 1.0$
- $m_{\text{Cl}^-}^* = 1.54$
- $m_{\text{DNA monomer}}^* = 160.41$
- $m_{\text{NCP}}^* = 4270.26$.

A.5 Velocity

In energy term, velocity is a part of the kinetic energy. From the previous definitions, we have

$$E_k = \frac{1}{2}mv^2 \Rightarrow E_k^* \epsilon_{LJ} = \frac{1}{2}m^* m_s v^2 \Rightarrow E_k^* = \frac{1}{2}m^* \left(\frac{m_s}{\epsilon_{LJ}} v^2 \right). \quad (\text{A.5})$$

By the invariance principle, for any reduced unit, the definition for kinetic energy remains the same. Hence

$$E_k^* = \frac{1}{2}m^*(v^*)^2. \quad (\text{A.6})$$

Comparing Eq. A.5 and A.6, we have

$$(v^*)^2 = \frac{m_s}{\epsilon_{LJ}} v^2 \quad (\text{A.7})$$

or

$$v^* = \sqrt{\frac{m_s}{\epsilon_{LJ}}} v. \quad (\text{A.8})$$

A.6 Time

The time variable (t) is always related to the velocity v and distance r as follows:

$$r = vt \Rightarrow r^* l_B = \sqrt{\frac{\epsilon_{LJ}}{m_s}} v^* t \Rightarrow r^* = v^* \sqrt{\frac{\epsilon_{LJ}}{m_s}} \frac{1}{l_B} t. \quad (\text{A.9})$$

By the invariance principle,

$$r^* = v^* t^*. \quad (\text{A.10})$$

By comparing (A.9) and (A.10), we derive

$$t^* = \frac{1}{l_B} \sqrt{\frac{\epsilon_{LJ}}{m_s}} t . \quad (\text{A.11})$$

From the known values of l_B , ϵ_{LJ} and m_s , Eq. A.11 yields

$$t = 2.165 \times 10^{-12} t^* .$$

If we used the reduced time-step 0.001 in simulation, its actual time-step in laboratory units would be 2.165×10^{-15} seconds or 2.165 femtoseconds.

A.7 Harmonic Bond Constant

The Hooke's law is used to model the bonding potential, given by

$$E_h = \frac{1}{2} k_h (\Delta x)^2 ,$$

where E_h is the harmonic oscillator energy ; $\Delta x = x_t - x_0$ where x_t is the interparticle distance and x_0 the equilibrium distance. Therefore

$$E_h^* \epsilon_{LJ} = \frac{1}{2} k_h (\Delta x^* l_B)^2 \Rightarrow E_h^* = \frac{1}{2} \frac{k_h l_B^2}{\epsilon_{LJ}} (\Delta x^*)^2 = \frac{1}{2} k_h^* (\Delta x^*)^2 . \quad (\text{A.12})$$

Thus, by the usual method of comparison,

$$k_h^* = \frac{k_h l_B^2}{\epsilon_{LJ}} . \quad (\text{A.13})$$

By submitting known values for the physical constant, we have:

$$= \frac{k_h (7.13 \times 10^{-10})^2}{4.14 \times 10^{-21}} = 122.794 \times k_h . \quad (\text{A.14})$$

A.8 Bending Angle Constant

Similarly, the bending angle energy is expressed by:

$$E_\theta = \frac{1}{2} k_\theta \theta^2 .$$

Since θ is dimensionless, we have $\theta = \theta^*$. Therefore

$$E_\theta^* \epsilon_{LJ} = \frac{1}{2} k_\theta (\theta^*)^2 \Rightarrow E^* = \frac{1}{2} \frac{k_\theta}{\epsilon_{LJ}} \theta^2 . \quad (\text{A.15})$$

Hence by the invariant principle,

$$k_\theta^* = \frac{k_\theta}{\epsilon_{LJ}} = \frac{k_\theta}{k_B T} T^* , \quad (\text{A.16})$$

and therefore

$$k_\theta^* = \frac{k_\theta}{k_B T} . \quad (\text{A.17})$$

A.9 Acceleration

The definition of acceleration for an infinitesimal time increment t is

$$a = \frac{V_2 - V_1}{t} = \frac{\sqrt{\frac{\epsilon_{LJ}}{m_s}} (V_2^* - V_1^*)}{l_B \sqrt{\frac{m_s}{\epsilon_{LJ}}} t^*} . \quad (\text{A.18})$$

Converting Eq. A.18 to reduced units for all the variables, implies

$$a = \frac{\epsilon_{LJ}}{l_B m_s} \frac{V_2^* - V_1^*}{t^*} = \frac{\epsilon_{LJ}}{l_B m_s} a^* . \quad (\text{A.19})$$

Thus:

$$a^* = \frac{l_B}{m_s \epsilon_{LJ}} a . \quad (\text{A.20})$$

A.10 Force

By the invariance principle, we can similarly write

$$F = ma = \frac{m^* m_s \epsilon_{LJ} a^*}{l_B m_s} = m^* a^* \frac{\epsilon_{LJ}}{l_B} = F^* \frac{\epsilon_{LJ}}{l_B} . \quad (\text{A.21})$$

Thus

$$F^* = F \frac{l_B}{\epsilon_{LJ}} . \quad (\text{A.22})$$

Appendix B

Tool Command Language (TCL)

Scripts

To run a molecular dynamics (MD) simulation with ESPResSo, the ESPResSo executable file will read the user's commands in a TCL script. In this appendix, two main TCL scripts for running MD simulation are given. Some codes are written (in either C++ or TCL) for analyzing the output of the MD run. The analysis centres about calculations to determine such parameters as the persistence length, electrostatic energy and entropy, coordination number and harmonic stretching constant. Various approximations are used in the calculation (e.g. the Percus-Yevick and hypernetted chain approximation). Several equations are also utilized (e.g. Debye-Huckel-Poisson-Boltzmann, Ornstein-Zernike and modified Grosberg equations). Those codes are not loaded in this thesis.

B.1 Equilibrium Run

The TCL script in this section is used to derive a system in equilibrium. This script contains the system set up, particles properties, force fields assignment, interaction parameters, trajectory file generator, system checkpoint/recorder for the next run, and the energy measurement.

The script is reproduced below:

```

set t1 [clock second]
puts "[code_info]"
##### PROCEDURE #####
# Add configuration in TXYZ (Tinker) format to the trajectory file "f_tinker"
# Below puts all particle including salts in trajectory file
proc conf_tinker { f_tinker } {
  global box_l n_MD n_part chem_typ atom_typ
  puts $f_tinker "$n_part after [setmd time] i.u., BOX: $box_l"
  for {set j 0} {$j < [expr $n_MD -1]} {incr j} {
    puts $f_tinker "[expr $j+1] $chem_typ($j) [part $j print pos] $atom_typ($j) [expr $j+2] "
  }
  for {set j [expr $n_MD-1]} {$j < $n_part} {incr j} {
    puts $f_tinker "[expr $j+1] $chem_typ($j) [part $j print pos] $atom_typ($j)"
  }
}
# Add configuration in XMOL format to the trajectory file "f_xmol"
# Below only puts DNA Monomers and NCP in trajectory file
proc confxmol { f_xmol } {
  global box_l chem_typ n_MD n_NCP
  puts $f_xmol [expr int([expr $n_MD + $n_NCP])]
  puts $f_xmol " after [setmd time] i.u., BOX: $box_l $box_l $box_l"
  for {set j 0} {$j < [expr $n_MD + $n_NCP]} {incr j} {
    puts $f_xmol " $chem_typ($j) [part $j print pos]"
  }
}
proc DNAToCENTER { } {
  global box_l nt_MD pid_MD
  # assumes the same mass for each group of particles in a molecule etc..
  set xcom 0.
  set ycom 0.
  set zcom 0.
  # set ii [expr $i + 1]
  # loop over molecules
  # pid_MD=0 nt_MD=total monomers, j will be the pid
  for {set j $pid_MD} {$j < $nt_MD} {incr j} {
    # compute center-of-mass for each molecule
    set xcom [expr $xcom + [lindex [part $j print pos] 0 ] ]
    set ycom [expr $ycom + [lindex [part $j print pos] 1 ] ]
    set zcom [expr $zcom + [lindex [part $j print pos] 2 ] ]
  }
  set xcom [ expr $xcom / $nt_MD ]
  set ycom [ expr $ycom / $nt_MD ]
  set zcom [ expr $zcom / $nt_MD ]
  # Here we already get the COM polymer.

```

```

# Below moving all particle, where COM polymer will be in the middle of box
for {set j $pid_MD} {$j < $nt_MD} { incr j } {
set xnew [expr [lindex [part $j print pos] 0] - $xcom + 0.5*$box_l]
set ynew [expr [lindex [part $j print pos] 1] - $ycom + 0.5*$box_l]
set znew [expr [lindex [part $j print pos] 2] - $zcom + 0.5*$box_l]
part $j pos $xnew $ynew $znew
}}
##### Length Units= Bjerrum Length #####
set l_b 7.13
set l_unit $l_b
#####PARAMETERS#####
set title "24type1"
#first, set polymer concentration you wish
set polymer_concentration 0.005
# bead/monomer weight and volume scale =( ... x 10^-20) gr and ml
set bead_weight 0.61262
set polymer_weight [expr $bead_weight*360]
set v_solvent [expr ($polymer_weight*1000)/$polymer_concentration]
set box_iori [expr pow(($v_solvent/100000.),0.33333)*1000.]
puts "polymer concentration = $polymer_concentration mg/ml"
puts "box length = $box_iori Angstrom"
## Simulation Box
set shield 1.
#volume box (v_box) below later for calculating salt amount
set v_box [expr pow($box_iori,3)]
set box_l [expr $box_iori/$l_unit]
setmd box_l $box_l $box_l $box_l
## Thermostat, here use thermostat for NVT type
set gamma 1.0
set temp 1.0
integrate set nvt
thermostat langevin $temp $gamma
set time_step 0.015
set skin 0.4
setmd time_step $time_step
setmd skin $skin
setmd temp
puts " "
puts "===== "
puts "= 360-chain polymer ="
puts "===== "
puts " "

```

```

#### Particles ####
##DNA
# num of DNA
set n_D 1
# num of monomers per DNA
set n_MD 360
# total monomers
set nt_MD [expr $n_D*$n_MD]
# mass of monomers
set m_MD 160.41
# radius of monomers (l_unit*A), equal to the width o DNA double helix
set r_MD [expr 10./$l_unit]
# radius of monomers soft core (effective radii)
set r_sMD [expr 2./$l_unit]
# radius of monomers hard core radii (ineffective radii)
set r_hMD [expr $r_MD-$r_sMD]
# charge of monomers
set q_MD -12
# total charge of monomers
set q_totalMD [expr $q_MD*$n_MD*$n_D]
# bond length between two nucleus, equal to 6*3.4 A
set l_MD [expr 20.4/$l_unit]
# type of monomers
set type_MD 0
# start monomer pid, for creating id of the first monomers
set pid_MD 0
# creating DNA chem_typ for gopenmol needs
for {set i $pid_MD} {$i < [expr $pid_MD+$nt_MD]} {incr i} {set chem_typ($i) "O" ;
set atom_typ($i) "40" }
## NCP
# Num of NCP
set n_NCP 12
# mass
set m_NCP 4270.26
# radius
set r_NCP [expr 35./$l_unit]
# radius of NCP soft core (effective radii)
set r_sNCP [expr 2./$l_unit]
# radius of NCP hard core radii (ineffective radii)
set r_hNCP [expr $r_NCP-$r_sNCP]
# charge
set q_NCP 150

```

```

# total charge of NCP
set q_totalNCP [expr $q_NCP*$n_NCP]
# type of NCP
set type_NCP 1
# start NCP pid, for creating id of the first NCP
set pid_NCP [expr $pid_MD+$nt_MD]
# creating NCP chem_typ for gopenmol needs
for {set ii $pid_NCP} {$ii < [expr $pid_NCP+$n_NCP]} {incr ii} {set chem_typ($ii)
"C" ; set atom_typ($ii) "14"}
## Calculating additional ion for neutralizing system
set q_add [expr -1*($q_totalMD+$q_totalNCP)]
if {$q_add > 0} {
set addNa $q_add
set addCl 0
} elseif {$q_add < 0} {
set addNa 0
set addCl [expr abs($q_add)]
} else {
set addNa 0
set addCl 0
}
}
## SALT
# Just put your desired salt concentration, in mM unit
set c_salt 0.25
set n_salt [expr int($c_salt*602e-09*$v_box)]
# (Na+1)
set n_Na [expr $n_salt+$addNa]
set m_Na 1.
###set r_Na [expr 0.98/$l_unit]
set r_Na [expr 2./$l_unit]
set q_Na 1
set q_totalNa [expr $q_Na*$n_Na]
set type_Na 2
set pid_Na [expr $pid_NCP+$n_NCP]
# creating Na chem_typ for gopenmol needs
for {set j $pid_Na} {$j < [expr $pid_Na+$n_Na]} {incr j} {set chem_typ($j) "Na" ; set
atom_typ($j) "81"}
# (Cl-1)
set n_Cl [expr $n_salt+$addCl]
set m_Cl 1.54
###set r_Cl [expr 1.81/$l_unit]
set r_Cl [expr 2./$l_unit]

```

```

set q_Cl -1
set q_totalCl [expr $q_Cl*$n_Cl]
set type_Cl 3
set pid_Cl [expr $pid_Na+$n_Na]
# creating Cl chem_typ for gopenmol needs
for {set jj $pid_Cl} {$jj < [expr $pid_Cl+$n_Cl]} {incr jj} {set chem_typ($jj) "Cl" ; set
atom_typ($jj) "93"}
## particle amount
set n_part [expr $nt_MD+$n_NCP+$n_Na+$n_Cl]
### Interaction ###
## (1) Between type 0 & 0 ##
# Lennard-Jones
set lj_eps00 1.0
set lj_sig00 [expr 2*$r_sMD]
set lj_rcut00 [expr 1.122462*$lj_sig00]
set lj_cshift00 [calc_lj_shift $lj_sig00 $lj_rcut00]
set lj_roff00 [expr $r_hMD+$r_hMD]
# Harmonic stiffer bonds
set r_spring 20.3417956
set harm_r [expr $r_spring/$l_unit]
set harm_k 2500.
# Angle bending (DNA persistence length)
# Bond angle
set bend_k [expr 500./20.4]
set bend_phi0 [PI]
puts $bend_phi0
## (2) Between type 0 & 1 ##
# Lennard-Jones
set lj_eps01 1.0
set lj_sig01 [expr $r_sMD+$r_sNCP]
set lj_rcut01 [expr 1.122462*$lj_sig01]
set lj_cshift01 [calc_lj_shift $lj_sig01 $lj_rcut01]
set lj_roff01 [expr $r_hMD+$r_hNCP]
## (3) Between type 0 & 2 ##
# Lennard-Jones
set lj_eps02 1.0
set lj_sig02 [expr $r_sMD+$r_Na]
set lj_rcut02 [expr 1.122462*$lj_sig02]
set lj_cshift02 [calc_lj_shift $lj_sig02 $lj_rcut02]
set lj_roff02 [expr $r_hMD+0.]
## (4) Between type 0 & 3 ##
# Lennard-Jones

```

```

set lj_eps03 1.0
set lj_sig03 [expr $r_sMD+$r_CI]
set lj_rcut03 [expr 1.122462*$lj_sig03]
set lj_cshift03 [calc_lj_shift $lj_sig03 $lj_rcut03]
set lj_roff03 [expr $r_hMD+0.]
## (5) Between type 1 & 1 ##
# Lennard-Jones
set lj_eps11 1.0
set lj_sig11 [expr 2*$r_sNCP]
set lj_rcut11 [expr 1.122462*$lj_sig11]
set lj_cshift11 [calc_lj_shift $lj_sig11 $lj_rcut11]
set lj_roff11 [expr $r_hNCP+$r_hNCP]
## (6) Between type 1 & 2 ##
# Lennard-Jones
set lj_eps12 1.0
set lj_sig12 [expr $r_sNCP+$r_Na]
set lj_rcut12 [expr 1.122462*$lj_sig12]
set lj_cshift12 [calc_lj_shift $lj_sig12 $lj_rcut12]
set lj_roff12 [expr $r_hNCP+0.]
## (7) Between type 1 & 3 ##
# Lennard-Jones
set lj_eps13 1.0
set lj_sig13 [expr $r_sNCP+$r_CI]
set lj_rcut13 [expr 1.122462*$lj_sig13]
set lj_cshift13 [calc_lj_shift $lj_sig03 $lj_rcut13]
set lj_roff13 [expr $r_hNCP+0.]
## (8) Between type 2 & 2 ##
# Lennard-Jones
set lj_eps22 1.0
set lj_sig22 [expr 2*$r_Na]
set lj_rcut22 [expr 1.122462*$lj_sig22]
set lj_cshift22 [calc_lj_shift $lj_sig22 $lj_rcut22]
set lj_roff22 0.0
## (9) Between type 2 & 3 ##
# Lennard-Jones
set lj_eps23 1.0
set lj_sig23 [expr $r_Na+$r_CI]
set lj_rcut23 [expr 1.122462*$lj_sig23]
set lj_cshift23 [calc_lj_shift $lj_sig23 $lj_rcut23]
set lj_roff23 0.0
## (10) Between type 3 & 3 ##
# Lennard-Jones

```

```

set lj_eps33 1.0
set lj_sig33 [expr 2*$r_CI]
set lj_rcut33 [expr 1.122462*$lj_sig33]
set lj_cshift33 [calc_lj_shift $lj_sig33 $lj_rcut33]
set lj_roff33 0.0
#####INTERACTIONS#####
### (1) Between 0 0 ###
# LJ
inter 0 0 lennard-jones $lj_eps00 $lj_sig00 $lj_rcut00 $lj_cshift00 $lj_roff00
# Harmonic
inter 0 harmonic $harm_k $harm_r
# Bond angle
inter 1 angle $bend_k $bend_phiO
### (2) Between 0 1 ###
inter 0 1 lennard-jones $lj_eps01 $lj_sig01 $lj_rcut01 $lj_cshift01 $lj_roff01
### (3) Between 0 2 ###
inter 0 2 lennard-jones $lj_eps02 $lj_sig02 $lj_rcut02 $lj_cshift02 $lj_roff02
### (4) Between 0 3 ###
inter 0 3 lennard-jones $lj_eps03 $lj_sig03 $lj_rcut03 $lj_cshift03 $lj_roff03
### (5) Between 1 1 ###
inter 1 1 lennard-jones $lj_eps11 $lj_sig11 $lj_rcut11 $lj_cshift11 $lj_roff11
### (6) Between 1 2 ###
inter 1 2 lennard-jones $lj_eps12 $lj_sig12 $lj_rcut12 $lj_cshift12 $lj_roff12
### (7) Between 1 3 ###
inter 1 3 lennard-jones $lj_eps13 $lj_sig13 $lj_rcut13 $lj_cshift13 $lj_roff13
### (8) Between 2 2 ###
inter 2 2 lennard-jones $lj_eps22 $lj_sig22 $lj_rcut22 $lj_cshift22 $lj_roff22
### (9) Between 2 3 ###
inter 2 3 lennard-jones $lj_eps23 $lj_sig23 $lj_rcut23 $lj_cshift23 $lj_roff23
### (10) Between 3 3 ###
inter 3 3 lennard-jones $lj_eps33 $lj_sig33 $lj_rcut33 $lj_cshift33 $lj_roff33
# Below decide whether we use the last configuration or start a new run
if { [ file exists "dna14.end" ] } {
puts "This script is not for reading the last data"
exit }
#####CREATING PARTICLES#####
### DNA ###
puts "Generating $n_D DNA of $n_MD monomers with charge $q_MD per-monomer
(total particle=$n_part)"
polymer $n_D $n_MD $l_MD start $pid_MD mode PSAW [expr $lj_roff00*1.2] charge
$q_MD distance 1 types $type_MD $type_MD bond 0
#fill the DNA mass

```

```

for {set m $pid_MD} {$m < [expr $pid_MD+$nt_MD]} {incr m} {
part $m mass $m_MD
}
#fill the dna angle bond
for {set m $pid_MD} {$m < [expr $pid_MD+$nt_MD-2]} {incr m} {
set m1 [expr $m+1]
set m2 [expr $m+2]
part $m1 bond 1 $m $m2
}
#move DNA to COB
DNAToCENTER
puts "Creating NCP"
### NCP ###
counterions $n_NCP start $pid_NCP mode SAW $lj_roff01 charge $q_NCP type $type_NCP
#fill the NCP mass
for {set m $pid_NCP} {$m < [expr $pid_NCP+$n_NCP]} {incr m} {
part $m mass $m_NCP
}
puts "Creating SALT"
### SALT ###
salt $n_Na $n_Cl start $pid_Na mode SAW $lj_roff02 charge $q_Na $q_Cl types $type_Na
$type_Cl
#fill the salt mass
for {set m $pid_Na} {$m < [expr $pid_Na+$n_Na]} {incr m} {
part $m mass $m_Na
}
for {set m $pid_Cl} {$m < [expr $pid_Cl+$n_Cl]} {incr m} {
part $m mass $m_Cl
}
## create initial position coord. file
set f_xmol [open init$title.xmol "w"]
confxmol $f_xmol
close $f_xmol
##### WARMING UP #####
# the importance of warming up is not only for relaxing DNA polymer but also for over-
lapping molecule position inside DNA & NCP hard sphere radius
## Equilibrating use ljforcecap
set min01 0
set min02 0
set min03 0
set min11 0
set min12 0

```

```

set min13 0
inter 0 1 lennard-jones $lj_eps01 [expr $r_MD+$r_NCP] [expr 1.122462*[expr $r_MD+$r_NCP]]
[calc_lj_shift [expr $r_MD+$r_NCP] [expr 1.122462*[expr $r_MD+$r_NCP]]] 0.0
inter 0 2 lennard-jones $lj_eps02 [expr $r_MD+$r_Na] [expr 1.122462*[expr $r_MD+$r_Na]]
[calc_lj_shift [expr $r_MD+$r_Na] [expr 1.122462*[expr $r_MD+$r_Na]]] 0.0
inter 0 3 lennard-jones $lj_eps03 [expr $r_MD+$r_Cl] [expr 1.122462*[expr $r_MD+$r_Cl]]
[calc_lj_shift [expr $r_MD+$r_Cl] [expr 1.122462*[expr $r_MD+$r_Cl]]] 0.0
inter 1 1 lennard-jones $lj_eps11 [expr $r_NCP+$r_NCP] [expr 1.122462*[expr $r_NCP+$r_NCP]]
[calc_lj_shift [expr $r_NCP+$r_NCP] [expr 1.122462*[expr $r_NCP+$r_NCP]]] 0.0
inter 1 2 lennard-jones $lj_eps12 [expr $r_NCP+$r_Na] [expr 1.122462*[expr $r_NCP+$r_Na]]
[calc_lj_shift [expr $r_NCP+$r_Na] [expr 1.122462*[expr $r_NCP+$r_Na]]] 0.0
inter 1 3 lennard-jones $lj_eps13 [expr $r_NCP+$r_Cl] [expr 1.122462*[expr $r_NCP+$r_Cl]]
[calc_lj_shift [expr $r_NCP+$r_Cl] [expr 1.122462*[expr $r_NCP+$r_Cl]]] 0.0
set min01 [analyze mindist 0 1]
set min02 [analyze mindist 0 2]
set min03 [analyze mindist 0 3]
set min11 [analyze mindist 1 1]
set min12 [analyze mindist 1 2]
set min13 [analyze mindist 1 3]
puts "Minimum distance before warm up: min01= $min01; min02= $min02; min03=
$min03; min11= $min11; min12= $min12; min13= $min13"
set F_max 2000
while { $min01 < [expr ($r_MD+$r_NCP)*0.9] || $min02 < [expr ($r_MD+$r_Na)*0.9]
|| $min03 < [expr ($r_MD+$r_Cl)*0.9] || $min11 < [expr ($r_NCP+$r_NCP)*0.9] ||
$min12 < [expr ($r_NCP+$r_Na)*0.9] || $min13 < [expr ($r_NCP+$r_Cl)*0.9] } {
# setting ljforcecap
inter ljforcecap $F_max
#integrate a number of steps, e.g. 20
integrate 2000
#check the system status
set min01 [analyze mindist 0 1]
set min02 [analyze mindist 0 2]
set min03 [analyze mindist 0 3]
set min11 [analyze mindist 1 1]
set min12 [analyze mindist 1 2]
set min13 [analyze mindist 1 3]
puts "F_max in warming = $F_max, minimum distance: min01= $min01; min02= $min02;
min03= $min03; min11= $min11; min12= $min12; min13= $min13"
incr F_max 2000
# confxmol $f_xmol
flush stdout
}

```

```

#close $f_xmol
puts "Warm up finished. Minimal Distance now: min01= $min01; min02= $min02;
min03= $min03; min11= $min11; min12= $min12; min13= $min13"
# turn off the ljforcecap
inter ljforcecap 0
puts "return the truth LJ interaction"
inter 0 0 lennard-jones $lj_eps00 $lj_sig00 $lj_rcut00 $lj_cshift00 $lj_roff00
inter 0 1 lennard-jones $lj_eps01 $lj_sig01 $lj_rcut01 $lj_cshift01 $lj_roff01
inter 0 2 lennard-jones $lj_eps02 $lj_sig02 $lj_rcut02 $lj_cshift02 $lj_roff02
inter 0 3 lennard-jones $lj_eps03 $lj_sig03 $lj_rcut03 $lj_cshift03 $lj_roff03
inter 1 1 lennard-jones $lj_eps11 $lj_sig11 $lj_rcut11 $lj_cshift11 $lj_roff11
inter 1 2 lennard-jones $lj_eps12 $lj_sig12 $lj_rcut12 $lj_cshift12 $lj_roff12
inter 1 3 lennard-jones $lj_eps13 $lj_sig13 $lj_rcut13 $lj_cshift13 $lj_roff13
inter 2 2 lennard-jones $lj_eps22 $lj_sig22 $lj_rcut22 $lj_cshift22 $lj_roff22
inter 2 3 lennard-jones $lj_eps23 $lj_sig23 $lj_rcut23 $lj_cshift23 $lj_roff23
inter 3 3 lennard-jones $lj_eps33 $lj_sig33 $lj_rcut33 $lj_cshift33 $lj_roff33
# bonus integration
integrate 2000
##### TESTING #####
puts ""
puts "Below to test the truth of the system:"
puts ""
puts "box_length original = $box_ori A"
puts "salt concentration = $c_salt mM"
puts "harmonic constant and r_spring = $harm_k & $r_spring"
puts "angle constant and theta_0 = $bend_k & $bend_phiO"
puts "time step= $time_step"
puts "total monomer = $nt_MD"
puts "total NCP = $n_NCP"
puts "total Na = $n_Na"
puts "total Cl = $n_Cl"
puts "total particle = $n_part"
puts "addNa = $addNa"
puts "addCl = $addCl"
puts "DNA mass = $m_MD"
puts "NCP mass = $m_NCP"
puts "Na mass = $m_Na"
puts "Cl mass = $m_Cl"
#particle id
puts "the last particle id = [expr $pid_Cl+$n_Cl-1] must the same with [expr $n_part-1]"
# total charge of particles
puts "charge of particle: $q_totalMD+$q_totalNCP+$q_totalNa+$q_totalCl=[expr $q_totalMD+

```

```

$q_totalNCP+$q_totalNa+$q_totalCl]"
puts [inter]
##### Tuning P3M #####3##
puts "here I tune P3M with Bjerrum length = 1 and accuracy 1.e-02"
inter coulomb 1 p3m tune accuracy 1.e-02
puts "[inter coulomb 1 p3m tune accuracy 1.e-02]"
##### Set Up Observable Files #####
##### Temperature and Energy #####
set energyinp [open "energy.dat" "a+"]
puts $energyinp "Time Temperature Energy_Total Energy_Kinetic Energy_Potential"
##### INTEGRATION #####
puts "Integration"
## create trajectory coord. file
set f_xmol [open traj$title.xmol "a+"]
set t_half [clock second]
## simulation
set n_cycle 1000
set n_steps 10000
set i 0
set t_loop [clock second]
while { $i<$n_cycle } {
integrate $n_steps
##### TEMPERATURE AND ENERGY #####
set ek [analyze energy kinetic]
set etot [analyze energy total]
set ep [expr $etot-$ek]
set temperature [expr $ek/(([degrees_of_freedom]/2.0)*$n_part)]
#write in file
puts $energyinp "[setmd time] $temperature $etot $ek $ep"
#show in screen
puts -nonewline "Temp = $temperature Etot = $etot"; puts " Ek = $ek Ep = $ep"
flush stdout
### checkpoint each 200,000 step ###
if { [expr (($i+1)*$n_steps)%200000] == 0 } {
checkpoint_set $title.[expr ($i+1)*$n_steps].cpt
}
##### CONFIGURATION #####
#take configuration each 100.000 step
if { [expr (($i+1)*$n_steps)%100000] == 0 } {
confxmol $f_xmol
}
incr i

```

```

puts "$i"
flush stdout
}
close $f_xmol
##### AFTER INTEGRATION #####
##### Record the last configuration #####
set varinp [open "variable.dat" w]
blockfile $varinp write variable all
checkpoint_set end$title.cpt
set f_tinker [open end$title.txyz "w"]
conf_tinker $f_tinker
##### ENDING #####
close $f_tinker
close $energyinp
close $varinp
puts "FINISH"
set t2 [clock second]
set time [expr $t2-$t1]
set time_setup [expr $t_half-$t1]
puts "time_setup = $time_setup"
puts "total time = $time second"
##### END OF PROGRAM #####

```

B.2 Production Run

The TCL script which follows is used to sample properties of the system in equilibrium. This script contains the commands for sampling the energies, radial distribution function, end-to-end distance, radii of gyration, contour length, average bond length, average bending angle, particles trajectory, and activating the system checkpoint/recorder for the next run.

The script run as follows:

```

-----
set t1 [clock second]
puts "[code_info]"
##### PROCEDURE #####
# Add configuration in TXYZ (Tinker) format to the trajectory file "f_tinker"

```

```

# Below puts all particle including salts in trajectory file
proc conf_tinker { f_tinker } {
  global box_l n_MD n_part chem_typ atom_typ
  puts $f_tinker "$n_part after [setmd time] i.u., BOX: $box_l"
  for {set j 0} {$j < [expr $n_MD -1]} {incr j} {
    puts $f_tinker "[expr $j+1] $chem_typ($j) [part $j print folded_position] $atom_typ($j)
    [expr $j+2] "
  }
  for {set j [expr $n_MD-1]} {$j < $n_part} {incr j} {
    puts $f_tinker "[expr $j+1] $chem_typ($j) [part $j print pos] $atom_typ($j)"
  }}
# Add configuration in XMOL format to the trajectory file "f_xmol"
# Below only puts DNA Monomers and NCP in trajectory file
proc confxmol { f_xmol } {
  global box_l chem_typ n_MD n_NCP
  puts $f_xmol [expr int([expr $n_MD + $n_NCP])]
  puts $f_xmol " after [setmd time] i.u., BOX: $box_l $box_l $box_l"
  for {set j 0} {$j < [expr $n_MD + $n_NCP]} {incr j} {
    puts $f_xmol " $chem_typ($j) [part $j print pos] "
  }}
# procedure to get sqrt of a double/real number. Since tcl cannot sqrt a real/long format
proc NewSqrt { n } {
  return [expr {exp(log($n)/2)}]
}
puts " "
puts "=====
puts "= Taking the Last Polymer Data Run ="
puts "=====
puts " "
set rlast [open "dna14.end" r]
while { [blockfile $rlast read auto] != "eof" } {}
close $rlast
setmd time_step $time_step
# creating DNA, NCP, Na, Cl chem_typ for gopenmol needs
for {set i $pid_MD} {$i < [expr $pid_MD+$nt_MD]} {incr i} {set chem_typ($i) "O"
;set atom_typ($i) "40"}
for {set ii $pid_NCP} {$ii < [expr $pid_NCP+$n_NCP]} {incr ii} {set chem_typ($ii)
"C" ; set atom_typ($ii) "14"}
for {set j $pid_Na} {$j < [expr $pid_Na+$n_Na]} {incr j} {set chem_typ($j) "Na" ; set
atom_typ($j) "81"}
for {set jj $pid_Cl} {$jj < [expr $pid_Cl+$n_Cl]} {incr jj} {set chem_typ($jj) "Cl" ; set
atom_typ($jj) "93"}

```

```

## create initial position coord. file
set f_xmol [open init$title.xmol "w"]
confxmol $f_xmol
close $f_xmol
##### TESTING #####
puts ""
puts "Below for to test the truth of the system:"
puts ""
puts "box_length original = $box_ori A"
puts "salt concentration = $c_salt mM"
puts "harmonic constant= $harm_k"
puts "angle constant= $bend_k"
puts "time step= $time_step"
puts "total monomer = $nt_MD"
puts "total NCP = $n_NCP"
puts "total Na = $n_Na"
puts "total Cl = $n_Cl"
puts "total particle = $n_part"
puts "addNa = $addNa"
puts "addCl = $addCl"
#particle id
puts "the last particle id = [expr $pid_Cl+$n_Cl-1] must the same with [expr $n_part-1]"
# total charge of particles
puts "charge of particle: $q_totalMD+$q_totalNCP+$q_totalNa+$q_totalCl= [expr $q_totalMD+
$q_totalNCP+$q_totalNa+$q_totalCl]"
##### Set Up Observable Files #####
##### Energy #####
set energyinp [open "energy.dat" "w"]
puts $energyinp "Temperature Energy_Kinetic Ek_stddev Energy_Potential Ep_stddev
Energy_Total Etot_stddev Energy_Coulomb Ecol_stddev"
set dump_ek 0.
set dump_ep 0.
set dump_etot 0.
set dump_ecoulomb 0.
# later $avg_xx2 will be standard deviation sigma. But I only count std dev for energy, re
and rg, not for rdf. since rdf is a list and for simplicity
set avg_ek 0.
set avg_ek2 0.
set avg_ep 0.
set avg_ep2 0.
set avg_etot 0.
set avg_etot2 0.

```

```

set avg_ecoulomb 0.
set avg_ecoulomb2 0.
##### RDF #####
set rdffin [open "rdf.dat" "w"]
puts $rdffin "rlist rdf00 rdf01 rdf02 rdf03 rdf11 rdf12 rdf13 rdf22 rdf23 rdf33"
set rdf_bin 5000
set rdf_fac 1.
set rdf_min 0.
set rdf_count 0
#below is the initial setting of avg_rdfxx containing a list of number, not only a single
number
#later it will be an accumulation of each rdf value sampled, which then averaged after
total amount of rdf sampled = $dump
set dump_rdf00 ""
set dump_rdf01 ""
set dump_rdf02 ""
set dump_rdf03 ""
set dump_rdf11 ""
set dump_rdf12 ""
set dump_rdf13 ""
set dump_rdf22 ""
set dump_rdf23 ""
set dump_rdf33 ""
for {set ii 0} {$ii < $rdf_bin} {incr ii} {
set dump_rdf00 [concat $dump_rdf00 0]
set dump_rdf01 [concat $dump_rdf01 0]
set dump_rdf02 [concat $dump_rdf02 0]
set dump_rdf03 [concat $dump_rdf03 0]
set dump_rdf11 [concat $dump_rdf11 0]
set dump_rdf12 [concat $dump_rdf12 0]
set dump_rdf13 [concat $dump_rdf13 0]
set dump_rdf22 [concat $dump_rdf22 0]
set dump_rdf23 [concat $dump_rdf23 0]
set dump_rdf33 [concat $dump_rdf33 0]
}
#below is the initial setting of avg_rdfxx containing a list of number, not only a single
number
#later it will be an accumulation of averaged rdf value from each dump, which then aver-
aged after total amount of dumping = $total_dump
set avg_rdf00 ""
set avg_rdf01 ""
set avg_rdf02 ""

```

```

set avg_rdf03 ""
set avg_rdf11 ""
set avg_rdf12 ""
set avg_rdf13 ""
set avg_rdf22 ""
set avg_rdf23 ""
set avg_rdf33 ""
for {set ii 0} {$ii < $rdf_bin} {incr ii} {
set avg_rdf00 [concat $avg_rdf00 0]
set avg_rdf01 [concat $avg_rdf01 0]
set avg_rdf02 [concat $avg_rdf02 0]
set avg_rdf03 [concat $avg_rdf03 0]
set avg_rdf11 [concat $avg_rdf11 0]
set avg_rdf12 [concat $avg_rdf12 0]
set avg_rdf13 [concat $avg_rdf13 0]
set avg_rdf22 [concat $avg_rdf22 0]
set avg_rdf23 [concat $avg_rdf23 0]
set avg_rdf33 [concat $avg_rdf33 0]
}
#below is only for taking rlist, rdf02 is only a means for recording rlist. It lists the distance
containing the particle density. clearly will be x variable on RDF plot
set rlist ""
set rdf02 [analyze rdf 0 2 $rdf_min [expr $box_1/$rdf_fac] $rdf_bin]
foreach value [lindex $rdf02 1] {
lappend rlist [expr [lindex $value 0]*$l_unit]
}
##### TOPOLOGY #####
set topoinp [open "topology.dat" w]
puts $topoinp "re redev re2 rg rgdev rg2"
## setting below have the same idea with rdf above. but below only contain a value, not a
list of value like rdf
set dump_re 0.
set dump_re2 0.
set dump_rg 0.
set dump_rg2 0.
set avg_re 0.
set avg_re2 0.
set avg_redev 0.
set avg_rg 0.
set avg_rg2 0.
set avg_rgdev 0.
##### CONTOUR LENGTH #####

```

```

set contourinp [open "contour.dat" w]
puts $contourinp "contour_length stddev_cont_length average_bond_length average_bond_angle
dev_bond_angle"
set tmp_contour 0.
set avg_contour 0.
set dev_contour 0.
set tmp_angle 0.
set avg_angle 0.
set dev_angle 0.
set n_contour 0
### INTEGRATION ###
puts "Integration"
## create trajectory coord. file
set f_xmol [open traj$title.xmol "a+"]
set t_half [clock second]
set n_cycle 200
set n_steps 10000
set dump 10
set total_dump [expr $n_cycle/$dump]
set i 0
set t_loop [clock second]
while { $i<$n_cycle } {
integrate $n_steps
##### ENERGY #####3#####
#set energylist [analyze energy]
set etot [analyze energy total]
set ek [analyze energy kinetic]
set ep [expr $etot-$ek]
set ecoulomb [analyze energy coulomb]
set temperature [expr $ek/(([degrees_of_freedom]/2.0)*$n_part)]
set dump_ek [expr $dump_ek+$ek]
set dump_ep [expr $dump_ep+$ep]
set dump_etot [expr $dump_etot+$etot]
set dump_ecoulomb [expr $dump_ecoulomb+$ecoulomb]
#show in screen
#puts "E = $energy list"
puts -nonewline "Temp = $temperature Etot = $etot"; puts " Ek = $ek Ep = $ep"
##### rdf #####
#it must zero for putting a RENEWED list, that will be added to avg_rdf
set rdfflist00 ""
set rdfflist01 ""
set rdfflist02 ""

```

```

set rdflist03 ""
set rdflist11 ""
set rdflist12 ""
set rdflist13 ""
set rdflist22 ""
set rdflist23 ""
set rdflist33 ""

set rdf00 [analyze rdf 0 0 $rdf_min [expr $box_1/$rdf_fac] $rdf_bin]
set rdf01 [analyze rdf 0 1 $rdf_min [expr $box_1/$rdf_fac] $rdf_bin]
set rdf02 [analyze rdf 0 2 $rdf_min [expr $box_1/$rdf_fac] $rdf_bin]
set rdf03 [analyze rdf 0 3 $rdf_min [expr $box_1/$rdf_fac] $rdf_bin]
set rdf11 [analyze rdf 1 1 $rdf_min [expr $box_1/$rdf_fac] $rdf_bin]
set rdf12 [analyze rdf 1 2 $rdf_min [expr $box_1/$rdf_fac] $rdf_bin]
set rdf13 [analyze rdf 1 3 $rdf_min [expr $box_1/$rdf_fac] $rdf_bin]
set rdf22 [analyze rdf 2 2 $rdf_min [expr $box_1/$rdf_fac] $rdf_bin]
set rdf23 [analyze rdf 2 3 $rdf_min [expr $box_1/$rdf_fac] $rdf_bin]
set rdf33 [analyze rdf 3 3 $rdf_min [expr $box_1/$rdf_fac] $rdf_bin]
foreach value00 [lindex $rdf00 1] value01 [lindex $rdf01 1] value02 [lindex $rdf02 1]
value03 [lindex $rdf03 1] value11 [lindex $rdf11 1] value12 [lindex $rdf12 1] value13
[lindex $rdf13 1] value22 [lindex $rdf22 1] value23 [lindex $rdf23 1] value33 [lindex
$rdf33 1] {
set rdflist00 [concat $rdflist00 [lindex $value00 1]]
set rdflist01 [concat $rdflist01 [lindex $value01 1]]
set rdflist02 [concat $rdflist02 [lindex $value02 1]]
set rdflist03 [concat $rdflist03 [lindex $value03 1]]
set rdflist11 [concat $rdflist11 [lindex $value11 1]]
set rdflist12 [concat $rdflist12 [lindex $value12 1]]
set rdflist13 [concat $rdflist13 [lindex $value13 1]]
set rdflist22 [concat $rdflist22 [lindex $value22 1]]
set rdflist23 [concat $rdflist23 [lindex $value23 1]]
set rdflist33 [concat $rdflist33 [lindex $value33 1]]
}
set dump_rdf00 [vecadd $dump_rdf00 $rdflist00]
set dump_rdf01 [vecadd $dump_rdf01 $rdflist01]
set dump_rdf02 [vecadd $dump_rdf02 $rdflist02]
set dump_rdf03 [vecadd $dump_rdf03 $rdflist03]
set dump_rdf11 [vecadd $dump_rdf11 $rdflist11]
set dump_rdf12 [vecadd $dump_rdf12 $rdflist12]
set dump_rdf13 [vecadd $dump_rdf13 $rdflist13]
set dump_rdf22 [vecadd $dump_rdf22 $rdflist22]
set dump_rdf23 [vecadd $dump_rdf23 $rdflist23]
set dump_rdf33 [vecadd $dump_rdf33 $rdflist33]

```

```

##### TOPOLOGY #####
analyze set chains 0 1 $n_MD
set dump_re [expr $dump_re+[lindex [analyze re] 0]]
set dump_re2 [expr $dump_re2+[lindex [analyze re] 2]]
set dump_rg [expr $dump_rg+[lindex [analyze rg] 0]]
set dump_rg2 [expr $dump_rg2+[lindex [analyze rg] 2]]
#####
### DUMPING ###
#####
if { [expr ($i+1)%$dump]==0 } {
### ENERGY
# below we average values of one dump then accumulate to avg_exx
set avg_ek [expr $avg_ek+[expr $dump_ek/$dump]]
set avg_ek2 [expr $avg_ek2+[expr pow(($dump_ek/$dump),2)]]
set avg_ep [expr $avg_ep+[expr $dump_ep/$dump]]
set avg_ep2 [expr $avg_ep2+[expr pow(($dump_ep/$dump),2)]]
set avg_etot [expr $avg_etot+[expr $dump_etot/$dump]]
set avg_etot2 [expr $avg_etot2+[expr pow(($dump_etot/$dump),2)]]
set avg_ecoulomb [expr $avg_ecoulomb+[expr $dump_ecoulomb/$dump]]
set avg_ecoulomb2 [expr $avg_ecoulomb2+[expr pow(($dump_ecoulomb/$dump),2)]]
#after this we must make the value of $dump_exx become zero again, since we will start
a new dump
set dump_ek 0.
set dump_ep 0.
set dump_etot 0.
set dump_ecoulomb 0.
### RDF
#here we dumping $dump_rdfxxx, containing each rdf data sampled, after each $dump
step, then average over in each dump
set avg_rdf00 [vecadd $avg_rdf00 [vecscale [expr 1.0/$dump] $dump_rdf00]]
set avg_rdf01 [vecadd $avg_rdf01 [vecscale [expr 1.0/$dump] $dump_rdf01]]
set avg_rdf02 [vecadd $avg_rdf02 [vecscale [expr 1.0/$dump] $dump_rdf02]]
set avg_rdf03 [vecadd $avg_rdf03 [vecscale [expr 1.0/$dump] $dump_rdf03]]
set avg_rdf11 [vecadd $avg_rdf11 [vecscale [expr 1.0/$dump] $dump_rdf11]]
set avg_rdf12 [vecadd $avg_rdf12 [vecscale [expr 1.0/$dump] $dump_rdf12]]
set avg_rdf13 [vecadd $avg_rdf13 [vecscale [expr 1.0/$dump] $dump_rdf13]]
set avg_rdf22 [vecadd $avg_rdf22 [vecscale [expr 1.0/$dump] $dump_rdf22]]
set avg_rdf23 [vecadd $avg_rdf23 [vecscale [expr 1.0/$dump] $dump_rdf23]]
set avg_rdf33 [vecadd $avg_rdf33 [vecscale [expr 1.0/$dump] $dump_rdf33]]
set dump_rdf00 ""
set dump_rdf01 ""
set dump_rdf02 ""

```

```

set dump_rdf03 ""
set dump_rdf11 ""
set dump_rdf12 ""
set dump_rdf13 ""
set dump_rdf22 ""
set dump_rdf23 ""
set dump_rdf33 ""
for {set ii 0} {$ii < $rdf_bin} {incr ii} {
set dump_rdf00 [concat $dump_rdf00 0]
set dump_rdf01 [concat $dump_rdf01 0]
set dump_rdf02 [concat $dump_rdf02 0]
set dump_rdf03 [concat $dump_rdf03 0]
set dump_rdf11 [concat $dump_rdf11 0]
set dump_rdf12 [concat $dump_rdf12 0]
set dump_rdf13 [concat $dump_rdf13 0]
set dump_rdf22 [concat $dump_rdf22 0]
set dump_rdf23 [concat $dump_rdf23 0]
set dump_rdf33 [concat $dump_rdf33 0]
}
### TOPOLOGY
set avg_re [expr $avg_re+[expr $dump_re/$dump]]
set avg_re2 [expr $avg_re2+[expr $dump_re2/$dump]]
set avg_redev [expr $avg_redev+[expr pow(($dump_re/$dump),2)]]
set avg_rg [expr $avg_rg+[expr $dump_rg/$dump]]
set avg_rg2 [expr $avg_rg2+[expr $dump_rg2/$dump]]
set avg_rgdev [expr $avg_rgdev+[expr pow(($dump_rg/$dump),2)]]
#return the $dump_rx value to zero
set dump_re 0.
set dump_re2 0.
set dump_rg 0.
set dump_rg2 0.
flush stdout
}
#### DUMPING FINISHED #####
### checkpoint each 200,000 step ###
if { [expr (($i+1)*$n_steps)%200000] == 0 } {
checkpoint_set $title.[expr ($i+1)*$n_steps].cpt
}
### trajectory and take contour length data each 100000 step ###
if { [expr (($i+1)*$n_steps)%100000] == 0 } {
confxmol $f_xmol
### contour_length data #####

```

```

for {set k $pid_MD} {$k < [expr $pid_MD+$n_MD-1]} {incr k} {
set tmp_contour [expr $tmp_contour+ [bond_length $k [expr $k+1]]]
if {$k > $pid_MD} {
set tmp_angle [expr $tmp_angle+ [bond_angle [expr $k-1] $k [expr $k+1] d]]
}}
set avg_contour [expr $avg_contour + $tmp_contour]
set dev_contour [expr $dev_contour+ pow($tmp_contour,2)]
set tmp_contour 0.
set avg_angle [expr $avg_angle+ ($tmp_angle/($n_MD-2))]
set dev_angle [expr $dev_angle+pow(($tmp_angle/($n_MD-2)),2)]
set tmp_angle 0.
incr n_contour
}
flush stdout
incr i
puts "$i"
}
close $f_xmol
### AFTER INTEGRATION ###
##### energy data #####3
set avg_ek [expr $avg_ek/$total_dump]
set avg_ek2 [NewSqrt [expr ($avg_ek2/$total_dump)-pow($avg_ek,2)]]
set avg_ep [expr $avg_ep/$total_dump]
set avg_ep2 [NewSqrt [expr ($avg_ep2/$total_dump)-pow($avg_ep,2)]]
set avg_etot [expr $avg_etot/$total_dump]
set avg_etot2 [NewSqrt [expr ($avg_etot2/$total_dump)-pow($avg_etot,2)]]
set avg_ecoulomb [expr $avg_ecoulomb/$total_dump]
set avg_ecoulomb2 [NewSqrt [expr ($avg_ecoulomb2/$total_dump)-pow($avg_ecoulomb,2)]]
puts $energyinp "$temperature $avg_ek $avg_ek2 $avg_ep $avg_ep2 $avg_etot $avg_etot2
$avg_ecoulomb $avg_ecoulomb2"
##### rdf data #####
set avg_rdf00 [vecscale [expr 1.0/$total_dump] $avg_rdf00]
set avg_rdf01 [vecscale [expr 1.0/$total_dump] $avg_rdf01]
set avg_rdf02 [vecscale [expr 1.0/$total_dump] $avg_rdf02]
set avg_rdf03 [vecscale [expr 1.0/$total_dump] $avg_rdf03]
set avg_rdf11 [vecscale [expr 1.0/$total_dump] $avg_rdf11]
set avg_rdf12 [vecscale [expr 1.0/$total_dump] $avg_rdf12]
set avg_rdf13 [vecscale [expr 1.0/$total_dump] $avg_rdf13]
set avg_rdf22 [vecscale [expr 1.0/$total_dump] $avg_rdf22]
set avg_rdf23 [vecscale [expr 1.0/$total_dump] $avg_rdf23]
set avg_rdf33 [vecscale [expr 1.0/$total_dump] $avg_rdf33]
foreach x_rdf $rlist y_rdf00 $avg_rdf00 y_rdf01 $avg_rdf01 y_rdf02 $avg_rdf02 y_rdf03

```

```

$avg_rdf03 y_rdf11 $avg_rdf11 y_rdf12 $avg_rdf12 y_rdf13 $avg_rdf13 y_rdf22 $avg_rdf22
y_rdf23 $avg_rdf23 y_rdf33 $avg_rdf33 {
puts $rdfinp "$x_rdf $y_rdf00 $y_rdf01 $y_rdf02 $y_rdf03 $y_rdf11 $y_rdf12 $y_rdf13
$y_rdf22 $y_rdf23 $y_rdf33" }
##### topology data #####
set avg_re [expr $avg_re/$total_dump]
set avg_re2 [expr $avg_re2/$total_dump]
set avg_redev [NewSqrt [expr ($avg_redev/$total_dump)-pow($avg_re,2)]]
set avg_rg [expr $avg_rg/$total_dump]
set avg_rg2 [expr $avg_rg2/$total_dump]
set avg_rgdev [NewSqrt [expr ($avg_rgdev/$total_dump)-pow($avg_rg,2)]]
# puts in real unit
puts $topoinp "[expr $avg_re*$l_unit] [expr $avg_redev*$l_unit] [expr $avg_re2*(pow($l_unit,2))]
[expr $avg_rg*$l_unit] [expr $avg_rgdev*$l_unit] [expr $avg_rg2*(pow($l_unit,2))]"
##### Record the last configuration #####
set varinp [open "variable.dat" w]
blockfile $varinp write variable all
checkpoint_set end$title.cpt
set f_tinker [open end$title.txyz "w"]
conf_tinker $f_tinker
##### CONTOUR LENGTH DATA #####
set avg_contour [expr $avg_contour/$n_contour]
set dev_contour [NewSqrt [expr ($dev_contour/$n_contour)-pow($avg_contour,2)]]
set avg_angle [expr $avg_angle/$n_contour]
set dev_angle [NewSqrt [expr ($dev_angle/$n_contour)-pow($avg_angle,2)]]
# puts in file in real unit (angstrom)
puts $contourinp "[expr $avg_contour*$l_unit] [expr $dev_contour*$l_unit]
[expr ($avg_contour*$l_unit)/[expr $pid_MD+$n_MD-1]] $avg_angle $dev_angle"
##### ENDING #####
close $f_tinker
close $energyinp
close $topoinp
close $rdfinp
close $varinp
close $contourinp
puts "FINISH"
set t2 [clock second]
set time [expr $t2-$t1]
set time_setup [expr $t_half-$t1]
puts "time_setup = $time_setup"
puts "total time = $time second"
##### END OF PROGRAM #####

```