On Using Automated Algorithms to Parameterise Molecules for Molecular Dynamics Simulations and Investigating Suitable Ensembles for the Simulation of Naphthalimide Monolayers

Ivan Welsh

MASSEY UNIVERSITY
Te Kunenga Ki Purehuroa

Institute of Natural and Mathematical Sciences

A thesis
submitted to Massey University in Albany, Auckland
in fulfilment of the requirements for the degree of
Doctor of Philosophy
in Chemistry.

Massey University Auckland
2017

Abstract

Molecular dynamics simulations provide a means to investigate the spatial and temporal evolution of systems of molecules at atomic resolution. Force fields are used to describe the interactions between atoms contained within the system. A number of such force fields have been developed over the years, with a focus on force fields for use in simulations of biochemical systems, in particular, protein systems. This thesis is primarily focused on extending the range of systems that can be simulated through providing means for automated generation of force field parameters for large novel molecules.

One component of existing force fields that is generally poorly parameterised are the dihedral terms. In combination with the non-bonded terms, the dihedral terms are used to describe the rotational energy profile about bonds, and have a large influence on the conformational properties of a simulated system. A new method for the determination of dihedral parameters is developed, utilising high level quantum mechanical calculations. With the use of local elevation molecular dynamics simulations, this method is applied to the case of protein backbone dihedrals within the GROMOS force field.

When one desires to simulate the interaction of a novel molecule with some biochemical system, the novel molecule must be parameterised in a manner that is compatible with the force field used to describe the biochemical system. However, doing so is a slow, tedious, and error prone process, especially when the novel molecule is large. To combat this, a new algorithm, known as CherryPicker, was developed. CherryPicker is a graph based algorithm which enables rapid parameterisation of large molecules through fragment comparison with a library of previously parameterised small molecules. The algorithm design is discussed and tested on a few simple test cases in part II.

Part III steps away from the parameterisation focus of this thesis and looks at the simulation of naphthalimide monolayers. Naphthalimides have applications in sensing environments as they have absorption and fluorescence emission spectra lying within the UV and visible regions of light. With a long chain alkane substituted at the N-imide site, they become amphiphilic and can form monolayers on the surface of water, and can be transferred to a solid substrate when at a desired compression level. Molecular dynamics simulations can be used to provide insight into the formation of compressed monolayer phase. Here, the effect of different ensembles, namely NVT, NPT, and NγT are investigated for use in simulating a naphthalimide monolayer.

Acknowledgements

Lets keep this short. First person to thank is Dr Jane Allison, for hiring me as a PhD Student, and paying me, and basically letting me choose what work to do, even though it involved lots of things outside of her area of expertise. Also thanks to Professor Alan Mark and his group at the University of Queensland, particularly Martin and Bertrand, for hosting me a couple of times. To Dr yomcat (little y), thanks for dealing with my stoopid maths questions as they arose. And finally, thanks to the volunteer proofreaders who gave up their time to read my brain dumps. So cheers to Dr Thomas Collier, Bruce Welsh, and Nina Leeb. Nina, I look forward to reciprocating when its your time. Oh and big thanks to my lovely mother who decided to send me fantastic lunches three days a week. Because physics. Finally, because I forgot them in my examination copy, thanks to my group members for all the useful stuff anf things over the years, particularly Ashar.

Preface

With a preceding introductory chapter, this thesis is divided into three distinct parts. Part I focuses on the SpinningTop program, which is a program developed for determining dihedral parameters. A brief background to the reasons that such a method is required is given in chapter 2. Chapter 3 outlines the theory and implementation of the fitting method, and investigates some of the considerations that need to be made. Chapter 4 details the methods used to translate the developed fitting method to the case of protein backbone dihedral terms within the GROMOS force field, and chapter 5 discusses the results obtained in this proof of principle work.

Part II of the thesis focusses on the CherryPicker algorithm, which is a new algorithm developed to enable easy parameterisation of large biochemical molecules compatible with the GROMOS force field. As the algorithm is based on the concept of molecular fragmentation, a brief introduction to the state of computational molecular fragmentation is given in chapter 6. The design and mathematical background of the algorithm is presented in chapter 7, before a small amount of proof-of-concept testing is undertaken in chapter 8. As part of the CherryPicker algorithm development, a novel means to automatically determine bond order and formal charges of molecules was developed. This is presented in chapter 9.

Finally, part III presents work undertaken in the determination of suitable ensembles for the simulation of naphthalimide monolayers.

A large amount of code was developed as part of the work for this thesis. This code is available on request to j.allison@massey.ac.nz. The code will be provided as is, with no documentation on system requirements, installation, or usage.

Contents

List of Figures

List of Tables

List of Abbreviations

AA all atom.

ATB Automated Topology Builder.

BSSE basis set superposition error.

CIP Cahn–Ingold–Prelog priority rules.

CSD Cambridge Structural Database.

DAG directed acyclic graph.

FPT fixed parameter tractable.

GAFF Generalised AMBER force field.

NBO Natural Bond Orbital.

PDB Protein Data Bank.

QM quantum mechanical.

QMDFF Quantum Mechanically Derived Force Field.

RMSD root-mean-square deviation.

UA united atom.

UFF Universal Force Field.

1. Introduction

Molecular dynamics simulations provide a means to investigate the spatial evolution as a function of time of atoms and molecules within chemical or biological systems. Newton's equations of motion are numerically integrated in discrete time steps for a set of interacting particles. Descriptions of how atoms should interact with one another are provided by a force field. Such force fields include terms for confining atoms within molecules, e.g. describing the bonds between atoms, as well as terms describing longer range effects, such as electrostatic and dispersion interactions.

Like the majority of technical scientific methods, molecular dynamics was first developed as a theoretical physics model for solid state systems in the late 1950's.[1] Since then, the use of molecular dynamics has expanded into other fields, in particular the modelling of biomolecules and biological systems. There are many different force fields available for use in biomolecular simulations. These force fields are fundamentally similar to one another, though differing parametrisation philosophies, techniques, and choices of reference data have lead to noticeable differences between them.

1.1. Molecular Dynamics

Molecular dynamics simulations are very similar to tangible experimentation. A sample is prepared and connected to a measurement device. The measurement device is used to measure the property of interest over a period of time. As the length of the time period increases, the results become more accurate as statistical noise is averaged out. A molecular dynamics simulation follows a similar path. First a system of particles is initialised and equilibrated by solving Newton's equations of motion until the properties of the system no longer change over time. After equilibration, the simulation is continued for a period of time so the measurement can be taken.

1.1.1. Integration

As mentioned, molecular dynamics simulations solve Newton's equations of motion for a system of N interacting particles:

$$\mathbf{F}_i = m_i \frac{\partial^2 \mathbf{r}_i}{\partial t^2};\ i = 1 \ldots N \tag{1.1}$$

where the forces $\mathbf{F}$ on a particle i are determined from the negative derivative of some potential energy function (see section 1.2). The affect of these forces on the atomic positions and velocities are determined through the use of numerical integration. A common form of numerical integrator used in molecular dynamics simulations is *leapfrog* integration.[2] Leapfrog integration updates the positions $\mathbf{r}$ and velocities $\mathbf{v}$ at interleaved time points, in such a way that they leap over one another much like frogs leaping. The process works as follows. Given a time step Δt, positions $\mathbf{r}_i$ and velocities $\mathbf{v}_{i+\frac{1}{2}}$, the positions at $i+1$ are given by

$$\mathbf{r}_{i+1} = \mathbf{r}_i + \mathbf{v}_{i+\frac{1}{2}} \Delta t \tag{1.2}$$

and the velocities are given by

$$\mathbf{v}_{i+\frac{1}{2}} = \mathbf{v}_{i-\frac{1}{2}} + \mathbf{F}(\mathbf{r}_i)\, \Delta t. \tag{1.3}$$

1.1.2. Initialisation

System initialisation for molecular dynamics consists of three parts; a set of parameters to control the simulation, the definition of the force field to be used, and a set of initial atomic positions for the system. Initial atomic positions are defined as (x, y, z) coordinates in Cartesian space. The boundaries of the set of coordinates define the simulation box. Initial velocities for the atoms are generated by randomly choosing from a Maxwell distribution

$$P(\mathbf{v}_j) = \left(\frac{2\pi k_B T_i}{m_j} \right)^{\frac{3}{2}} \exp\left(\frac{-m_j \mathbf{v}_j^2}{2 k_B T_i} \right) \tag{1.4}$$

where k_B is Boltzmann's constant and $P(\mathbf{v}_j)$ is the probability of atom j with mass m_j having velocity $\mathbf{v}_j$ given an initial temperature T_i. The ensemble of randomly generated velocities also need to be rescaled to correct for centre of mass motion of the entire system, and to give the true desired temperature as random generation does not guarantee obtaining the required temperature. As part of the initialisation, the choice of boundary conditions plays an important role in the simulation. A *boundary condition* is any restriction applied globally during the course

of the simulation. A restriction may be either *hard*, applied at every time step during the simulation, or *soft*, a restriction applied to the average of some value over a longer period of time. There are three types of boundary conditions, *spatial* boundary conditions, *geometric* boundary conditions, and *thermodynamic* boundary conditions. Each is discussed below.

1.1.2.1. Spatial Boundary Conditions

Spatial boundary conditions concern the nature of the confinement or periodicity of the system. Generally they fall into three categories: *vacuum* boundary conditions, *fixed* boundary conditions, and *periodic* boundary conditions. Vacuum boundary conditions apply when simulating a molecular system *in vacuo*, which corresponds to a gas phase simulation at zero pressure. This type of simulation can be desirable as the vast majority of computational time in a normal molecular dynamics simulation is spent on calculating solvent–solvent interactions. In vacuum there is no solvent present so greater time scales can be reached. There are, however, drawbacks. Properties of atoms at or near the surface will be distorted, and the shape of a non-spherical molecule may also be affected.[3,4] Additionally, the shielding effect of a solvent is missed *in vacuo*. Fixed boundary conditions confine the system to a region of space through the use of walls. Such boundary conditions also suffer from the edge effects seen in *in vacuo* simulations, though they can be minimised, by having an outer shell of particles which are restrained in position and so do not undergo full molecular dynamics. The most common spatial boundary conditions are periodic boundary conditions. Here the system is placed into a periodic space filling box and surrounded by identical images of itself as shown in figure 1.1. In effect, this allows the finite system to mimic an infinite system.

1.1.2.2. Geometric Boundary Conditions

Geometric boundary conditions concern the use of constraints or restraints on specific internal coordinates of the system. A constraint is used to keep a specific internal coordinate fixed at some defined value throughout the simulation and is regarded as a hard geometric boundary condition. A restraint however allows some motion about a reference point through the application of a harmonic restraining force. The primary use of constraints is to constrain bond lengths within a molecule to a fixed value. Bonds generally undergo symmetric, high frequency-low amplitude vibrations about an equilibrium point. To model these vibrations accurately in a numerically stable manner, short time steps are required. By constraining the bonds to their equilibrium lengths, these vibrations are removed and a much larger time step, Δt which is the time between numerical integration points, may be utilised. Algorithms for applying constrains include SHAKE[5] and LINCS.[6,7]

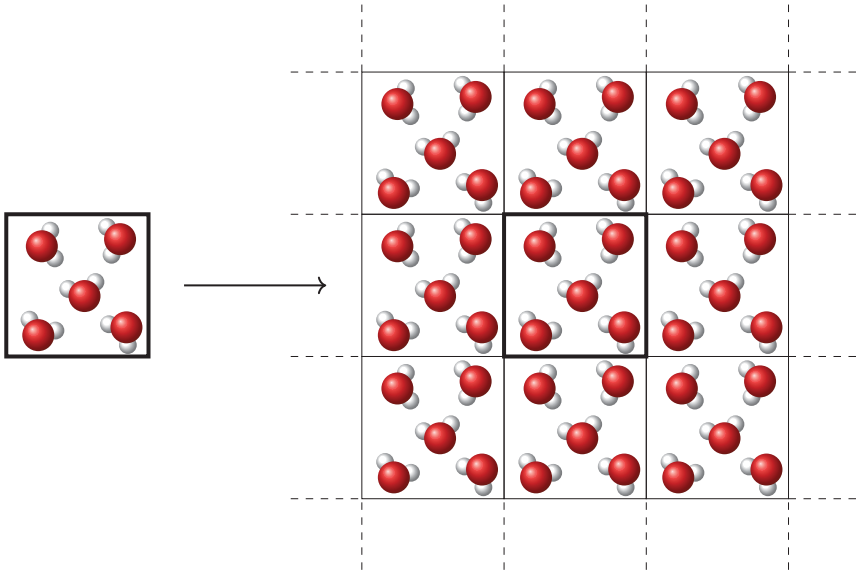


Figure 1.1: Periodic boundary conditions as applied to a box of water. The image on the left shows a box of water with fixed boundary conditions. On the right, the same box is shown surrounded by identical images of itself, giving rise to periodic boundary conditions.

1.1.2.3. Thermodynamic Boundary Conditions

Thermodynamic boundary conditions concern the thermodynamic state point associated with the collection of atomic configurations generated by simulation of the system, i.e. its equilibrium state. In molecular dynamics, the most common examples are a constant number of particles versus a constant chemical potential, a constant energy versus a constant temperature, and a constant volume versus a constant pressure. A molecular dynamics simulation generally contains a constant number of particles, so the constant chemical potential case will not be considered here. With a set of initial atomic positions and velocities, it becomes possible to integrate Newton's equations of motion. Doing so will result in atomic configurations taken from the NVE ensemble, that is the thermodynamic state where the number of particles, volume and energy are kept constant. In a number of cases it is useful to be able to keep temperature and/or pressure approximately constant, particularly as doing so emulates experimental conditions more closely.

Thermostats The instantaneous temperature $\mathcal{T}$ of a system at time point t depends on the kinetic energy present within the system at that time:

$$\mathcal{T}(t) = \sum_{i=1}^{N} \frac{m_i v_i^2(t)}{k_B N_f} \tag{1.5}$$

where m_i and $v_i(t)$ are the mass and velocity of particle i respectively, N is the total number of particles in the system, k_B is Boltzmann's constant, and N_f is the number of degrees of freedom $N_f = 3N - N_c - N_r$ with N_c coming from the constraints applied by the geometric boundary conditions and N_r depending on the spatial boundary conditions and any other source of degrees of freedom removal such as removal of centre of mass translation. A number of means to keep temperature constant, known as *thermostats*, are available. The natural way of modifying the temperature is by changing the velocities. The Woodcock thermostat enforces a strict $\mathcal{T}(t) = T_0$ condition, i.e there are no fluctuations allowed in the system, and scales the velocities accordingly.[8] The Berendsen thermostat mimics weak coupling with first order kinetics to an external heat bath with a given temperature T_0.[9] The deviation of the system temperature from T_0 is slowly corrected according to

$$\frac{d\mathcal{T}}{dt} = \frac{T_0 - \mathcal{T}}{\tau} \tag{1.6}$$

which means that a temperature deviation decays exponentially with a time constant τ. When $\tau = \Delta t$, the Berendsen thermostat reduces to the Woodcock thermostat. Additional thermostats include the velocity rescaling thermostat,[10] the Nose-Hoover thermostat,[11,12] and the Anderson thermostat.[13]

Barostats Barostats are similar to thermostats except that they control pressure rather than temperature. The instantaneous pressure $\mathcal{P}$ at time point t can be determined as

$$\mathcal{P}(t) = \frac{1}{V}\left(\frac{1}{3}\sum_{i=1}^{N} m_i \mathbf{v}_i^2 + \mathbf{r}_i \mathbf{f}_\mathbf{i}\right) \tag{1.7}$$

where V is the volume of the simulation box, N is the total number of particles in the system, m_i, $\mathbf{r}_i$ and $\mathbf{v}_i$ are the mass, position and velocity of atom i respectively and $\mathbf{f}_i$ is the force acting on it. The force is calculated as part of the integration of Newton's equations of motion (see section 1.1.1). This dependence on position means that the pressure can be affected by scaling the box size and consequently the atomic positions. For a given pressure change ΔP, the required change in volume ΔV is related by

$$\Delta\mathcal{P}(t) = \frac{-\Delta V(t)}{\kappa_T V(t)} \tag{1.8}$$

where κ_T is the isothermal compressibility of the system. Again, there are a number of different barostats, such as the Berendsen barostat which is similar to the Berendsen thermostat (equation 1.6) except that it scales pressure rather than temperature,[9] and the Parrinello-Rahman

barostat.[14,15]

1.2. Force Fields

As used in molecular dynamics, the term "force field" is a misnomer. Force fields are empirically parameterised sets of equations that describe the energy of a system of atoms, dependent on the atomic coordinates, $\mathcal{V}(\mathbf{r})$. There are no force terms in these equations. Rather, forces are determined from the negative gradient of this energy function. Nevertheless, the terminology is widespread and understood.

The majority of the widely used biochemical force fields, such as CHARMM,[16–20] AMBER,[21–24] and OPLS[25–32] follow the same, simple functional form; combining a summation of the bonded and non-bonded energy terms, as shown below

$$\begin{aligned}
\mathcal{V}(\mathbf{r}) &= E_{\text{bonded}} + E_{\text{nonbonded}} \\
E_{\text{bonded}} &= \sum_{\text{bonds}} \mathcal{V}^{b}(b;k_b,b_0) + \sum_{\text{angles}} \mathcal{V}^{\theta}(\theta;k_\theta,\theta_0) + \sum_{\text{dihedrals}} \mathcal{V}^{\phi}(\phi;k_\phi,m,\delta) \\
E_{\text{nonbonded}} &= \mathcal{V}^{\text{elec}}(\mathbf{r};q) + \sum_{i}\sum_{j} \mathcal{V}^{\text{vdw}}(\mathbf{r}_{ij};\varepsilon_{ij},\sigma_{ij})
\end{aligned} \tag{1.9}$$

Bonded terms are in turn a summation of the 1–2 interactions (bonds, figure 1.2a), 1–3 interactions (angles, figure 1.2b), and 1–4 interactions (dihedrals, figure 1.2c) and confine the atoms into molecules. Each term is some function of the property (b, θ, and ϕ) with parameters for a force constant (k_b, k_θ, and k_ϕ) and some equilibrium value of the property (b_0, θ_0, and m and δ). Non-bonded terms (figure 1.2d) are a pairwise summation of terms describing the electrostatic (q) and van der Waals (ε and σ) interactions, scaled by the cartesian distance between the atom pair, r_{ij}. Further discussion of the force field potential function will focus on the form employed in the GROMOS force fields.[33–36]

The basis of all these terms is the choice of atom types; i.e. the selection of atoms which are chemically and physically similar enough to be grouped together. Unlike in the quantum realm, where the "atom types" are defined by the number of electrons (thus, barring isotopic differences, there is only one atom type per element), classical force fields can have many different atom types for the same element. During the course of a simulation, there is no way for the type assigned to an atom to change, meaning all the parameter values remain constant throughout. For example, a carbonyl oxygen atom is a different atom type to an alcohol oxygen atom, and will never change into an alcohol oxygen regardless of the electronic environment it is located in at an arbitrary time step. As such, atom type choices are a trade off between the ability to accurately describe each atom, and having a manageable number of atom types.

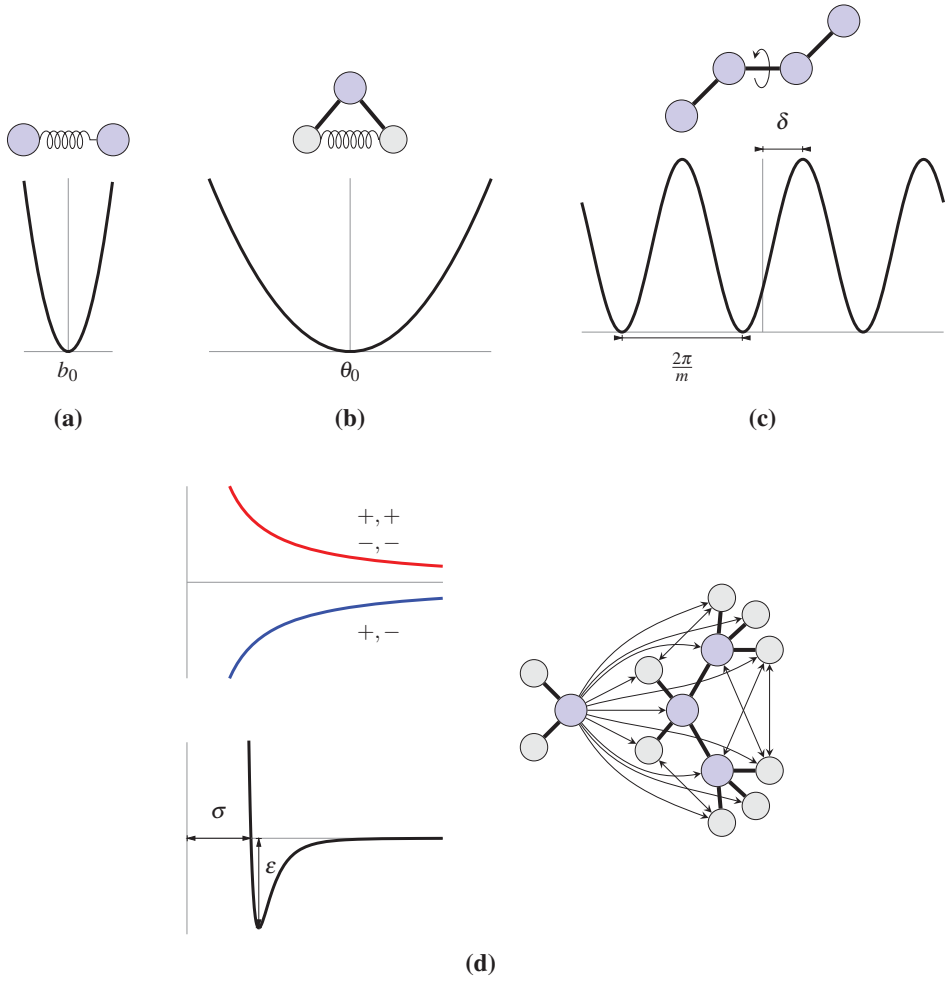


Figure 1.2: Diagrammatic representation of force field terms. (a) shows the bond stretch term for the distance between two bonded atoms, (b) shows the angle bend term defined as the angle θ between two atoms bonded to a central atom, (c) shows the dihedral term defined as the angle ϕ between two bonds through rotation about a central bond, and (d) shows the pairwise non-bonded interactions between atoms, with the upper graph showing the electrostatic interaction and the lower graph showing the van der Waals interaction. Graphs show qualitative plots of the energy each term has in the general form (detailed below), with the x-axis being the relevant property, and the y-axis showing the energy.

In addition to having different atom types for chemically different atoms within the force field, many force fields were originally parameterised as united atom (UA) force fields. These UA force fields were a trade off between computational efficiency and accuracy. Non-polar hydrogen atoms, generally only those bonded to carbon or sulfur, are incorporated into their parent atom, which decreases the computational load of the system by up to a factor of four. Though the

UA approach gives similar results to all atom (AA) approaches, where all the hydrogen atoms are explicitly included,[21] there are a number of deficiencies in the method. Amongst these are that a UA force field can "hide" the quadrupolar charge distribution of aromatic rings as the opposite charges on carbon and hydrogen atoms are not separated as they should be, and that it is difficult to compare computed vibrational frequencies with those measured experimentally.[37] Also, computation of properties requiring explicit hydrogen atom positions, such as ^{1}H NMR shifts and deuterium order parameters, becomes potentially problematic as hydrogen atom positions need to be built, and properties may be highly dependent on the geometry in which the hydrogen atoms are built. These problems, coupled with the improvement in compute power, have lead most force fields to step away from a UA model and embrace AA systems. The GROMOS force fields are a notable exception to this trend.

1.2.1. Bonded Terms

Along with dihedral terms, bond stretch and angle bend terms (figure 1.2a and b) define the geometry of a system. Measurements of these terms are able to be made in a well defined manner both experimentally and computationally, and as such there is an abundance of data available to use as a reference. Equilibrium parameter values for the bond stretch and angle bends are required, b_0 and θ_0 respectively, as well as force constants (k_b and k_θ). Both bond stretching and angle bending typically have the functional form of a harmonic potential, giving the potential function for the set of bonds N_b as

$$\sum_{\mathrm{N}_b} \mathcal{V}^b\left(b;k_b,b_0\right) = \sum_{\mathrm{N}_b} \frac{k_b}{2}\left(b_{ij}-b_0\right)^2 \tag{1.10}$$

and for the set of angles N_θ as

$$\sum_{\mathrm{N}_\theta} \mathcal{V}^\theta\left(\theta;k_\theta,\theta_0\right) = \sum_{\mathrm{N}_\theta} \frac{k_\theta}{2}\left(\theta_{ijk}-\theta_0\right)^2 \tag{1.11}$$

where b_{ij} is the distance between atoms i and j for bond stretches, θ_{ijk} is the angle defined by atoms i, j and k for angle bends and N_x is the relevant set of bond or angle terms. The forces on atoms i and j due to bond b_n are then

$$\begin{aligned}\mathbf{F}_i &= -\frac{\partial \mathcal{V}_n^b}{\partial b_n}\frac{\partial b_n}{\partial \mathbf{r}_i} \\ &= -k_{b_n}\left(b_{ij}-b_{0_n}\right)\frac{\mathbf{r}_{ij}}{r_{ij}}\end{aligned} \tag{1.12}$$

$$\text{and} \quad \mathbf{F}_j = -\mathbf{F}_i \tag{1.13}$$

and the forces $\mathbf{F}$ on atoms i, j and k due to the angle θ_n are

$$\begin{aligned}\mathbf{F}_i &= -\frac{\partial \mathcal{V}_n^\theta}{\partial \theta_n}\frac{\partial \theta_n}{\partial \mathbf{r}_i} \\ &= k_{\theta_n}\frac{\theta_{ijk}-\theta_{n_0}}{\sin\theta_{ijk}}\frac{1}{r_{ij}}\left(\frac{\mathbf{r}_{kj}}{r_{kj}}-\frac{\mathbf{r}_{ij}\cos\theta_{ijk}}{r_{ij}}\right) \qquad (1.14)\end{aligned}$$

$$\text{and} \quad \mathbf{F}_k = k_{\theta_n}\frac{\theta_{ijk}-\theta_{n_0}}{\sin\theta_{ijk}}\frac{1}{r_{kj}}\left(\frac{\mathbf{r}_{ij}}{r_{ij}}-\frac{\mathbf{r}_{kj}\cos\theta_{ijk}}{r_{kj}}\right) \qquad (1.15)$$

$$\text{and} \quad \mathbf{F}_j = -\mathbf{F}_i - \mathbf{F}_k \qquad (1.16)$$

In all cases, $\mathbf{r}_{xy}$ is the vector between atoms x and y and r_{xy} is its magnitude. For a slight increase in computational efficiency, a quartic bond stretch potential or cosine-harmonic angle bend potential can be used, though the effect is minimal.

Gas phase molecular structures obtained from microwave and electron data, as well as crystal structures from X-ray data, provide equilibrium parameter values for bond and angle parameters. An optimal fitting procedure would use both sources of reference data, with parameter values optimised to match gas phase results (where there are only intramolecular forces involved), and then tested to reproduce the crystal data, where intermolecular forces play an important role.[18] Force constant parameter values come from fitting to vibrational data, generally sourced from gas-phase infra-red and Raman spectroscopies, with normal mode calculations, though solution or crystal data can be used if needed. Quantum mechanical calculations provide an additional means of generating reference data. At a reasonable level of theory, which is determined as a trade off between computational effort and the knowledge that a lot of the information obtained will be "lost" in the process of transferring from the quantum realm to classical, results obtained from quantum mechanical calculations are perfectly valid as reference data.

Dihedral terms are generally the last terms fitted when developing a force field. The form of the potential function is

$$\sum_{\mathrm{N}_\phi}\mathcal{V}^\phi\left(\phi;k_\phi,m,\delta\right) = \sum_{\mathrm{N}_\phi}k_\phi\left[1+\cos\left(m\phi-\delta\right)\right] \qquad (1.17)$$

where N_ϕ is the set of defined dihedral terms; k_ϕ is the rotational force constant and defines the energy of the rotation; m is the multiplicity of the torsion which determines the number of potential wells in the energy profile, generally related to the number of non-hydrogen groups attached to each end of the central bond; and δ is the phase shift, which allows for shifting of the angular location of wells. This leads to the forces on atoms i, j, k and l due to the dihedral

ϕ_n being

$$\begin{aligned}\mathbf{F}_i &= -\frac{\partial \mathcal{V}_n^{\phi}}{\partial \phi_n}\frac{\partial \phi_n}{\partial \mathbf{r}_i} \\ &= k_{\phi_n} m_n \sin\left(m_n \phi_{ijkl} - \delta_n\right)\frac{r_{kj}\mathbf{r}_{mj}}{r_{mj}^2} \qquad (1.18)\end{aligned}$$

$$\mathbf{F}_l = k_{\phi_n} m_n \sin\left(m_n \phi_{ijkl} - \delta_n\right)\frac{r_{kj}\mathbf{r}_{nk}}{r_{nk}^2} \qquad (1.19)$$

$$\mathbf{F}_j = \left[\frac{\mathbf{r}_{ij}\cdot\mathbf{r}_{kj}}{r_{kj}^2} - 1\right]\mathbf{F}_i - \frac{\mathbf{r}_{kl}\cdot\mathbf{r}_{kj}}{r_{kj}^2}\mathbf{F}_l \qquad (1.20)$$

$$\text{and} \quad \mathbf{F}_k = -\mathbf{F}_i - \mathbf{F}_j - \mathbf{F}_l \qquad (1.21)$$

with $\mathbf{r}_{mj}$ and $\mathbf{r}_{nk}$ calculated according to

$$\begin{aligned}\mathbf{r}_{mj} &\coloneqq \mathbf{r}_{ij} \times \mathbf{r}_{kj} \quad \text{and} \quad r_{mj} \coloneqq \left(\mathbf{r}_{mj}\cdot\mathbf{r}_{mj}\right)^{\frac{1}{2}} \\ \mathbf{r}_{nk} &\coloneqq \mathbf{r}_{kj} \times \mathbf{r}_{kl} \quad \text{and} \quad r_{nk} \coloneqq \left(\mathbf{r}_{nk}\cdot\mathbf{r}_{nk}\right)^{\frac{1}{2}} \qquad (1.22)\end{aligned}$$

In contrast to the other bonded terms, multiple terms can be applied to each torsion, generating a Fourier series which better matches reference data.

In principle, parameter values for dihedral terms can be derived from similar data to that used for bond and angle terms, however there are issues. Rotations about bonds can result in large scale changes to the conformational structure of a molecule. Consequently there is a large degree of coupling between the dihedral and the non-bonded parameter values. Whereas bond and angle parameter values are transferable to some extent between force fields, this coupling means that dihedral terms are not, and need to be derived separately for differing charge distributions. Furthermore, experimental results on which to base dihedral parameter values are scarce. AMBER terms were originally obtained from experimental conformational equilibria of molecules[21] but were modified due to the coupling with non-bonded parameters. A more general approach, as taken by Jorgensen *et al.* during development of the OPLS-AA force field,[29] is to fit a Fourier series to the potential obtained from quantum mechanical calculations of rotation about the bond in question. The fit is performed such that the potential obtained using the force field reproduces that of the quantum dihedral rotation scan, taking into account the non-bonded terms.

Within the GROMOS force fields, there is an additional bonded term known as an "improper" dihedral, where the four atoms defining the dihedral are required to be sequentially connected by covalent bonds, and a harmonic functional form is used for the potential. Improper dihedrals are used to enforce planarity around an sp^2 carbon atom centre, or tetrahedral geometry, as well as maintain the desired chirality of a centre within UA force fields. The functional form taken is

$$\sum_{N_\xi} \mathcal{V}^\xi\left(\xi; k_\xi, \xi_0\right) = \sum_{N_\xi} \frac{k_\xi}{2}\left(\xi - \xi_0\right)^2 \quad \text{with} \quad -\pi \leq \xi - \xi_0 \leq \pi \tag{1.23}$$

where N_ξ is the set of all improper terms, k_ξ is the force constant and ξ_0 is the equilibrium value of the improper dihedral ξ_{ijkl}. Consequently, the forces on atoms i, j, k and l due to the improper dihedral ξ_n are

$$\begin{aligned}
\mathbf{F}_i &= -\frac{\partial \mathcal{V}_n^\xi}{\partial \xi_n}\frac{\partial \xi_n}{\partial \mathbf{r}_i} \\
&= k_{\xi_n}\left(\xi_{ijkl} - \xi_{n_0}\right)\frac{r_{kj}\mathbf{r}_{mj}}{r_{mj}^2} && (1.24)\\
\mathbf{F}_l &= k_{\xi_n}\left(\xi_{ijkl} - \xi_{n_0}\right)\frac{r_{kj}\mathbf{r}_{nk}}{r_{nk}^2} && (1.25)\\
\mathbf{F}_j &= \left[\frac{\mathbf{r}_{ij}\cdot\mathbf{r}_{kj}}{r_{kj}^2} - 1\right]\mathbf{F}_i - \frac{\mathbf{r}_{kl}\cdot\mathbf{r}_{kj}}{r_{kj}^2}\mathbf{F}_l && (1.26)\\
\text{and} \quad \mathbf{F}_k &= -\mathbf{F}_i - \mathbf{F}_j - \mathbf{F}_l && (1.27)
\end{aligned}$$

with $\mathbf{r}_{mj}$ and $\mathbf{r}_{nk}$ calculated according to equation 1.22.

1.2.2. Non-bonded Terms

The majority of computational time during a molecular dynamics time step is spent on calculating the non-bonded interactions. In theory each pair of atoms should have an electrostatic and a van der Waals interaction calculated. In practice, this is not always the case. Due to the short distance between 1–2 and 1–3 bonded pairs, the non-bonded interaction term for these pairs will be very large and is therefore often excluded from the final energy term. Similarly, the 1–4 non-bonded interactions can use slightly different parameter values for the van der Waals interaction than those for atom pairs separated by more than three covalent bonds. At the opposite end of the spectrum are atom pairs which are far apart. At large distances, the potential between two atoms tends toward zero. To exploit this and improve computational efficiency a cut-off distance is used. Beyond this cut-off distance, the interactions between atom pairs are not computed.

To determine the potential due to van der Waal interactions, each atom type is assigned an ε_i value, the depth of the potential well, and a σ_i value, the "size" of the atom type as it is the finite distance at which the inter-particle potential is zero. In the GROMOS force fields, van der Waals

interactions take the form of a 6–12 potential:

$$\sum_i \sum_j \mathcal{V}^{\text{vdw}}(\mathbf{r}_{ij};\varepsilon_{ij},\sigma_{ij}) = \sum_i^{N-1} \sum_{j=i+1}^{N} 4\varepsilon_{ij}\left[\left(\frac{\sigma_{ij}}{\mathbf{r}_{ij}}\right)^{12} - \left(\frac{\sigma_{ij}}{\mathbf{r}_{ij}}\right)^{6}\right] \tag{1.28}$$

where ε_{ij} and σ_{ij} are derived from single atom ε_i and σ_i values by the use of the geometrical average

$$\begin{aligned}\sigma_{ij} &= (\sigma_i\sigma_j)^{\frac{1}{2}} \\ \varepsilon_{ij} &= (\varepsilon_i\varepsilon_j)^{\frac{1}{2}}.\end{aligned} \tag{1.29}$$

A different combination rule, known as Lorentz-Berthelot mixing, could alternatively be used:

$$\begin{aligned}\sigma_{ij} &= \frac{1}{2}(\sigma_i + \sigma_j) \\ \varepsilon_{ij} &= (\varepsilon_i\varepsilon_j)^{1/2}.\end{aligned} \tag{1.30}$$

Therefore, the forces experienced by atoms i and j due to the van der Waals interaction between them are

$$\begin{aligned}\mathbf{F}_i &= -\frac{\partial \mathcal{V}^{\text{vdw}}}{\partial \mathbf{r}_i} \\ &= \frac{48\varepsilon_{ij}}{r_{ij}^2}\left[\left(\frac{\sigma_{ij}}{r_{ij}}\right)^{12} - \frac{1}{2}\left(\frac{\sigma_{ij}}{r_{ij}}\right)^{6}\right]\mathbf{r}_{ij}\end{aligned} \tag{1.31}$$

$$\text{and} \quad \mathbf{F}_j = -\mathbf{F}_i \tag{1.32}$$

In a 6–12 potential, the r^{-6} term describes the attractive long-range interaction and is physically well justified. The r^{-12} term provides the short range repulsion but has no physical justification. It approximates Pauli repulsion, but was originally used as it is easy to obtain from the r^{-6} term, even though a 6–exp function would provide a better representation of the potential.[38] Use of the 6–12 functional form has continued even as computer power has increased markedly over the years since force fields were originally developed.

The second component of the non-bonded terms are the electrostatics. The electrostatic term of equation 1.9 can be partitioned into three separate terms; a *surface* term, a *pairwise* term and a *self* interaction term, as given by

$$\mathcal{V}^{\text{elec}}(\mathbf{r};q) = \mathcal{V}^{\text{srf}}(\mathbf{r};q) + \mathcal{V}^{\text{pws}}(\mathbf{r};q) + \mathcal{V}^{\text{self}}(\mathbf{r};q) \tag{1.33}$$

The form of these terms varies depending on the scheme used to evaluate the long range electrostatic interactions. Such schemes include reaction-field,[39,40] Ewald summation,[41] and Particle-mesh Ewald (PME) summation.[42] The GROMOS force fields were parameterised for use with the reaction-field scheme, which is explained here.

The surface term arises from the definition of the medium surrounding an infinite periodic system such as when periodic spatial boundary conditions are utilised. It is defined as

$$\mathcal{V}^{\mathrm{srf}}(\mathbf{r};q) \coloneqq \frac{\mathbf{M}^2}{2\pi\varepsilon_0(2\varepsilon_R+1)V} \tag{1.34}$$

where ε_0 is the permittivity of vacuum, ε_R is the relative permittivity of the medium in which the simulation is performed, V is the volume of the simulation box, and $\mathbf{M}$ is the dipole moment of the box.

The pairwise term is the most expensive and is only evaluated for atom pairs which are not excluded, i.e. 1–2 and 1–3 atom pairs, and if the distance between the atoms is less than the cut-off distance. It is given by

$$\mathcal{V}^{\mathrm{pws}}(\mathbf{r};q) = \frac{1}{4\pi\varepsilon_0\varepsilon_R}\sum_{i}^{N-1}\sum_{j=i+1}^{N} q_i q_j \psi_{ij}(\mathbf{r}) \tag{1.35}$$

where q_i is the charge on atom i and $\psi_{ij}(\mathbf{r})$ is the *electrostatic influence function* associated with the atom pair ij

$$\psi_{ij}(\mathbf{r}) \coloneqq \frac{1}{r_{ij}} - \frac{Cr_{ij}^2}{2R_F^3} - \frac{1-\frac{1}{2}C}{R_F} \tag{1.36}$$

$$C \coloneqq \frac{(2\varepsilon_R-2\varepsilon_F)(1+\kappa R_F)-\varepsilon_F(\kappa R_F)^2}{(\varepsilon_R+2\varepsilon_F)(1+\kappa R_F)+\varepsilon_F(\kappa R_F)^2} \tag{1.37}$$

where ε_F is the permittivity of the reaction-field, κ is the inverse Debye screening length and R_F is the radius of the reaction-field. Finally, the self interaction term is given by

$$\mathcal{V}^{\mathrm{self}}(\mathbf{r};q) = \frac{1}{8\pi\varepsilon_0\varepsilon_R}\sum_{i}^{N} -\frac{q_i^2\left(1-\frac{1}{2}C\right)}{R_F} \tag{1.38}$$

With all terms combined together, the force on atoms i and j due to the reaction-field electrostatic interaction between them is given by

$$\mathbf{F}_i = \frac{q_i q_j}{4\pi\varepsilon_0\varepsilon_R}\left(\frac{1}{r_{ij}^3}+\frac{Cr_{ij}}{R_F}\right)\mathbf{r}_{ij} \tag{1.39}$$

$$\text{and} \quad \mathbf{F}_j = -\mathbf{F}_i \tag{1.40}$$

Compared with the more realistic PME scheme for calculating long range electrostatic interactions, the reaction-field scheme has some deficiencies. PME is fundamentally a computationally efficient means to approximate a solution to the Ewald summation problem, which gives the exact electrostatics but is slow to converge. As such, PME describes the electrostatics of heterogeneous systems much more realistically than a reaction-field, which assumes a constant dielectric continuum, does. However, this comes at a much greater computational cost. Additionally, it has been shown that small changes in the force field terms have a much greater effect on simulations than any difference due to treating the long range electrostatics differently.**Poger2012**

Additional non-bonded components can be included within the force field. For example, AMBER was originally developed with an additional 10–12 term to allow greater control over the hydrogen bonding interactions within systems.[21,22] This term was later removed by Cornell *et al.* as it was no longer necessary due to improvements in the electrostatic and van der Waals parameter values.[23]

Parameter values for the non-bonded components of force fields are the most difficult, and in some ways the most important, values to define. Electronic point charges are relatively simple to obtain. Quantum mechanical calculations can be performed on the molecule, at an appropriate level of theory, and the electronic distribution obtained collapsed into point charges for each of the atoms. This approach does have a number of drawbacks.

Firstly, it is well known that charge distributions are highly conformationally dependent.[43–46] These charge variances are not small. Stouch and Williams noted variation of nearly one electronic unit in the charge on a methylene carbon in glycerylphosphorylcholine, depending on the conformation.[43] Such a dependence indicates that point charges derived from a single conformation may not represent the molecule sufficiently accurately during a molecular dynamics simulation. The obvious solution to this is to fit the charges based on a number of conformations. Cornell *et al.* have shown that such multi-conformational derived charges do have improved results over single conformational charges.[47] This will not always be the case though as the choices of which conformations to use will have an effect on the results obtained. Unwisely chosen source conformations will give charges just as poorly transferable as single conformation charges. In general, all of conformational space is too large to sample efficiently, so informed decisions on which conformations to use need to be made.

Secondly, the symmetry of a molecule plays an important role in the charge distribution. Quantum mechanical calculations on an isolated benzene molecule in implicit water would be expected to produce similar charges on all the hydrogen atoms. Figure 1.3 shows that there are dramatic differences between the charges assigned to the benzene hydrogen atoms when no

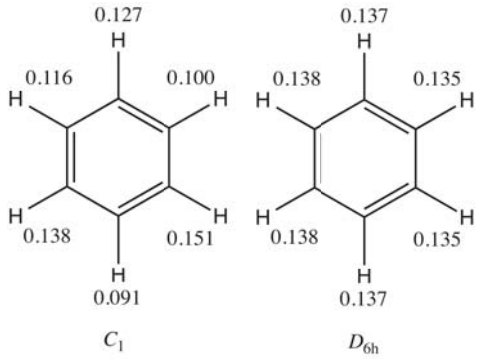


Figure 1.3: The partial atomic charges of hydrogen atoms in benzene derived by fitting to the electrostatic potential at B3LYP/6-31G* level of theory optimised assuming C_1 and D_{6h} symmetry using the Kollmann-Singh method.

symmetry is enforced (C_1) compared with when D_{6h} symmetry is enforced. Symmetry, both local to certain areas of the molecule and global, must be considered when assigning point charges to atoms.

Finally, the point charges obtained are highly dependent on the level of quantum theory used to calculate the electronic density, in particular the basis set used. Pople's 6-31G* basis set[48] is a popular choice, and is used in the derivation of a number of force fields.[18,21,23,49,50] The 6-31G* basis set is known to uniformly overestimate molecular polarity in the gas-phase. However, molecules in the condensed phase are expected to be more polarised than those in the gas phase. As such, Kuyper *et al.* suggested that the 6-31G* basis set is the logical choice for deriving partial charges.[51]

Transferability is another consideration. OPLS is developed so that functional group subunits are neutrally charged overall, making derivation of charges for novel molecules trivial. In contrast, AMBER has charges derived on a case-by-case basis.[23] Which option is chosen is a matter of preference, though transferability of functional group charges between different kinds of molecules may not always be applicable.

Van der Waals parameter values are more difficult to define than the electrostatic term. Rappe *et al.* developed a means to systematically derive van der Waals parameter values in their Universal Force Field (UFF), based mainly on the element type with some empirical data included.[52] In general though, van der Waals parameter values are derived by an initial estimate of reasonable values, then refined through empirical means until selected reference data can be reproduced.

There are a number of potential sources for reference data against which to parameterise the van der Waals terms. Hagler *et al.* derived parameters for sp^2 atoms from fits of lattice energies and crystal structures in amides,[53] Dunfield *et al.* determined 6–12 parameters for UA

CH, CH_2 and CH_3 parameters based on crystal packing calculations of hydrocarbons,[54] and Jorgensen calculated parameters for OPLS from Monte Carlo liquid simulations of ethers and alcohols.[25] With a wide range of experimental data available, there is no single right type of data to aim to reproduce. AMBER is parameterised to fit experimental conformational and vibrational energy profiles,[23] whereas GROMOS is parameterised to reproduce liquid densities, heats of vaporisation and free energies of hydration.[33,34,36] The fact that different force fields exist and can generate results consistent with their designed purpose shows that there is no one right philosophy when it comes to force field parameterisation.

1.3. Other Force Field Types

The majority of force fields mentioned above follow the same functional form as shown in equation 1.9. Several ways to improve upon this functional form are available. Firstly, the intramolecular terms of standard force fields can be improved upon. As mentioned earlier, a number of force fields include improper torsional terms which aid in maintaining chirality and out-of-plane vibrational modes. Such terms can be defined between any arbitrary set of atoms, though generally they are defined between non-interconnected atoms around the same centre.

Some force fields include additional bonded terms to those shown in equation 1.9. For instance, CHARMM has an additional Urey-Bradley term for in-plane deformations, and separating symmetric and asymmetric bond stretching.[18,19] Only the quadratic portion of the Urey-Bradley function is used as the linear term can be excluded.[55] MMFF94 goes even further. The bond stretch term is expanded to include cubic and quartic components as well as the general quadratic portion, the angle bend term is expanded to include a cubic portion or, in the case of near-linear bond angles, replaced with a sinusoidal term.[56] Additional terms for representing the coupling between the *i*–*j* and *k*–*j* bond stretches and the *i*–*j*–*k* angle bend as well as out-of-plane bending at tricoordinate centres are included. Non-bonded terms are also modified, employing a "buffered" form for both the van der Waals and electrostatic terms.[57] Such improvements come at a computational cost, so as always, using a force field that better represents the energy of a system has to be traded off with the ability to simulate larger systems for longer time periods.

In the majority of standard force fields, a major limitation of the intermolecular terms is the use of fixed atomic point charges. Electron distributions are inherently anisotropic, for example with lone pairs of electrons and delocalised π-bond clouds. The isotropic nature of fixed atomic point charges means they lack the mathematical flexibility to describe certain features of molecular charge distributions, such as the anisotropic nature, and are unable to respond to changes in the molecular environment, for example through conformational changes. No amount of reparameterisation can change this basic fact, meaning fixed atomic point charge model force fields

will never be able to describe the electrostatics of general polar molecules to within chemical accuracy. Polarisable force fields extend the fixed atomic point charge concept to allow for some degree of polarisation of the electron distribution.

There are a number of existing methods which add polarisation effects to a force field. Fluctuating charges/charge equilibration allows individual atomic partial charges to change over the course of a simulation. By treating the charges as additional degrees of freedom, charges are allowed to flow between atoms until instantaneous electronegativities are neutralised.[58] The CHEQ force field of Bauer and Patel implements this method of polarisation, and is developed within the CHARMM program.[59,60] A fluctuating charges implementation has very little additional computational overhead when compared with a fixed point charge force field, though as charges flow along bonds only, it cannot easily represent polarisation orthogonal to bonds, as is required to correctly represent planar molecules such as benzene. In principle, this could be alleviated by adding additional point charges to represent charge density not localised to specific atoms.[61]

Drude oscillators represent charges on each atomic centre as a pair of point charges, the sum of which gives the total charge on the atom. The first charge is centred on the nucleus and the second is a massless particle (Drude particle) attached to the nucleus by a spring.[62,63] This approach is relatively easy to incorporate into existing force fields but because of the large number of extra charges added, computational cost is increased markedly. MacKerell and Roux's groups have developed a force field based on the Drude model.[64–66] In this model, polarisability is determined solely by the charge on the Drude particle (as the force constant on the springs is the same across all atom types), and additional point charges with fixed magnitude and location are included to allow for better representation of hydrogen bonding.

The AMBER ff02 force field includes polarisation through the use of inducible dipoles.[67] Fixed atomic charges are retained, with additional inducible point dipoles added, generally to the atomic nuclei,[68] but occasionally to the bonds between atoms.[69] The induced dipole at a particular site is determined by the electric field at that site. Use of inducible dipoles means that extra terms to account for the various interactions between dipoles and charges have to be introduced into the force field. This makes implementation a challenging process, though the parameterisation of such a force field is relatively straight forward[61]. The Polarisable Simulations with Second order Interaction Model (POSSIM) also employs an inducible dipoles model.[70–73] Software developed along with the force field allows for a speed up of the polarisable component of the calculations by around an order of magnitude, without loss of accuracy, when compared with traditional evaluation procedures.

Finally, a more rigorous approach is to do away with point charges all together. By using multipole moments that include terms up to hexadecapoles, electronic charge density can be

modelled in a way that naturally captures the anisotropic and non-spherical nature of the density.[74,75] AMOEBA is the most widely used force field that includes multipole electrostatics.[76–78] Terms up to quadrupole moments are included. AMOEBA has been used to generate incredibly accurate peptide electrostatic properties,[79] results which would be impossible to obtain with a non-polarisable force field. Of course, this accuracy comes at a cost. AMOEBA is a computationally intensive force field, meaning a choice must be made between the accuracy it provides, and the simulation system sizes and timescales attainable with a simpler model.

1.4. Free Energy

For a dynamic system kept at constant particle number, volume and temperature, the probability of finding the system in some state i is given by:

$$P_i = \frac{\exp^{-\beta E_i}}{Z} \tag{1.41}$$

where $\beta = {}^{1}/_{k_B T}$, E_i is the energy of state i, and $Z = \sum_i \exp^{-\beta E_i}$ and is known as the partition function. From this, the average internal energy of the system, U, can be calculated as

$$U = \sum_i P_i E_i \tag{1.42}$$

and the entropy of the system, S, can be determined as

$$S = -k_B \sum_i P_i \log P_i. \tag{1.43}$$

The Helmholtz free energy, A, is defined as

$$A = U - TS \tag{1.44}$$

Substituting equations 1.41 to 1.43 into equation 1.44, it can be shown that the Helmholtz free energy is equivalent to

$$A = -kT \log Z. \tag{1.45}$$

As such, if the partition function for a system can be determined, the free energy of that system can be calculated. Molecular dynamics provides a means to generate samples of system configurations which, given sufficient sampling, can be used to directly calculate the free energy.

1.5. Local Elevation

Molecular dynamics simulations are a method to search the conformational space of a molecule. Often times, there are large energy barriers on the conformational energy surface which are difficult for a general simulation to traverse. This leads to simulations having a tendency to repeatedly sample from a small portion of conformational space. To overcome these large energy barriers and avoid excessive resampling, a penalty potential can be added to bias conformations away from those already visited. The local elevation method is designed as a means to produce such a penalty potential by gradually adding small, local, repulsive potentials during the simulation.[80] Let χ be a conformation and χ^0 be a previously visited conformation. Using a Gaussian function, the bias potential for that conformation can then be given as:

$$\mathcal{V}(\chi) = k n_{\chi^0} \exp^{\frac{-(\chi-\chi^0)^2}{2w^2}} \tag{1.46}$$

where n^0_χ is the number of times the conformation has been previously sampled, k (> 0) is the magnitude and w the width of the penalty function.

In practice, equation 1.46 works as follows. A conformation can be described as a set of dihedral angles, each with a corresponding dihedral term. For practical reasons, the periodic range of values that a dihedral can take is divided into a grid. At each step of a simulation, the dihedral angles are measured and the corresponding grid point counter incremented by one. This increases the penalty potential of that grid point, thereby biasing further conformations away from the grid point.

Bibliography

(1) B. J. Alder and T. E. Wainwright, "Studies in Molecular Dynamics. I. General Method", *The Journal of Chemical Physics*, 1959, **31**, 459–466.

(2) S. Pronk, S. Páll, R. Schulz, P. Larsson, P. Bjelkmar, R. Apostolov, M. R. Shirts, J. C. Smith, P. M. Kasson, D. van der Spoel, B. Hess and E. Lindahl, "GROMACS 4.5: a high-throughput and highly parallel open source molecular simulation toolkit", *Bioinformatics*, 2013, **29**, 845–854.

(3) F. Fraternali and W. van Gunsteren, "An Efficient Mean Solvation Force Model for Use in Molecular Dynamics Simulations of Proteins in Aqueous Solution", *Journal of Molecular Biology*, 1996, **256**, 939–948.

(4) W. F. Van Gunsteren and M. Karplus, "Protein dynamics in solution and in a crystalline environment: a molecular dynamics study", *Biochemistry*, 1982, **21**, 2259–2274.

(5) W. van Gunsteren, "Constrained dynamics of flexible molecules", *Molecular Physics*, 1980, **40**, 1015–1019.

(6) B. Hess, H. Bekker, H. J. C. Berendsen and J. G. E. M. Fraaije, "LINCS: A linear constraint solver for molecular simulations", *Journal of Computational Chemistry*, 1997, **18**, 1463–1472.

(7) B. Hess, "P-LINCS: A Parallel Linear Constraint Solver for Molecular Simulation", *Journal of Chemical Theory and Computation*, 2008, **4**, 116–122.

(8) L. Woodcock, "Isothermal molecular dynamics calculations for liquid salts", *Chemical Physics Letters*, 1971, **10**, 257–261.

(9) H. J. C. Berendsen, J. P. M. Postma, W. F. van Gunsteren, A. DiNola and J. R. Haak, "Molecular dynamics with coupling to an external bath", *The Journal of Chemical Physics*, 1984, **81**, 3684–3690.

(10) G. Bussi, D. Donadio and M. Parrinello, "Canonical sampling through velocity rescaling", *The Journal of Chemical Physics*, 2007, **126**, 014101:1–7.

(11) S. Nosé, "A molecular dynamics method for simulations in the canonical ensemble", *Molecular Physics*, 1984, **52**, 255–268.

(12) W. G. Hoover, "Canonical dynamics: Equilibrium phase-space distributions", *Physical Review A*, 1985, **31**, 1695–1697.

(13) H. C. Andersen, "Molecular dynamics simulations at constant pressure and/or temperature", *The Journal of Chemical Physics*, 1980, **72**, 2384–2393.

(14) S. Nosé and M. Klein, "Constant pressure molecular dynamics for molecular systems", *Molecular Physics*, 1983, **50**, 1055–1076.

(15) M. Parrinello and A. Rahman, "Polymorphic transitions in single crystals: A new molecular dynamics method", *Journal of Applied Physics*, 1981, **52**, 7182–7190.

(16) B. R. Brooks, R. E. Bruccoleri, B. D. Olafson, D. J. States, S. Swaminathan and M. Karplus, "CHARMM: A program for macromolecular energy, minimization, and dynamics calculations", *Journal of Computational Chemistry*, 1983, **4**, 187–217.

(17) W. E. Reiher, "Theoretical studies of hydrogen bonding", Ph.D. Thesis, Harvard University, 1985.

(18) A. D. MacKerell, D. Bashford, M. Bellott, R. L. Dunbrack, J. D. Evanseck, M. J. Field, S. Fischer, J. Gao, H. Guo, S. Ha, D. Joseph-McCarthy, L. Kuchnir, K. Kuczera, F. T. K. Lau, C. Mattos, S. Michnick, T. Ngo, D. T. Nguyen, B. Prodhom, W. E. Reiher, B. Roux, M. Schlenkrich, J. C. Smith, R. Stote, J. Straub, M. Watanabe, J. Wirkiewicz-Kuczera, D. Yin and M. Karplus, "All-Atom Empirical Potential for Molecular Modeling and Dynamics Studies of Proteins", *The Journal of Physical Chemistry B*, 1998, **102**, 3586–3616.

(19) A. D. MacKerell, N. Banavali and N. Foloppe, "Development and current status of the CHARMM force field for nucleic acids", *Biopolymers*, 2000, **56**, 257–265.

(20) N. Foloppe and A. D. MacKerell, Jr., "All-atom empirical force field for nucleic acids: I. Parameter optimization based on small molecule and condensed phase macromolecular target data", *Journal of Computational Chemistry*, 2000, **21**, 86–104.

(21) S. J. Weiner, P. A. Kollman, D. A. Case, U. C. Singh, C. Ghio, G. Alagona, S. Profeta and P. Weiner, "A new force field for molecular mechanical simulation of nucleic acids and proteins", *Journal of the American Chemical Society*, 1984, **106**, 765–784.

(22) S. J. Weiner, P. A. Kollman, D. T. Nguyen and D. A. Case, "An all atom force field for simulations of proteins and nucleic acids", *Journal of Computational Chemistry*, 1986, **7**, 230–252.

(23) W. D. Cornell, P. Cieplak, C. I. Bayly, I. R. Gould, K. M. Merz, D. M. Ferguson, D. C. Spellmeyer, T. Fox, J. W. Caldwell and P. A. Kollman, "A Second Generation Force Field for the Simulation of Proteins, Nucleic Acids, and Organic Molecules", *Journal of the American Chemical Society*, 1995, **117**, 5179–5197.

(24) Y. Duan, C. Wu, S. Chowdhury, M. C. Lee, G. Xiong, W. Zhang, R. Yang, P. Cieplak, R. Luo, T. Lee, J. Caldwell, J. Wang and P. Kollman, "A point-charge force field for molecular mechanics simulations of proteins based on condensed-phase quantum mechanical calculations", *Journal of Computational Chemistry*, 2003, **24**, 1999–2012.

(25) W. L. Jorgensen, "Quantum and statistical mechanical studies of liquids. 10. Transferable intermolecular potential functions for water, alcohols, and ethers. Application to liquid water", *Journal of the American Chemical Society*, 1981, **103**, 335–340.

(26) W. L. Jorgensen, J. D. Madura and C. J. Swenson, "Optimized intermolecular potential functions for liquid hydrocarbons", *Journal of the American Chemical Society*, 1984, **106**, 6638–6646.

(27) W. L. Jorgensen and C. J. Swenson, "Optimized intermolecular potential functions for amides and peptides. Structure and properties of liquid amides", *Journal of the American Chemical Society*, 1985, **107**, 569–578.

(28) W. L. Jorgensen and J. Tirado-Rives, "The OPLS [optimized potentials for liquid simulations] potential functions for proteins, energy minimizations for crystals of cyclic peptides and crambin", *Journal of the American Chemical Society*, 1988, **110**, 1657–1666.

(29) W. L. Jorgensen, D. S. Maxwell and J. Tirado-Rives, "Development and Testing of the OPLS All-Atom Force Field on Conformational Energetics and Properties of Organic Liquids", *Journal of the American Chemical Society*, 1996, **118**, 11225–11236.

(30) G. A. Kaminski, R. A. Friesner, J. Tirado-Rives and W. L. Jorgensen, "Evaluation and Reparametrization of the OPLS-AA Force Field for Proteins via Comparison with Accurate Quantum Chemical Calculations on Peptides", *The Journal of Physical Chemistry B*, 2001, **105**, 6474–6487.

(31) R. C. Rizzo and W. L. Jorgensen, "OPLS All-Atom Model for Amines: Resolution of the Amine Hydration Problem", *Journal of the American Chemical Society*, 1999, **121**, 4827–4836.

(32) D. S. Maxwell, J. Tirado-Rives and W. L. Jorgensen, "A comprehensive study of the rotational energy profiles of organic systems by ab initio MO theory, forming a basis for peptide torsional parameters", *Journal of Computational Chemistry*, 1995, **16**, 984–1010.

(33) L. D. Schuler, X. Daura and W. F. van Gunsteren, "An improved GROMOS96 force field for aliphatic hydrocarbons in the condensed phase", *Journal of Computational Chemistry*, 2001, **22**, 1205–1218.

(34) C. Oostenbrink, A. Villa, A. E. Mark and W. F. Van Gunsteren, "A biomolecular force field based on the free enthalpy of hydration and solvation: The GROMOS force-field parameter sets 53A5 and 53A6", *Journal of Computational Chemistry*, 2004, **25**, 1656–1676.

(35) D. Poger, W. F. Van Gunsteren and A. E. Mark, "A new force field for simulating phosphatidylcholine bilayers", *Journal of Computational Chemistry*, 2010, **31**, 1117–1125.

(36) N. Schmid, A. Eichenberger, A. Choutko, S. Riniker, M. Winger, A. Mark and W. van Gunsteren, "Definition and testing of the GROMOS force-field versions 54A7 and 54B7", *European Biophysics Journal*, 2011, **40**, 843–856.

(37) J. W. Ponder and D. A. Case, in *Protein Simulations*, ed. V. Daggett, Academic Press, 2003, vol. 66, pp. 27–85.

(38) H. Margenau and N. R. Kestner, *Theory of Intermolecular Forces*, Pergamon Press, Oxford, Second, 1971.

(39) J. Barker and R. Watts, "Monte Carlo studies of the dielectric properties of water-like models", *Molecular Physics*, 1973, **26**, 789–792.

(40) R. Watts, "Monte Carlo studies of liquid water", *Molecular Physics*, 1974, **28**, 1069–1083.

(41) P. P. Ewald, "Die Berechnung optischer und elektrostatischer Gitterpotentiale", *Annalen der Physik*, 1921, **369**, 253–287.

(42) T. Darden, D. York and L. Pedersen, "Particle mesh Ewald: An Nlog(N) method for Ewald sums in large systems", *The Journal of Chemical Physics*, 1993, **98**, 10089–10092.

(43) T. Stouch and D. E. Williams, "Conformational dependence of electrostatic potential derived charges of a lipid headgroup: Glycerylphosphorylcholine", *Journal of Computational Chemistry*, 1992, **13**, 622–632.

(44) T. R. Stouch and D. E. Williams, "Conformational dependence of electrostatic potential-derived charges: Studies of the fitting procedure", *Journal of Computational Chemistry*, 1993, **14**, 858–866.

(45) J. J. Urban and G. R. Famini, "Conformational dependence of the electrostatic potential-derived charges of dopamine: Ramifications in molecular mechanics force field calculations in the gas phase and in aqueous solution", *Journal of Computational Chemistry*, 1993, **14**, 353–362.

(46) U. Koch and A. J. Stone, "Conformational dependence of the molecular charge distribution and its influence on intermolecular interactions", *Journal of the Chemical Society, Faraday Transactions*, 1996, **92**, 1701–1708.

(47) W. D. Cornell, P. Cieplak, C. I. Bayly and P. A. Kollmann, "Application of RESP charges to calculate conformational energies, hydrogen bond energies, and free energies of solvation", *Journal of the American Chemical Society*, 1993, **115**, 9620–9631.

(48) R. Ditchfield, W. J. Hehre and J. A. Pople, "Self-Consistent Molecular-Orbital Methods. IX. An Extended Gaussian-Type Basis for Molecular-Orbital Studies of Organic Molecules", *The Journal of Chemical Physics*, 1971, **54**, 724–728.

(49) U. C. Singh and P. A. Kollman, "An approach to computing electrostatic charges for molecules", *Journal of Computational Chemistry*, 1984, **5**, 129–145.

(50) A. D. MacKerell, *Atomistic Models and Force Fields*, ed. O. M. Becker, A. D. MacKerell, Jr., B. Roux and M. Watanabe, Marcel Dekker, 2001, pp. 7–38.

(51) L. F. Kuyper, R. N. Hunter and D. Ashton, "Free energy calculations on the relative solvation free energies of benzene, anisole, and 1,2,3-trimethoxybenzene: theoretical and experimental analysis of aromatic methoxy solvation", *The Journal of Physical Chemistry*, 1991, **95**, 6661–6666.

(52) A. K. Rappe, C. J. Casewit, K. S. Colwell, W. A. Goddard and W. M. Skiff, "UFF, a full periodic table force field for molecular mechanics and molecular dynamics simulations", *Journal of the American Chemical Society*, 1992, **114**, 10024–10035.

(53) A. T. Hagler, E. Huler and S. Lifson, "Energy functions for peptides and proteins. I. Derivation of a consistent force field including the hydrogen bond from amide crystals", *Journal of the American Chemical Society*, 1974, **96**, 5319–5327.

(54) L. G. Dunfield, A. W. Burgess and H. A. Scheraga, "Energy parameters in polypeptides. 8. Empirical potential energy algorithm for the conformational analysis of large molecules", *The Journal of Physical Chemistry*, 1978, **82**, 2609–2616.

(55) B. M. Pettitt and M. Karplus, "Role of electrostatics in the structure, energy and dynamics of biomolecules: a model study of N-methylalanylacetamide", *Journal of the American Chemical Society*, 1985, **107**, 1166–1173.

(56) T. A. Halgren, "Merck molecular force field. I. Basis, form, scope, parameterization, and performance of MMFF94", *Journal of Computational Chemistry*, 1996, **17**, 490–519.

(57) T. A. Halgren, "The representation of van der Waals (vdW) interactions in molecular mechanics force fields: potential form, combination rules, and vdW parameters", *Journal of the American Chemical Society*, 1992, **114**, 7827–7843.

(58) W. J. Mortier, S. K. Ghosh and S. Shankar, "Electronegativity-equalization method for the calculation of atomic charges in molecules", *Journal of the American Chemical Society*, 1986, **108**, 4315–4320.

(59) B. A. Bauer and S. Patel, "Recent applications and developments of charge equilibration force fields for modeling dynamical charges in classical molecular dynamics simulations", *Theoretical Chemistry Accounts*, 2012, **131**.

(60) T. R. Lucas, B. A. Bauer and S. Patel, "Charge equilibration force fields for molecular dynamics simulations of lipids, bilayers, and integral membrane protein systems", *Biochimica et Biophysica Acta (BBA) - Biomembranes*, 2012, **1818**, 318–329.

(61) C. M. Baker, "Polarizable force fields for molecular dynamics simulations of biomolecules", *Wiley Interdisciplinary Reviews: Computational Molecular Science*, 2015, 241–254.

(62) G. Lamoureux and B. Roux, "Modeling induced polarization with classical Drude oscillators: Theory and molecular dynamics simulation algorithm", *The Journal of Chemical Physics*, 2003, **119**, 3025–3039.

(63) G. Lamoureux, A. D. MacKerell and B. Roux, "A simple polarizable model of water based on classical Drude oscillators", *The Journal of Chemical Physics*, 2003, **119**, 5185–5197.

(64) P. E. M. Lopes, J. Huang, J. Shim, Y. Luo, H. Li, B. Roux and A. D. MacKerell, "Polarizable Force Field for Peptides and Proteins Based on the Classical Drude Oscillator", *Journal of Chemical Theory and Computation*, 2013, **9**, 5430–5449.

(65) A. Savelyev and A. D. MacKerell, "All-atom polarizable force field for DNA based on the classical drude oscillator model", *Journal of Computational Chemistry*, 2014, **35**, 1219–1239.

(66) J. Chowdhary, E. Harder, P. E. M. Lopes, L. Huang, A. D. MacKerell and B. Roux, "A Polarizable Force Field of Dipalmitoylphosphatidylcholine Based on the Classical Drude Model for Molecular Dynamics Simulations of Lipids", *The Journal of Physical Chemistry B*, 2013, **117**, 9142–9160.

(67) P. Cieplak, J. Caldwell and P. Kollman, "Molecular mechanical models for organic and biological systems going beyond the atom centered two body additive approximation: aqueous solution free energies of methanol and N-methyl acetamide, nucleic acid base, and amide hydrogen bonding and chloroform/water partition coefficients of the nucleic acid bases", *Journal of Computational Chemistry*, 2001, **22**, 1048–1057.

(68) J. R. Maple, Y. Cao, W. Damm, T. A. Halgren, G. A. Kaminski, L. Y. Zhang and R. A. Friesner, "A Polarizable Force Field and Continuum Solvation Methodology for Modeling of Protein-Ligand Interactions", *Journal of Chemical Theory and Computation*, 2005, **1**, 694–715.

(69) B. Ma, J.-H. Lii and N. L. Allinger, "Molecular polarizabilities and induced dipole moments in molecular mechanics", *Journal of Computational Chemistry*, 2000, **21**, 813–825.

(70) G. A. Kaminski, S. Y. Ponomarev and A. B. Liu, "Polarizable Simulations with Second-Order Interaction Model—Force Field and Software for Fast Polarizable Calculations: Parameters for Small Model Systems and Free Energy Calculations", *Journal of Chemical Theory and Computation*, 2009, **5**, 2935–2943.

(71) S. Y. Ponomarev and G. A. Kaminski, "Polarizable Simulations with Second-Order Interaction Model (POSSIM) Force Field: Developing Parameters for Alanine Peptides and Protein Backbone", *Journal of Chemical Theory and Computation*, 2011, **7**, 1415–1427.

(72) S. Y. Ponomarev, Q. Sa and G. A. Kaminski, "Effects of Lysine Substitution on Stability of Polyalanine α Helix", *Journal of Chemical Theory and Computation*, 2012, **8**, 4691–4706.

(73) X. Li, S. Y. Ponomarev, D. L. Sigalovsky, J. P. Cvitkovic and G. A. Kaminski, "POSSIM: Parameterizing Complete Second-Order Polarizable Force Field for Proteins", *Journal of Chemical Theory and Computation*, 2014, **10**, 4896–4910.

(74) F. Colonna, E. Evleth and J. G. Ángyán, "Critical analysis of electric field modeling: Formamide", *Journal of Computational Chemistry*, 1992, **13**, 1234–1245.

(75) W. Sokalski, D. Keller, R. Ornstein and R. Rein, "Multipole correction of atomic monopole models of molecular charge distribution. I. Peptides", *Journal of Computational Chemistry*, 1993, **14**, 970–976.

(76) P. Ren and J. W. Ponder, "Polarizable Atomic Multipole Water Model for Molecular Mechanics Simulation", *The Journal of Physical Chemistry B*, 2003, **107**, 5933–5947.

(77) J. W. Ponder, C. Wu, P. Ren, V. S. Pande, J. D. Chodera, M. J. Schnieders, I. Haque, D. L. Mobley, D. S. Lambrecht, R. A. DiStasio, M. Head-Gordon, G. N. I. Clark, M. E. Johnson and T. Head-Gordon, "Current Status of the AMOEBA Polarizable Force Field", *The Journal of Physical Chemistry B*, 2010, **114**, 2549–2564.

(78) P. Ren, C. Wu and J. W. Ponder, "Polarizable Atomic Multipole-Based Molecular Mechanics for Organic Molecules", *Journal of Chemical Theory and Computation*, 2011, **7**, 3143–3161.

(79) Y. Shi, Z. Xia, J. Zhang, R. Best, C. Wu, J. W. Ponder and P. Ren, "Polarizable Atomic Multipole-Based AMOEBA Force Field for Proteins", *Journal of Chemical Theory and Computation*, 2013, **9**, 4046–4063.

(80) T. Huber, A. E. Torda and W. F. van Gunsteren, "Local elevation: a method for improving the searching properties of molecular dynamics simulation.", *Journal of Computer Aided Molecular Design*, 1994, **8**, 695–708.

Part I.

SpinningTop

2. Introduction

Molecular conformations are dictated by their dihedral angles. Whereas stretching a bond length or bending a bond angle generally result in only small scale local changes to a molecule's conformation, rotation about a dihedral can have large effects. As such, it is important for a molecular dynamics force field to accurately model the energetics of dihedral rotation. Within the amino acid/peptide/protein environment, two important dihedrals, Φ and Ψ (figure 2.1), are referred to as the backbone dihedrals as they generally define a protein's secondary structure.

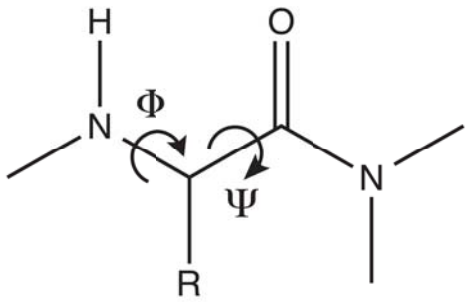


Figure 2.1: Schematic of the Φ and Ψ amino acid backbone dihedrals.

Dihedral energetics are modelled through a combination of the non-bonded parameters and the dihedral terms themselves. Generally, the non-bonded terms provide the impetus for steric hindrance and the general overarching electronic profile of the rotation. Dihedral terms provide smaller, but important, modifications to the electronic profile. Within the GROMOS 54A7 force field, parameterisation generally relies on the reproduction of experimental data obtained for small molecules and the transfer of the suitable parameters to larger systems.[1] However, in the case of the protein backbone dihedral parameters, this general scheme was not followed. Rather, parameters were assigned by chemical intuition and subsequent refinement at the peptide and protein level.[2]

As a consequence of this, the GROMOS 54A7 force field has a number of deficiencies when it comes to peptide and protein backbone angles. Margreitter and Oostenbrink showed that the 54A7 backbone parameters were not suited to describing the characteristics of dipeptides.[2] They failed to reproduce experimental NMR-derived coupling constants and secondary structure propensities from Raman spectroscopy for all twenty canonical dipeptides. Lin and Gunsteren showed that the backbone parameters were unsuitable for simulation of β-peptides.[3] Setz showed across a large range of protein simulations that the Φ and Ψ angle distributions have

poor correlation with distributions derived from structures in the Protein Data Bank (PDB), being particularly poor for the Ψ distributions.[4]

Additionally, from a purely theoretical point of view, there are a number of deficiencies. All amino acids have the same parameters for the backbone dihedrals. Conceptually, this seems ill-advised as the different amino acids should each have different backbone dihedral energy profiles, given their differing side chains. Similar amino acids should have similar dihedral energy profiles, but, for instance, glycine and phenylalanine should have vastly different profiles. Furthermore, the parameters are symmetric* which can be problematic. Only glycine should have a symmetric dihedral rotation profile, all other amino acids have at least C_α side chains, introducing a chiral centre and consequently an asymmetric dihedral rotation profile. Having a symmetric dihedral term on a dihedral that should be asymmetric can result in the dihedral term raising the energy of conformations that should be favoured, and lowering the energy of unfavourable conformations.

This section of work develops a general means to parameterise dihedral terms within molecular dynamics force fields, using high level quantum mechanical (QM) calculations as reference data. Initial tests of the developed fitting method are undertaken by fitting to QM energy profiles, before the method is applied to the case of amino acid backbone Φ and Ψ dihedral parameters within the GROMOS 54A7 force field. Here, it is used to fit the difference between a QM potential energy surface and a molecular dynamics free energy surface in order for the molecular dynamics surface to reproduce the QM potential energy surface when the fitted terms are added.

*i.e. having phase shifts of either 0 or π.

3. Theory and Method Development

This chapter presents the theoretical and mathematical underpinnings of a dihedral parameterisation scheme.

3.1. Derivation of Fitting Method

From equation 1.17 the potential energy due to dihedral terms within a molecule is given by:

$$\sum_{N_\phi} \mathcal{V}^\phi\left(\phi; k_\phi, m, \delta\right) = \sum_{N_\phi} k_\phi \left[1 + \cos\left(m\phi - \delta\right)\right].$$

Thus, any data set consisting of dihedral angle values and their corresponding potential energy values can be used to fit the values of parameters in equation 1.17. For the purposes of molecular dynamics, it is sufficient to limit the multiplicity, m, to the natural numbers, which means the energy profile is always periodic on 2π.

3.1.1. Fitting Symmetric Dihedrals

The simplest case for a fitting procedure is fitting a single, symmetric dihedral term. A symmetric dihedral term is limited to having a phase shift, δ, of either 0 or π, giving it symmetry about $\phi = 0$. This stipulation effectively removes δ from the equations as $\cos(m\phi - 0) \equiv \cos(m\phi)$ and $\cos(m\phi - \pi) \equiv -\cos(m\phi)$. This means that if δ is assumed to be 0 in all cases, any situation in which δ should be π will result in a negative value for k. Removing δ also removes any non-linearity in the function, meaning a fitting procedure can use a linear least squares approach to perform the fit.

For a single symmetric dihedral function, Φ, with $m \in \mathbb{N}$, the potential energy profile is given by:

$$\begin{aligned}
\mathcal{V}\left(\Phi\left(\varphi\right)\right) &= \sum_{m=1}^{n} k_m \left[1 + \cos(m\varphi)\right] \\
&= k_1\left[1 + \cos(\varphi)\right] + k_2\left[1 + \cos(2\varphi)\right] + \ldots + k_n\left[1 + \cos(n\varphi)\right] \\
&= k_1\cos(\varphi) + k_2\cos(2\varphi) + \ldots + k_n\cos(n\varphi) + k_1 + k_2 + \ldots + k_n
\end{aligned}$$

$$= k_1 \cos(\varphi) + k_2 \cos(2\varphi) + \ldots + k_n \cos(n\varphi) + C \tag{3.1}$$

where $C = \sum_{m=1}^{n} k_m$ and φ is the measured angle of the dihedral. Performing a fit to determine the values of the parameters of equation 1.17, N molecular conformations with different φ values are required, giving a series of simultaneous equations to solve:

$$\begin{aligned}
\mathcal{V}(\Phi(\varphi_1)) &= k_1 \cos(\varphi_1) + k_2 \cos(2\varphi_1) + \ldots + k_n \cos(n\varphi_1) + C \\
\mathcal{V}(\Phi(\varphi_2)) &= k_1 \cos(\varphi_2) + k_2 \cos(2\varphi_2) + \ldots + k_n \cos(n\varphi_2) + C \\
&\vdots \qquad\qquad\qquad\qquad \vdots \\
\mathcal{V}(\Phi(\varphi_N)) &= k_1 \cos(\varphi_N) + k_2 \cos(2\varphi_N) + \ldots + k_n \cos(n\varphi_N) + C
\end{aligned} \tag{3.2}$$

The constant value, C, can be removed by subtracting the mean, μ, across all equations on a column by column basis. The equations can then be arranged in matrix form, $\mathbf{A}x = b$:

$$\begin{bmatrix}
\cos(\varphi_1) - \mu_1 & \cos(2\varphi_1) - \mu_2 & \ldots & \cos(n\varphi_1) - \mu_n \\
\cos(\varphi_2) - \mu_1 & \cos(2\varphi_2) - \mu_2 & \ldots & \cos(n\varphi_2) - \mu_n \\
\vdots & \vdots & \ddots & \vdots \\
\cos(\varphi_N) - \mu_1 & \cos(2\varphi_N) - \mu_2 & \ldots & \cos(n\varphi_N) - \mu_n
\end{bmatrix}
\begin{bmatrix} k_1 \\ k_2 \\ \vdots \\ k_n \end{bmatrix}^{\mathrm{T}}
=
\begin{bmatrix}
\mathcal{V}(\Phi(\varphi_1)) - \mu_{\mathcal{V}} \\
\mathcal{V}(\Phi(\varphi_2)) - \mu_{\mathcal{V}} \\
\vdots \\
\mathcal{V}(\Phi(\varphi_N)) - \mu_{\mathcal{V}}
\end{bmatrix} \tag{3.3}$$

As $N \neq n$ in general, the matrix equation cannot be solved exactly, so a linear least squares method must be used. Having $N \gg n$ will result in a more robust fit.

3.1.1.1. Fitting to Multiple Dihedral Terms

Fitting to multiple dihedral terms simultaneously can be desirable in some circumstances. For example, fitting the ϕ and ψ backbone dihedrals of a protein simultaneously could be advantageous due to their propensity to be somewhat coupled. Performing such a fit to multiple dihedrals simultaneously is a straightforward process. Given that the potential energy profiles of individual dihedrals in equation 1.17 are independent of one another, an additional dihedral only adds further parameters to fit, giving rise to the extended matrix form (subtraction of the mean

is implicit):

$$\begin{bmatrix} \cos(\varphi_{1,1}) & \dots & \cos(n\varphi_{1,1}) & \dots & \cos(\varphi_{1,M}) & \dots & \cos(n\varphi_{1,M}) \\ \cos(\varphi_{2,1}) & \dots & \cos(n\varphi_{2,1}) & \dots & \cos(\varphi_{2,M}) & \dots & \cos(n\varphi_{2,M}) \\ \vdots & \ddots & \vdots & \dots & \vdots & \ddots & \vdots \\ \cos(\varphi_{N,1}) & \dots & \cos(n\varphi_{N,1}) & \dots & \cos(\varphi_{N,M}) & \dots & \cos(n\varphi_{N,M}) \end{bmatrix} \begin{bmatrix} k_{1,1} \\ \vdots \\ k_{n,1} \\ \vdots \\ k_{1,M} \\ \vdots \\ k_{n,M} \end{bmatrix}^{\mathrm{T}}$$

$$= \begin{bmatrix} \mathcal{V}(\Phi(\varphi_{1,1}), \dots, \Phi(\varphi_{1,M})) \\ \mathcal{V}(\Phi(\varphi_{2,1}), \dots, \Phi(\varphi_{2,M})) \\ \vdots \\ \mathcal{V}(\Phi(\varphi_{N,1}), \dots, \Phi(\varphi_{N,M})) \end{bmatrix} \tag{3.4}$$

where M is the total number of dihedrals to fit simultaneously, $\varphi_{n,m}$ is the measured angle of the m-th dihedral in the n-th conformation and $k_{m,n}$ is the k value for the m-th multiplicity of the n-th dihedral. Thus it can be seen that any number of dihedrals can be fit simultaneously by merely extending the matrices. Different dihedrals could also be fit with different multiplicity terms by varying m on a dihedral by dihedral basis. Theoretically, other non-dihedral terms could also be included in the fit; they just need to be linear.

3.1.2. Fitting with Phase Shift

Not all dihedrals are symmetric. Thus, not all dihedral energy profiles will be sufficiently reproduced by a Fourier series with δ limited to 0 or π. It therefore becomes apparent that it would be desirable to be able to fit an experimental energy profile with a Fourier series including phase information within the cosine term. First, we must perform some vector algebra.

A cosine can be thought of as tracing the projection of a vector onto the x-axis as it rotates about the origin. A diagrammatic representation of this is shown in figure 3.1. The phase shift, δ, changes the starting position of the rotation. Fitting with variable phase requires linearising the function $k\cos(\varphi - \delta)$ which is non-linear in δ. Thinking of the function as a vector $\mathbf{r}$ in polar space with magnitude r and angle θ, we can define two other vectors, $\mathbf{a}$ and $\mathbf{b}$ with magnitudes a and b and angles α and β respectively, such that:

$$\mathbf{r} = \mathbf{a} + \mathbf{b} \tag{3.5}$$

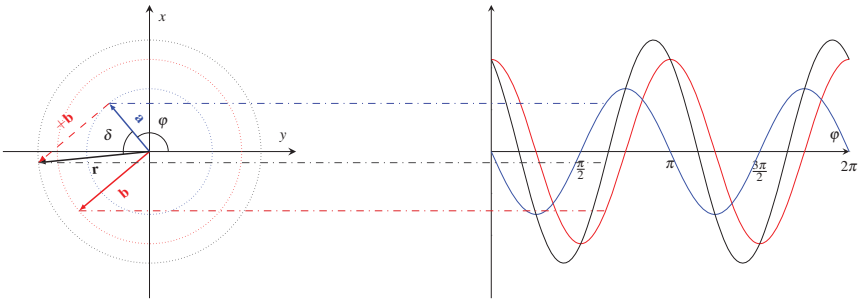


Figure 3.1: Diagramatic representation of addition of vectors and their projection onto the x-axis. The result of the sum of vectors **a** in blue and **b** in red is given by the black vector **r** in the phasor diagram on the left.

From the definition of the dot product:

$$\mathbf{u} \cdot \mathbf{v} = \|\mathbf{u}\| \|\mathbf{v}\| \cos(\gamma) \tag{3.6}$$

we can see that:

$$r^2 = \mathbf{r} \cdot \mathbf{r} \tag{3.7}$$

By combining equations 3.5 to 3.7 we get:

$$\begin{aligned} r^2 &= \mathbf{r} \cdot \mathbf{r} \\ &= (\mathbf{a} + \mathbf{b}) \cdot (\mathbf{a} + \mathbf{b}) \\ &= \mathbf{a} \cdot \mathbf{a} + \mathbf{b} \cdot \mathbf{b} + 2\mathbf{a} \cdot \mathbf{b} \\ &= a^2 + b^2 + 2ab\cos(\beta - \alpha) \\ \therefore \quad r &= \sqrt{a^2 + b^2 + 2ab\cos(\beta - \alpha)} \end{aligned} \tag{3.8}$$

From vector algebra, we get the result that:

$$\tan\theta = \frac{a\sin\alpha + b\sin\beta}{a\cos\alpha + b\cos\beta} \tag{3.9}$$

which can be solved for θ to give:

$$\theta = \arctan2\left(\frac{a\sin\alpha + b\sin\beta}{a\cos\alpha + b\cos\beta}\right) \tag{3.10}$$

where $\arctan2\left(\frac{y}{x}\right)$ implicitly tests the signs of x and y to ensure the correct value of θ between

$-\pi$ and π is returned. Equations 3.8 and 3.10 provide the polar parameters of **r** strictly in terms of the polar parameters of **a** and **b**.

By employing this result, we can fit for phase. Through application of a known phase shift to α and β, for example 0 and $\frac{\pi}{2}$ respectively, the simultaneous equations of equation 3.4 can be used to solve for a and b with δ fixed to 0, then equations 3.8 and 3.10 can be used to determine the values of k and δ.

3.2. Robust Regression

A common problem when fitting data is noise. With respect to the fitting of dihedral rotation energy surfaces presented here, sources of noise can include rotation induced steric clashes causing singularities in the potential which are not well-captured by a Fourier expansion, or incorrect convergence of the underlying calculations. In any case, the noisy data should be accounted for so that any effect on the fitted results is minimised. Generally, least squares regression is used to fit data, but has no inherent ability to manage noisy data. Here, robust regression is used instead.

The residuals of a fit are the difference between the reference data, y_i, and the expected value given by the fit, $y(x_i)$. Least squares regression is a means to minimise the sum of these residuals, S:

$$S = \frac{1}{2}\sum_{i=1}^{n}\left(y(x_i) - y_i\right)^2 \tag{3.11}$$

The square of the residuals is used instead of the absolute residual values because it allows the residuals to be treated as a continuous differentiable quantity. However use of the squares of the residuals does have some drawbacks. In particular, outlying points can have a disproportionate effect on the fit. Take the example of determining the mean of a set of numbers* such as 1.05, 0.98, 0.93 and 12.2. The numbers have been taken from a sample with known mean of 1. The solution of $\mu = 3.79$ is far from the true mean due to the presence of a single outlier.

Robust regression is a means to incorporate robustness into the estimation of a fit to data. This is accomplished by introducing a loss function, $\rho(z)$, which grows slower than linear, to formulate a least squares like problem:

$$S = \frac{1}{2}\sum_{i=1}^{n}\rho\left(\left(y(x_i) - y_i\right)^2\right) \tag{3.12}$$

*Though trivial, this can be thought of as minimising the equation $y = mx + c$ where m is fixed at 0. When $y = c$ is substituted into equation 3.11 and differentiated with respect to c in order to minimise S, the result is $c = \frac{1}{n}\sum_{i=1}^{n} y_i$, which is the mean of the values.

A number of possible loss functions are available, from relatively mild functions such as Huber loss[5] to strongly sub-linear functions such as Cauchy loss.[6] Equation 3.12 collapses to equation 3.11 when the loss function is set to $\rho(z) = z$. For the toy example from earlier, we can apply Cauchy loss, where $\rho(z) = \ln(1+z)$, to obtain a robust estimate for the mean:

$$S = \frac{1}{2}\sum_{i=1}^{n} \ln\left(1 + (c - y_i)^2\right) \tag{3.13}$$

which when differentiated with respect to c gives:

$$\frac{\partial S}{\partial c} = \sum_{i=1}^{n} \frac{c - y_i}{(c - y_i)^2 + 1} = 0 \tag{3.14}$$

Solving for c gives $c \approx 1.017$, which given the parameters of the sample is a much better estimate for the mean than that obtained by minimising equation 3.11.

The form of the derivative of the Cauchy loss function is much more complex than that of the least squares function, even in this simple toy case. As such, in general solving the equations analytically will be difficult, if not impossible. Thus, iterative numerical methods, such as Newton's method[7] are required. Numerical methods require an initial estimate of the result, and due to the possibility of multiple minimum values, the quality of the initial estimate is important. There is a risk of divergence in the iterative process if the initial guess is too far from a root, and the possibility of getting stuck in a local minimum as opposed to the desired global minimum. Divergence can be avoided by taking step sizes in the iterative process such that each step reduces the sum of residuals. Convergence to a local minimum is more difficult to avoid. The simplest means of obtaining an initial estimate of the fit is by performing a least squares fit first, and using the result of that as an input for the robust regression fit. However, outliers affecting the least squares fit can lead to convergence to a local minimum.

To limit the possibility of convergence to a non optimal root, an iterative process is undertaken here for determining the initial guess. A least squares fit to the reference data is performed, and the residuals at each point calculated. If a datum has a residual larger than some cut-off value, in this case 25% of the median absolute reference data value, it is removed from the data set and the least squares fit is performed again. Once either five data points have been removed or all data points have residuals less than the cut-off, the least squares result is used as the initial conditions for the robust regression, and all previously removed data is returned to the data set. The final least squares fit from this iterative process is not used as the final fit as the discarded data may contain important information on the shape of the curve, which will be lost if it is discarded. Figure 3.2 shows the effect of using different fitting methods to fit slightly noisy data containing three outliers. The least squares approach performs poorly, being heavily influenced

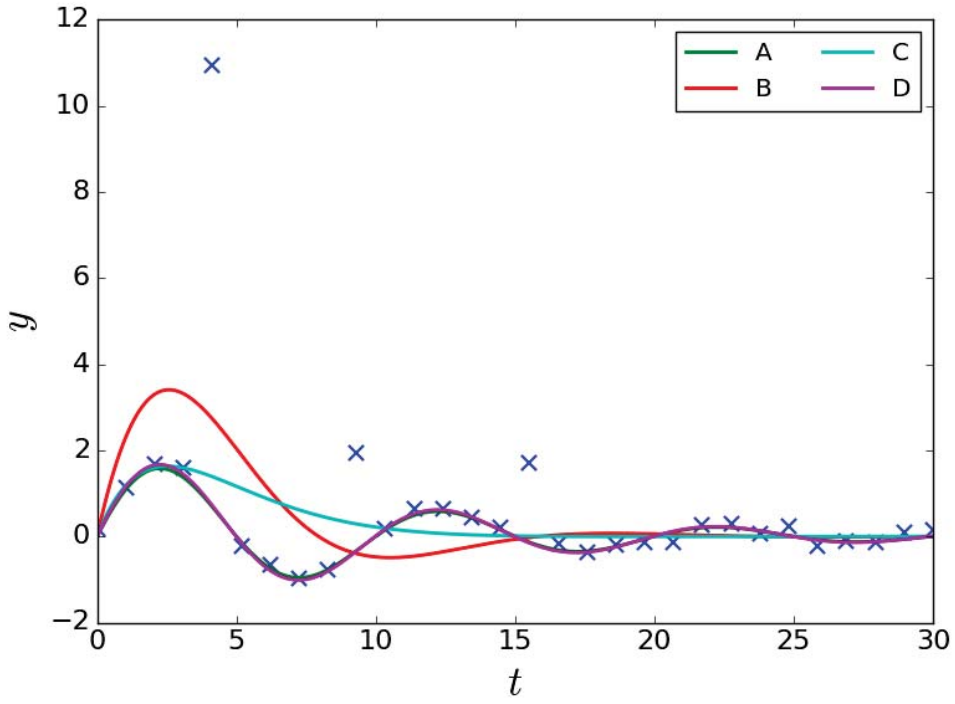


Figure 3.2: Fitting to slightly noisy data (blue crosses) with three outliers generated from the true function (A), $y = 2e^{0.1t}\sin(0.2\pi t)$. Least squares regression (B) performs poorly, as does the Cauchy loss function with least squares initial estimate (C). Cauchy loss with least squares from the trimmed data set as the initial estimate (D) provides a near perfect fit to the true function.

by the outliers. With a default least squares initial estimate, the Cauchy loss function is also heavily influenced by the outliers, though the amplitude of the fit is more in line with the source data. With initial estimate provided by a trimmed least squares fit, the Cauchy loss function nearly perfectly reproduces the real fit. Due to this result, all regression fits to dihedral energy profiles are performed with a Cauchy loss function with trimmed least squares initial estimate.

3.3. Electronic Effects of Distance of Substituents from Dihedral

Dihedral terms capture information about the electronics of a molecule which would otherwise be missing from a force field. Together with the non-bonded terms, they effectively describe how a molecule can sample conformational space. As the number of bonds between a substituent and a dihedral increases, the electronic effect of the substituent on the dihedral will be diminished.

Following this to its logical conclusion, at some distance from the dihedral, a substituent will have negligible affect on the dihedral electronics. This is important because it means that for each dihedral in a molecule, only a fraction of the molecule will require quantum mechanical calculation for a suitable rotational energy profile to be obtained, potentially dramatically reducing the required computational time.

To determine the range at which substituents affect the electronics of a dihedral rotation, a number of dihedral rotational energy profiles were calculated. Unsubstituted 3-ethyl hexane (figure 3.3 with R = H) was chosen as a simple basis molecule. Carbon (R = CH_3), Nitrogen (R = NH_2), Oxygen (R = OH), Sulfur (R = SH), and Chlorine (R = Cl) mono substituted 3-ethyl hexane[†] were chosen to show the effect of different substituents on the dihedral rotation electronics. The rotational energy profile for each substituent was calculated in each of the four R positions, with rotation about the wedged bond in figure 3.3.

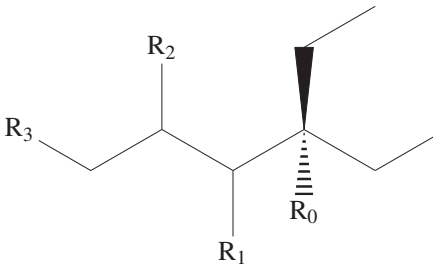


Figure 3.3: 3-ethyl hexane

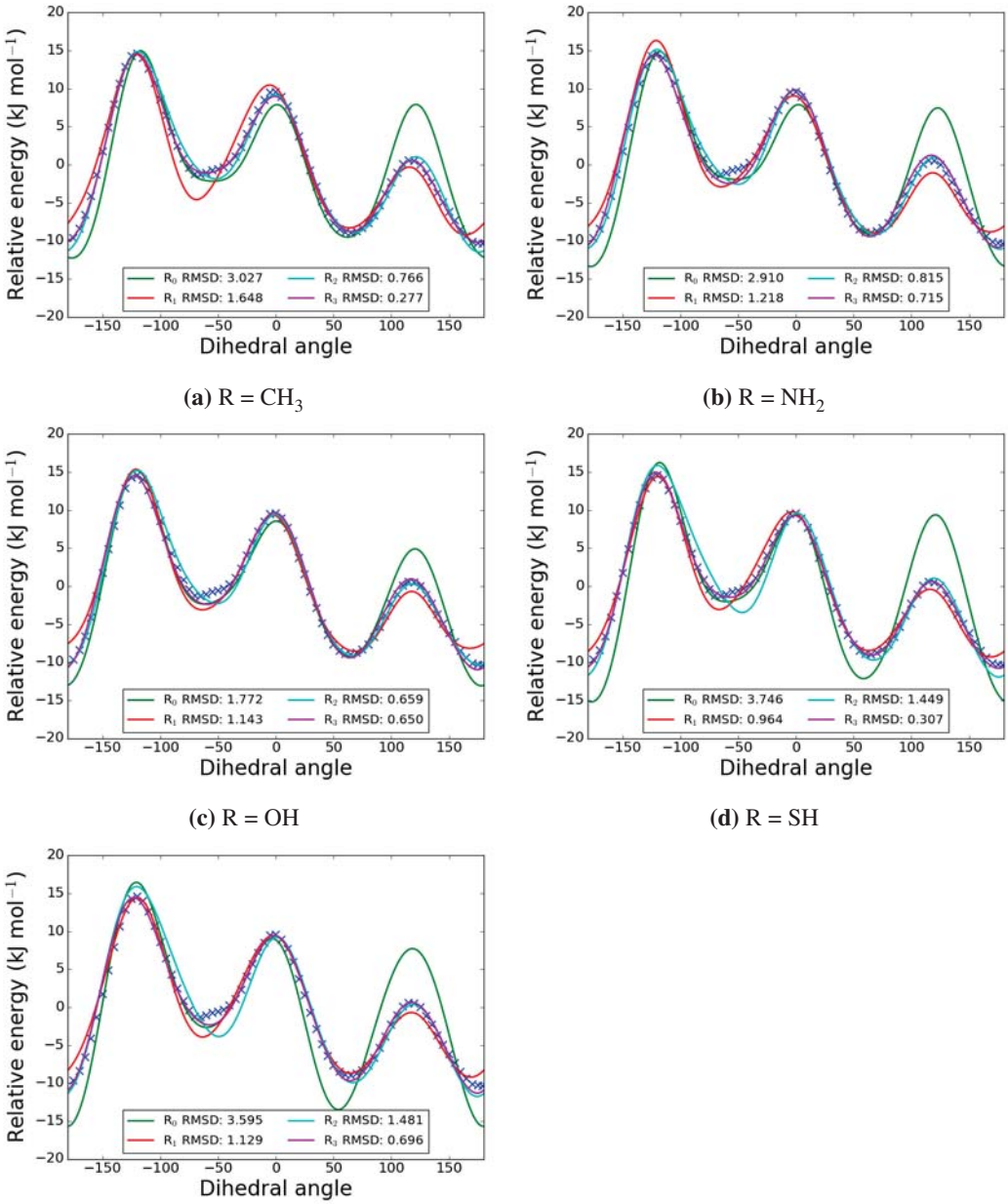
Calculations were performed utilising the B3LYP hybrid functional[8–10] with Grimme's D3 dispersion correction[11] and the Becke-Johnson damping function[12] in PCM[13,14] water. Ahlrichs' Def2-SVP basis set[15] as downloaded from the Basis Set Exchange[16–18] was employed. The software used was GAMESS version 18 August 2016 R1.[19,20] Every 5°, the dihedral was fixed and the rest of the molecule allowed to relax, giving 72 data points between 0° and 360°.

Each dihedral rotational energy profile was fitted for k and δ as detailed in section 3.1.2, with multiplicities from 1 to 6. To determine the effect of the substituent on the dihedral electronics, the Fourier series fit to each substituted 3-ethyl hexane energy profile was compared to the energy profile of the unsubstituted 3-ethyl hexane. The similarity between the sample fit and the unsubstituted energy profile was quantified by calculating the root-mean-square deviation (RMSD) between the two. These results are shown in figure 3.4.

Bulkier substituents tend to have a larger effect when closer to the dihedral of interest than the smaller substituents, showing that there are some steric effects incorporated into the dihedral energy profile. In molecular dynamics force fields, these steric effects should be accounted for by the non bonded parameters, though the dihedral terms will contain some of this information due to the limited number of atom types available.

There is a general trend that as the substituent moves further away from the dihedral, the

[†]one R position taken by the substituent, the other three taken by hydrogen atoms.



(a) R = CH_3

(b) R = NH_2

(c) R = OH

(d) R = SH

(e) R = Cl

Figure 3.4: Fits to substituted dihedral rotational energy profiles compared with the unsubstituted energy profile. The unsubstituted energy profile is given by the blue crosses. RMSD values are reported as the difference between the unsubstituted energy profile and the fit. For reference, the unsubstituted dihedral energy profile has an RMSD of 0.166 kJ mol^{-1} when fit using the same method as the substituted profiles. Source energy profiles for the fits can be found in appendix A.

RMSD between the unsubstituted dihedral energy profile and the fit from the substituted profile decreases, indicating a decreasing electronic effect on the dihedral. Figures 3.4d and 3.4e show an increase in RMSD when going from substitution at R_1 to R_2, which is against the general trend. This can be explained by a poorer fit to the substituted dihedral energy profile caused by discontinuities in the energy profile (see figures A.1 to A.5). The discontinuities arise due to subsequent points along the profile relaxing into different wells for the degrees of freedom away from the fixed dihedral. Performing an energy-directed tree search algorithm (EDTS) search[21] on each fixed conformation can be used to remove these discontinuities (data not shown), however the computational cost is greatly increased and so this was not performed for these systems. Accounting for this discrepancy, results indicate that only the R_0, R_1, and R_2 substituents need to be considered in order to produce energy profiles accurate to within $1\,\mathrm{kJ\,mol^{-1}}$ of the true energy profile. Even including only the R_0 and R_1 substituents would give reasonable results that are acceptable for use in molecular dynamics force field parameterisation.

3.4. Sampling Density Requirements

In order to generate an energy profile for rotation about a dihedral, energies at different dihedral angle values need to be calculated. The sampling density used to obtain such an energy profile is important and affects the quality of any results obtained. A high sampling density will give a better representation of the energy profile, whereas a lower sampling density will be computationally cheaper.

Using data that was produced in section 3.3, a study of the effect of sampling density was undertaken. The source data was sampled at regular 5° intervals between 0° and 360°. Subsets of the source data were taken, corresponding to regular sampling at 10°, 15°, 20° and 30° intervals. Fits were performed in the same manner as section 3.3 and RMSD values calculated between the full set of source data and the fit. Results are given in table 3.1.

Sampling densities of 5°, 10°, 15° and 20° show remarkably similar RMSD values across all data sets. This indicates that a sampling density of 20° will give similar results to sampling at 5° intervals at a quarter of the computational cost which is a significant saving. Sampling at 30° intervals shows a typical increase in RMSD of approximately 50%. This can be attributed mainly to the mathematics of the situation. Sampling a 360° range at 30° intervals gives 12 sampled points. Fitting this data with cosines of multiplicities 1 to 6, and including phase shifts in the fit gives 12 parameters to fit to. As such, an exact result will be determined. With an exact result, the RMSD between the sample subset and the fit will be 0. However, there is no guarantee that data points not part of the fitting subset fall on the fit curve, or that the curve fit is representative of reality. This is clearly shown with the results of chlorine at the R_1 position

Table 3.1: Results of sampling density changes with RMSD from full set of source data for substituted 3-ethyl hexane. RMSD values are in $kJ\,mol^{-1}$.

		5°	10°	15°	20°	30°
Hydrogen		0.166	0.166	0.168	0.174	0.313
Carbon	R_0	0.179	0.179	0.183	0.188	0.248
	R_1	0.143	0.145	0.150	0.149	0.212
	R_2	0.561	0.612	0.584	0.620	0.708
	R_3	0.105	0.109	0.111	0.115	0.256
Nitrogen	R_0	0.177	0.179	0.182	0.188	0.245
	R_1	0.288	0.291	0.296	0.311	0.439
	R_2	0.531	0.545	0.586	0.544	0.966
	R_3	0.114	0.119	0.119	0.119	0.339
Oxygen	R_0	0.190	0.192	0.192	0.202	0.303
	R_1	0.169	0.172	0.178	0.213	0.289
	R_2	0.358	0.366	0.396	0.360	0.551
	R_3	0.116	0.117	0.121	0.128	0.321
Sulfur	R_0	0.138	0.139	0.145	0.156	0.238
	R_1	0.124	0.125	0.126	0.131	0.176
	R_2	0.402	0.411	0.424	0.484	0.634
	R_3	0.149	0.151	0.154	0.156	0.359
Chlorine	R_0	0.234	0.236	0.237	0.245	0.819
	R_1	0.145	0.146	0.147	0.154	7.221
	R_2	0.435	0.434	0.464	0.433	1.069
	R_3	0.172	0.173	0.177	0.175	0.644

which has a very large RMSD value. This can additionally be shown in the results of performing the fit on different subsets of the data sampled at the same regular intervals (see table B.1), where the range of RMSD values obtained is small for all sampling densities except 30°.

4. Method

4.1. Amino Acid Choices

There are twenty* naturally occurring amino acids. A complete re-parameterisation of dihedral terms should consider each amino acid separately. However, the GROMOS force field has a minimalist ethos when it comes to parameters, which is evidenced by all amino acids having the same backbone parameters. Following this ethos, the twenty natural amino acids should be grouped such that each group has their own set of backbone parameters. These groupings can be rationalised by the results of section 3.3, and determined through comparisons of Ramachandran plots. This has the added benefit of reducing the computational cost of parameterisation. Ramachandran plots are visualisations of energetically favourable protein backbone angle (Φ and Ψ) distributions.[25] One of their main uses is in structural validation, for example to validate the Ramachandran plot obtained from an molecular dynamics simulation of a protein against structural data from the PDB. This can be produced either from theoretical models of the energetics, or from experimentally determined backbone angles.

The RCSB PDB is an online database for biological macromolecular structures, in particular proteins.[26,27] There are many thousands of structures, elucidated through various means like x-ray crystallography and NMR. Amino acids with similar backbone dihedral rotation profiles should have similar Ramachandran plots when generated from a large sample size. The PDB provides such a large sample size. All crystal structures for proteins with a resolution less than 2 Å and containing more than 40 residues were obtained from the PDB[26] and the Φ and Ψ angles measured, resulting in over 17 million individual Φ/Ψ angle pairs. Table 4.1 breaks down the counts per amino acid. Measured angles were rounded to the nearest degree on the periodic interval $[-180, 180)$ and normalised Ramachandran plots for each amino acid generated (see figures C.1 to C.20). All pairs of normalised plots were compared point-wise and an RMSD value calculated. The RMSD value gives a measure of similarity between the two plots, with more similar plots having lower RMSDs. With a cutoff value of 15 ppm, i.e. the number of times an amino acid has a particular Φ/Ψ pair per million total Φ/Ψ pairs, the amino acids were arranged into sets where each member of the set had an RMSD less than the cutoff with all other

*Selenocysteine and pyrrolysine are not universal[22–24] and so are not considered here.

Table 4.1: Counts of the occurrences of the individual amino acids in crystal structures obtained from the RCSB PDB.

Amino acid		Count
Glycine	GLY	1,365,056
Alanine	ALA	1,471,589
Serine	SER	972,879
Threonine	THR	982,311
Cysteine	CYS	232,751
Valine	VAL	1,235,406
Leucine	LEU	1,547,302
Isoleucine	ILE	960,267
Methionine	MET	314,492
Proline	PRO	832,359
Phenylalanine	PHE	706,551
Tyrosine	TYR	628,003
Tryptophan	TRP	269,942
Aspartic Acid	ASP	1,011,092
Glutamic Acid	GLU	1,059,659
Asparagine	ASN	735,158
Glutamine	GLN	618,325
Histidine	HIS	421,705
Lysine	LYS	948,480
Arginine	ARG	819,914
Total		17,133,241

members of the set. These results are given in table 4.2.

Generally speaking, the sets obtained follow the sets that would have been produced using only the results of section 3.3. Long side chain amino acids are grouped together, as are side chains with ring systems. Smaller amino acids tend to not be very similar to other amino acids. Organising the amino acids into sets such as this means that only one member of the set needs to go through parameterisation. Such a set member was chosen as the member with the lowest mean RMSD value; effectively the member most like all the others. This criterion gave the set of amino acids to go through backbone dihedral optimisation as glutamine, phenylalanine, valine, aspartic acid, cysteine, serine, alanine, glycine and proline. Because the Φ angle of proline is within a ring, and thus restricting free rotation, only Ψ was optimised for proline.

Table 4.2: RMSD values between normalised Ramachandran plots for all twenty natural amino acids. RMSD values are in ppm. Shaded boxes indicate similar amino acids which can use the same dihedral parameters.

	GLN	ARG	MET	LYS	LEU	GLU	PHE	TYR	HIS	TRP	VAL	ILE	ASP	ASN	CYS	THR	SER	ALA	GLY	PRO
GLN	0.000	9.924	9.744	10.757	10.850	9.808	32.431	34.097	28.481	26.406	34.428	33.518	24.043	33.442	26.796	33.206	32.370	23.576	46.119	67.226
ARG	9.924	0.000	10.086	6.132	11.368	13.894	26.063	27.786	23.163	20.104	29.935	29.576	20.878	30.394	21.735	28.410	28.649	26.124	42.380	63.409
MET	9.744	10.086	0.000	11.733	10.279	12.599	29.777	31.814	27.734	24.717	31.636	30.286	26.250	35.158	26.116	32.041	33.395	25.175	46.747	68.633
LYS	10.757	6.132	11.733	0.000	11.369	14.667	25.775	27.416	22.539	19.386	29.537	28.938	20.043	29.744	21.085	27.709	28.112	26.788	41.683	62.581
LEU	10.850	11.368	10.279	11.369	0.000	13.553	29.141	31.244	27.550	24.295	29.095	26.833	25.819	34.861	25.506	31.265	34.244	26.982	46.653	68.415
GLU	9.808	13.894	12.599	14.667	13.553	0.000	37.555	39.509	34.422	30.799	38.685	36.584	29.657	40.051	32.873	39.241	37.701	19.126	51.696	69.327
PHE	32.431	26.063	29.777	25.775	29.141	37.555	0.000	6.695	13.650	13.008	23.342	25.971	25.582	27.314	16.006	17.142	25.898	49.460	35.155	60.598
TYR	34.097	27.786	31.814	27.416	31.244	39.509	6.695	0.000	13.240	13.912	23.727	27.380	25.660	26.225	15.892	15.907	24.701	51.539	33.835	59.082
HIS	28.481	23.163	27.734	22.539	27.550	34.422	13.650	13.240	0.000	13.957	25.767	29.139	16.855	18.074	11.325	14.385	17.580	46.967	29.732	54.972
TRP	26.406	20.104	24.717	19.386	24.295	30.799	13.008	13.912	13.957	0.000	25.173	26.998	21.097	26.103	14.929	18.969	21.816	42.851	35.485	56.915
VAL	34.428	29.935	31.636	29.537	29.095	38.685	23.342	23.727	25.767	25.173	0.000	10.478	34.408	36.900	24.488	21.507	35.945	49.823	43.721	68.504
ILE	33.518	29.576	30.286	28.938	26.833	36.584	25.971	27.380	29.139	26.998	10.478	0.000	36.425	40.622	27.811	26.808	39.910	46.905	47.698	72.223
ASP	24.043	20.878	26.250	20.043	25.819	29.657	25.582	25.660	16.855	21.097	34.408	36.425	0.000	14.584	18.247	24.319	19.112	41.140	31.363	53.204
ASN	33.442	30.394	35.158	29.744	34.861	40.051	27.314	26.225	18.074	26.103	36.900	40.622	14.584	0.000	20.836	24.273	20.063	51.969	27.476	53.581
CYS	26.796	21.735	26.116	21.085	25.506	32.873	16.006	15.892	11.325	14.929	24.488	27.811	18.247	20.836	0.000	14.972	18.715	44.595	31.360	56.603
THR	33.206	28.410	32.041	27.709	31.265	39.241	17.142	15.907	14.385	18.969	21.507	26.808	24.319	24.273	14.972	0.000	20.630	51.810	31.799	57.624
SER	32.370	28.649	33.395	28.112	34.244	37.701	25.898	24.701	17.580	21.816	35.945	39.910	19.112	20.063	18.715	20.630	0.000	49.455	30.052	46.260
ALA	23.576	26.124	25.175	26.788	26.982	19.126	49.460	51.539	46.967	42.851	49.823	46.905	41.140	51.969	44.595	51.810	49.455	0.000	61.775	76.306
GLY	46.119	42.380	46.747	41.683	46.653	51.696	35.155	33.835	29.732	35.485	43.721	47.698	31.363	27.476	31.360	31.799	30.052	61.775	0.000	55.701
PRO	67.226	63.409	68.633	62.581	68.415	69.327	60.598	59.082	54.972	56.915	68.504	72.223	53.204	53.581	56.603	57.624	46.260	76.306	55.701	0.000

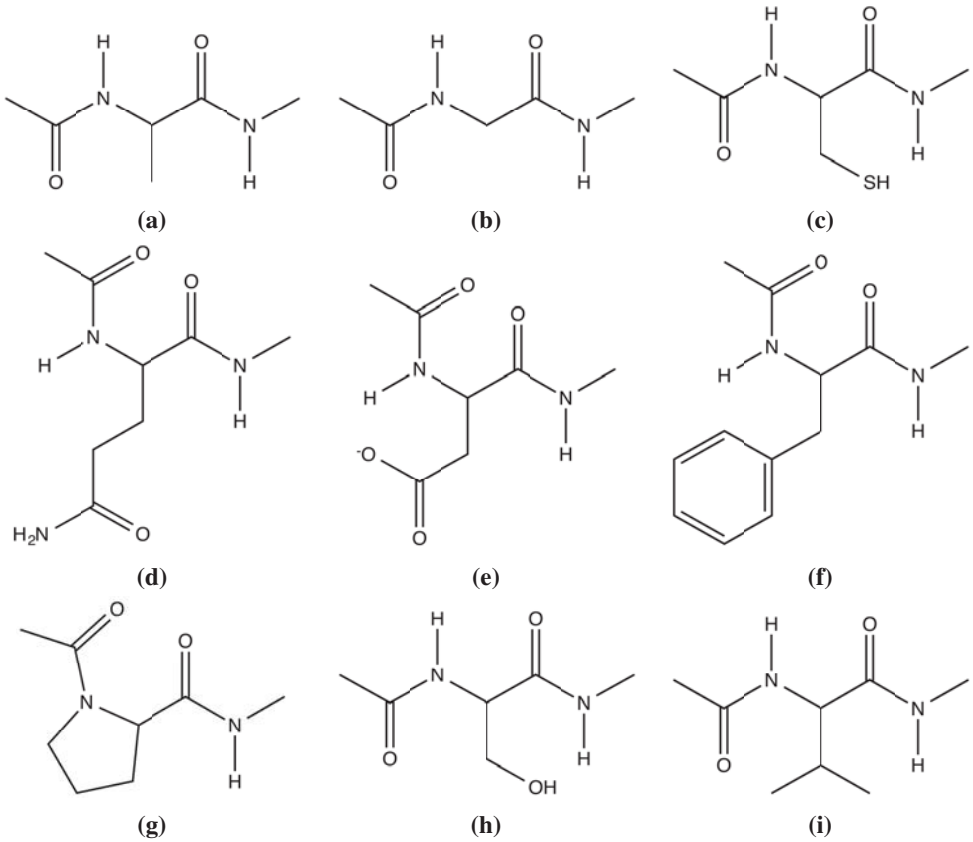


Figure 4.1: Structures of the capped amino acids used for dihedral parametrisation. The amino acids they relate to are: alanine (a), glycine (b), cysteine (c), glutamine (d), aspartic acid (e), phenylalanine (f), proline (g), serine (h), and valine (i).

4.2. Dihedral Parameter Fitting

To limit computational expense, model dipeptides for each of the nine amino acids chosen were produced by capping the N- and C- terminus with an acetyl group and methyl amide group respectively. Structures of each of these model dipeptides are given in figure 4.1.

4.2.1. Quantum Chemical Calculations

A two-dimensional potential energy surface was produced for each of the eight amino acids which allow free rotation about both Φ and Ψ backbone angles. A $15° \times 15°$ diagonal grid was

used, giving up to† 288 points on the potential energy surface. Optimisations were performed at the B3LYP-D3(BJ)/def2-TZVP level of theory,[8–12,15] using a polarisable continuum solvation model (PCM) to simulate bulk solvation effects,[13,14] using GAMESS-US version 18 AUG 2016 (R1).[19,20] The def2-TZVP basis set was downloaded from the Basis Set Exchange.[16–18] Calculations were performed on the NIWA High Performance Computing Facility (HPCF) IBM Power6 cluster, with access provided by NeSI project 00170.

An initial conformational search using the def2-SVP basis set[15] was undertaken to find a low energy starting conformation for the potential energy surface generation. At each gridpoint, one of the dihedrals across the Φ and Ψ bonds was fixed at the given value, and the rest of the molecule allowed to relax. Due to the nature of how the input files were generated, only in the cases of glycine dipeptide and proline dipeptide were the fixed dihedrals defined by the atoms that define the backbone dihedral. In all other cases, the fixed dihedral involved the C_α carbon of the sidechain. Proline dipeptide followed the same scheme, but instead of two dimensional gridpoints, Ψ was fixed at 15° intervals and Φ was unrestrained. This gave 24 regularly spaced data points.

4.2.2. Molecular Dynamics Simulations

Using the same 15° diagonal grid as used to generate the QM potential energy surface, a free energy surface was generated using molecular dynamics simulations without the backbone parameters. At each grid point, the Φ and Ψ dihedral angles were constrained using SHAKE.[28] Each constrained molecule was solvated in a periodic cubic water box in the absence of counter ions. The water boxes were initialised with a 15 Å distance of the solute to the box walls. Prior to the production simulations, the systems were equilibrated from 60 K to 300 K in five discrete steps with a simulation length of 10 ps each. All simulations were carried out at 300 K and a constant volume. A weak thermostat coupling with two baths for the solute and solvent was applied with a coupling constant of 0.1 ps. The SHAKE constraint algorithm was used to maintain the bond distances at the energy minimum, and the Φ and Ψ angles at their desired values. The 54A7 parameter set of the GROMOS force field was used,[1] with the backbone dihedral parameters removed, except for proline dipeptide where only the Ψ parameters were removed. A time step of 2 fs was used, and the energy saved every 100 fs. Interactions within 0.8 nm were calculated at every time step. Interactions up to a distance of 1.4 nm were calculated along with the pairlist update every five steps and kept constant between updates. Long-range interactions were approximated with a reaction field contribution,[29] accounting for a homogeneous

†some confirmations were unable to be minimised due to large initial gradients beyond the limit imposed by GAMESS

medium with relative dielectric constant of 61 beyond the 1.4 nm cut-off. Local elevation was used to gradually build up a bias potential and increase conformational space sampling. Each non-constrained dihedral was divided into thirty-six evenly spaced grid points, with each grid point having its own biasing potential applied. A magnitude of $100\,\mathrm{J\,mol^{-1}}$ was utilised, except in the cases of glycine dipeptide, where the magnitude was $10\,\mathrm{J\,mol^{-1}}$, and the dihedrals within the five-membered ring of proline dipeptide, where the magnitude was $1\,\mathrm{J\,mol^{-1}}$. No bias potential was applied to the internal dihedrals of the aromatic ring in phenylalanine dipeptide. Each simulation was run for 1 ns using a locally modified[‡] version of the GROMOS molecular dynamics engine.[30–32] All energies obtained were used to calculate the partition function, and thus the Helmholtz free energy of the constrained molecule.

4.2.3. Parameter Fitting

The difference between the QM potential energy surface and the free energy surface is used as the potential value for dihedral parameter fitting. To obtain this difference, each surface was interpolated onto a regular 5° rectangular grid using a piecewise cubic, continuously differentiable, and approximately curvature-minimising polynomial surface, as implemented in the cubic interpolation option of SciPy.[33,34] Each interpolated point of the free energy surface was subtracted from the corresponding interpolated point of the QM potential energy surface. Simultaneous fitting of both Φ and Ψ terms was performed on these difference energies. Fitting multiplicity values were limited to one, two, three, and six. Fits included fitting phase, except for glycine dipeptide where the symmetry of the potential energy surface means that fit terms were limited to phase values of 0° or 180°. To follow the minimalism ethos of GROMOS force fields and align with chemical sensibilities, fit parameters were manually modified to have phase values limited to multiples of 30°.

4.3. Experimental Comparisons

Using the dihedral angle population data derived from the PDB and as described in section 4.1, pseudo-energetic Φ/Ψ Ramachandran plots can be produced for each of the amino acids investigated here. Features of the free energy surfaces can then be qualitatively compared with the corresponding features on the experimental energy surfaces. Quantitative comparisons cannot be made as the experimental data comes from the backbone dihedral angles of long sequence peptides and proteins, where as the surfaces calculated here are for capped amino acids. The

[‡]A hard coded 1° tolerance for dihedral constraints was removed and replaced with a per constraint input defined tolerance. A value of 0.0001° was used here.

capped amino acids have greater mobility than amino acids in peptide chains, but the major features of the surfaces should be in similar positions.

4.3.1. Relative Energy Calculation

From equation 1.41 we know the relationship between the probability of a system state being occupied and its energy. Of course, the experimental Φ/Ψ distribution data contains no energy information. However, relative energies can be easily estimated. Assuming two independent system states, with occupation probabilities P_0 and P_1, the energy difference between the two states can be calculated as

$$E_1 - E_0 = \frac{1}{\beta} \log \frac{P_0}{P_1}. \tag{4.1}$$

If the system state with occupation probability P_0 is assumed to be the most populous state, and so given a relative energy of $0\,\mathrm{kJ\,mol^{-1}}$, then the energy of all other states can be calculated relative to this. This method implicitly assumes that the experimental data contains exhaustive sampling of all possible system states, which will never be the case. However, with a large enough sample size, the relative energies obtained in this manner will provide a good indication of the true energies, at least qualitatively. For the results presented here, the experimental Φ/Ψ distribution data was collated onto the same $15^\circ \times 15^\circ$ diagonal grid as used for the QM and free energy data generation prior to the relative energy estimates being generated.

4.3.2. Secondary Structure

Peptide and protein secondary structure is classified based on the hydrogen-bonding pattern of the backbone. In order to recognise structure using hydrogen bonding, a minimum fragment length of four amino acids is generally required. Results presented here consist of single, capped amino acids, which are much too short for hydrogen bonding classification. An alternative method is through the use of Ramachandran plots, where certain areas of the plots indicate backbone conformations associated with secondary structure elements.[25] Using the same classification scheme as Margreitter and Oostenbrink, we are able to distinguish between four secondary structure regions: polyproline-II helix (P_{II}), β-sheet (β), left-handed α-helix (α_L) and, α-helix (α).[2] These regions are represented on all two dimensional surface plots by solid, dotted, dashed and dash-dotted enclosures respectively.

5. Results and Discussion

The simulations used to determine free energies were specifically run with the backbone dihedral parameters removed, meaning that all energies calculated had no contribution from the backbone dihedral terms. As each simulation had the backbone dihedrals constrained, any energy contribution would be constant throughout the simulation, and have no effect on the sampling. Thus, free energy surfaces for any set of backbone dihedral parameters can be cheaply determined by adding the energy contribution at each constrained point on the energy surface prior to determining the partition function. These free energy surfaces can then be visually compared with both QM potential energy surfaces and experimentally derived Φ/Ψ distributions.

5.1. Fitting Outcomes

The Φ and Ψ dihedral potential parameters obtained by fitting the difference between the local elevation-molecular dynamics free energy surface and the QM derived potential energy surfaces for each capped amino acid are discussed individually below. Additionally, pseudo free energy surfaces derived from experimental data obtained from the PDB (section 4.3), free energy surfaces obtained by applying the standard GROMOS 54A7 backbone parameters to local elevation-molecular dynamics simulations run without them, and from the same simulations with the newly fitted backbone dihedral parameters applied, are shown. The fitting process was performed using robust regression with the Cauchy loss function. For reference, the GROMOS 54A7 backbone dihedral parameters are provided in table 5.1. All surface plots are shown with an upper relative energy limit of $20\,\mathrm{kJ\,mol^{-1}}$ for ease of comparison. QM potential energy and raw local elevation molecular dynamics free energy surfaces are provided for each dipeptide in appendix D.

Glycine Fitted parameters and Φ/Ψ energy surface plots for glycine dipeptide are given in figure 5.1. The experimental energy surface (figure 5.1b) shows wells in both the α and α_L regions, centred around $(-60°,-40°)$ and $(80°,10°)$ respectively, and a broad, shallow well around $(-180°,180°)$, spanning 130° in both Φ directions and 80° degrees in both Ψ directions. Coinciding with this large well in the β and P_{II} regions is a general level of sampling in all areas

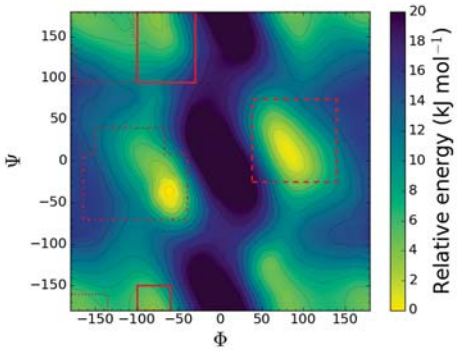
Table 5.1: Reference GROMOS 54A7 backbone dihedral parameters for all amino acids

Φ			Ψ		
k (kJ mol^{-1})	δ (°)	m	k (kJ mol^{-1})	δ (°)	m
2.8	0	3	3.5	180	2
0.7	180	6	0.4	0	6

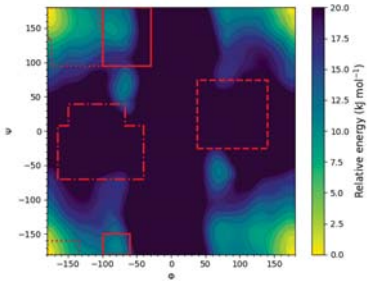
(a) Fitted glycine parameter values

Φ			Ψ		
k (kJ mol^{-1})	δ deg	m	k (kJ mol^{-1})	δ deg	m
3.98	0	2	9.99	180	1
2.99	0	1	9.48	180	2
1.10	0	3	2.57	180	3
0.09	180	6	0.37	0	6

(b) PDB surface



(c) GROMOS 54A7 surface



(d) Fitted surface

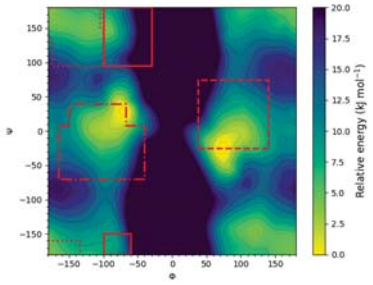


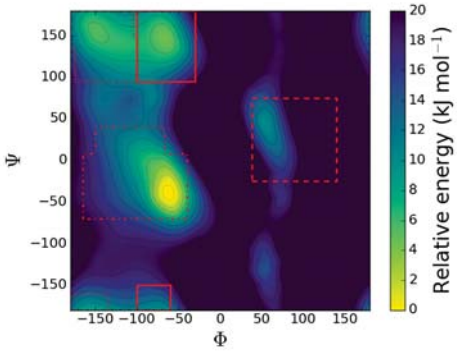
Figure 5.1: (a) Table of terms fitted to the difference between the QM potential energy surface and the local elevation molecular dynamics free energy surface for glycine dipeptide; (b) inverse log plot of the probability distribution derived from RCSB PDB protein crystal structures; (c) the free energy surface calculated with the GROMOS 54A7 backbone parameters; (d) the free energy surface calculated with the backbone parameters given in table a. Secondary structure regions are outlined in red. Contours are placed every 2 kJ mol^{-1}.

except between Φ values of −50° and 50°, afforded by the small size of the glycine amino acid. The free energy surface generated with the GROMOS 54A7 backbone parameters (figure 5.1c) shows clear inconsistencies with the experimentally derived energy surface. There are no wells in either of the α and α_L regions, and a large well around (−180°,180°) in the β region. In contrast, the energy surface generated with the parameters given in figure 5.1a (figure 5.1d)

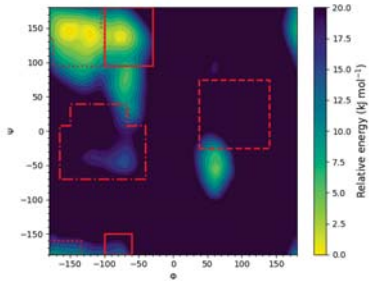
(a) Fitted alanine parameter values

Φ			Ψ		
k (kJ mol^{-1})	δ deg	m	k (kJ mol^{-1})	δ deg	m
8.39	0	1	5.61	-150	2
5.63	30	2	5.45	150	1
3.02	90	3	2.40	-150	3
0.35	0	6	0.09	-150	6

(b) PDB surface



(c) GROMOS 54A7 surface



(d) Fitted surface

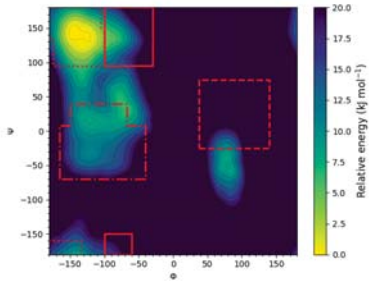


Figure 5.2: (a) Table of terms fitted to the difference between the QM potential energy surface and the local elevation molecular dynamics free energy surface for alanine dipeptide; (b) inverse log plot of the probability distribution derived from RCSB PDB protein crystal structures; (c) the free energy surface calculated with the GROMOS 54A7 backbone parameters; (d) the free energy surface calculated with the backbone parameters given in table a. Secondary structure regions are outlined in red. Contours are placed every 2 kJ mol^{-1}.

shows much better, though not entirely perfect, agreement with the experimental energy surface. The α and α_L regions have wells centred around (−70°,30°) and (70°,30°) respectively. Though these wells are on the edge of their respective regions, they are asymmetric and the shallow gradient side of the well points into the region. There are also similar shallow wells around (−180°,180°), and a general level of sampling in all areas except between Φ values of −50° and 50°, in agreement with the experimental energy surface. Overall, the parameters fit here should provide a significant improvement to the sampling obtained by glycine amino acid residues within peptides or proteins.

Alanine Fitted parameters and Φ/Ψ energy surface plots for alanine dipeptide are given in figure 5.2. As is the case for all amino acid dipeptides with side chain groups, the experimental

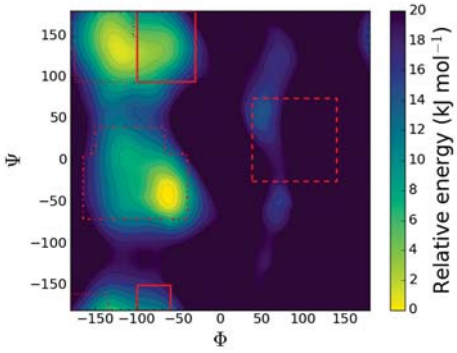
energy surface for alanine dipeptide (figure 5.2b) shows significant sampling only in Φ ranges of between $-50°$ and $-180°$. There is a deep well centred around ($-60°$,$-40°$) in the α region, and shallower wells in the β and P_{II} regions centred around ($-145°$,$155°$) and ($-70°$,$145°$) respectively. A low level of sampling is evident in the area between the α and β/P_{II} regions, as well as in the α_L region. Again, the GROMOS 54A7 free energy surface (figure 5.2c) shows obvious discrepancies with the experimental energy surface, though not as major as with glycine dipeptide. Wells are present in the β and P_{II} regions, centred around ($-75°$,$140°$) and ($-140°$,$150°$) respectively, which coincides with the well positions in the experimental energy surface. However, the GROMOS 54A7 wells are much deeper than the experimental ones. Further, though there is a well in the α region centred around ($-70°$,$-40°$), it is much shallower than the corresponding well in the experimental surface. There is also close to no further sampling within the rest of the α region, which is in stark contrast broad sampling seen in the experimental energy surface. Two further wells of interest are located around ($-70°$,$-80°$) between the P_{II} and α regions, and a deep well around ($60°$,$-50°$), outside the α_L region. Neither of these wells are present in the experimental energy surface. The free energy surface produced using the parameters in figure 5.2a (figure 5.2d) offers slight improvements over the GROMOS 54A7 energy surface. The well at ($75°$,$-50°$) outside of the α_L region has become shallower, and the well between the P_{II} and α regions has shifted closer to the edge of the α region at ($-70°$,$30°$). As a consequence of this, there is improved general sampling within the α region, though there is also no significant well within the region. The well within the P_{II} around ($-80°$,$135°$) has diminished relative to the corresponding well in the GROMOS 54A7 free energy surface, though the well in the β region around ($-150°$,$145°$) is relatively unchanged.

Valine Fitted parameters and Φ/Ψ energy surface plots for valine dipeptide are given in figure 5.3. The experimental energy surface (figure 5.3c) shows two major wells, one centred around ($-120°$,$135°$) in the β region and one centred around ($-60°$,$-40°$) in the α region. These wells also have broad sampling filling out the remainder of their regions and in the space between the β/P_{II} and α regions. There is also some sampling within the α_L region, though it is very minor. In contrast, the GROMOS 54A7 free energy surface (figure 5.3c) has a deep well in both of the β and P_{II} regions, centred around ($-115°$,$140°$) and ($-70°$,$125°$) respectively. Again, just like the alanine dipeptide GROMOS 54A7 free energy surface, there is minimal sampling within the α region, and a slight well outside the α_L region centred around ($60°$,$-60°$). With the backbone dihedral parameters given in figure 5.3a, the free energy surface produced (figure 5.3d) is missing the well just outside the α_L region, which is in line with the experimental energy surface. There is also a well in the β region centred around ($-120°$,$140°$), corresponding to the well in the same location in the experimental energy surface, though it is slightly deeper.

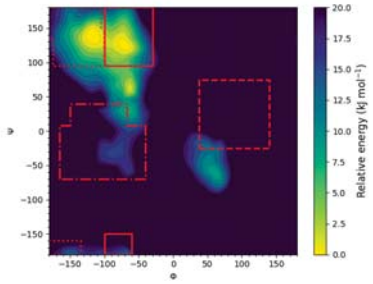
(a) Fitted valine parameter values

Φ			Ψ		
k (kJ mol^{-1})	δ deg	m	k (kJ mol^{-1})	δ deg	m
9.65	0	1	10.74	-150	2
3.14	30	2	5.85	150	1
0.60	60	3	2.55	-120	3
0.53	150	6	0.47	-150	6

(b) PDB surface



(c) GROMOS 54A7 surface



(d) Fitted surface

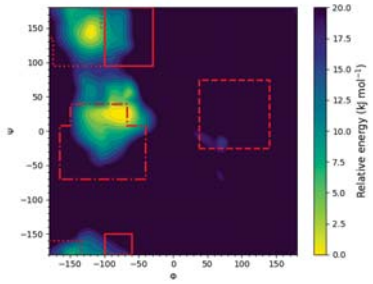


Figure 5.3: (a) Table of terms fitted to the difference between the QM potential energy surface and the local elevation molecular dynamics free energy surface for valine dipeptide; (b) inverse log plot of the probability distribution derived from RCSB PDB protein crystal structures; (c) the free energy surface calculated with the GROMOS 54A7 backbone parameters; (d) the free energy surface calculated with the backbone parameters given in table a. Secondary structure regions are outlined in red. Contours are placed every 2 kJ mol^{-1}.

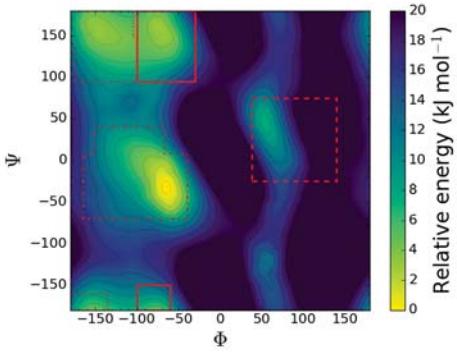
The well in the α region centred around $(-80°,20°)$ causes increased sampling in the region relative to the GROMOS 54A7 free energy surface, but still less than in the experimental free energy surface. A particular note about the fitted free energy surface is that the sampling occupies a noticeably narrower area relative to the experimental energy surface, indicating that the force constants are potentially too large.

Serine Fitted parameters and Φ/Ψ energy surface plots for serine dipeptide are given in figure 5.4. The experimentally derived energy surface (figure 5.4b) is very similar to that seen for alanine (figure 5.2b), with the most noticeable difference being a slightly deeper well in the α_L region. Neither the free energy surface produced with the GROMOS 54A7 backbone dihedral parameters (figure 5.4c), nor the free energy surface produced with the fitted parameters in

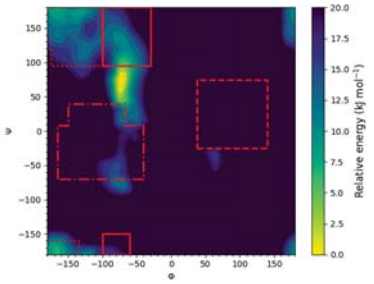
(a) Fitted serine parameter values

Φ			Ψ		
k (kJ mol^{-1})	δ deg	m	k (kJ mol^{-1})	δ deg	m
6.48	30	1	9.81	-150	1
3.10	-30	3	5.38	-150	2
0.92	0	2	2.47	-150	3
0.60	-60	6	0.70	0	6

(b) PDB surface



(c) GROMOS 54A7 surface



(d) Fitted surface

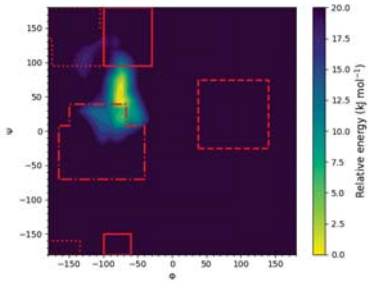


Figure 5.4: (a) Table of terms fitted to the difference between the QM potential energy surface and the local elevation molecular dynamics free energy surface for serine dipeptide; (b) inverse log plot of the probability distribution derived from RCSB PDB protein crystal structures; (c) the free energy surface calculated with the GROMOS 54A7 backbone parameters; (d) the free energy surface calculated with the backbone parameters given in table a. Secondary structure regions are outlined in red. Contours are placed every 2 kJ mol^{-1}.

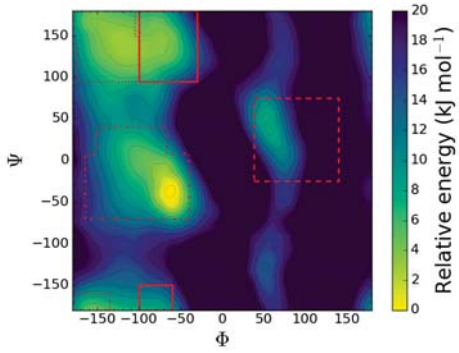
figure 5.4a (figure 5.4d) even come close to reproducing the experimental energy surface. The GROMOS 54A7 free energy surface is the better of the two as it has some level of sampling in the β and P_{II} regions which is not present in the fitted free energy surface. Such a poor outcome of the fitting process indicates that one or both of the QM potential energy surface or the raw free energy surface generated are not as reliable as they should be.

Cysteine Fitted parameters and Φ/Ψ energy surface plots for cysteine dipeptide are given in figure 5.5. Like the serine experimental energy surface, the cysteine experimental energy surface (figure 5.5b) has a deep well in the α region centred around $(-60°,-40°)$, a large shallow well in the α_L region centred around (50°,50°), and a very broad well straddling the β and P_{II} regions. The GROMOS 54A7 free energy surface (figure 5.5c) shows a broad sampling area straddling

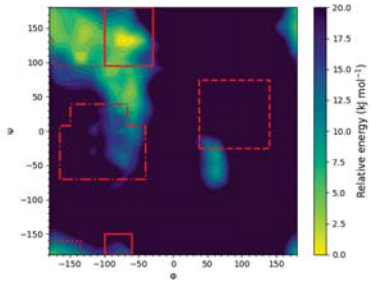
(a) Fitted cysteine parameter values

Φ			Ψ		
k (kJ mol^{-1})	δ deg	m	k (kJ mol^{-1})	δ deg	m
9.62	30	1	9.42	180	2
3.21	90	2	7.12	180	1
2.09	0	3	0.43	180	6
0.24	90	6	0.32	120	3

(b) PDB surface



(c) GROMOS 54A7 surface



(d) Fitted surface

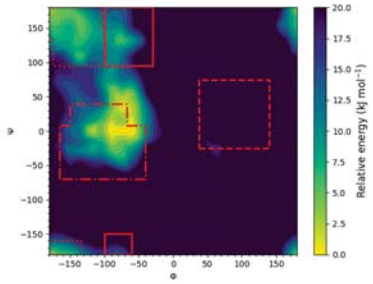


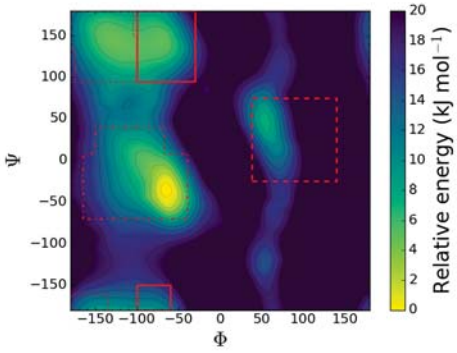
Figure 5.5: (a) Table of terms fitted to the difference between the QM potential energy surface and the local elevation molecular dynamics free energy surface for cysteine dipeptide; (b) inverse log plot of the probability distribution derived from RCSB PDB protein crystal structures; (c) the free energy surface calculated with the GROMOS 54A7 backbone parameters; (d) the free energy surface calculated with the backbone parameters given in table a. Secondary structure regions are outlined in red. Contours are placed every 2 kJ mol^{-1}.

the β and P_{II} regions, similar to that in the experimental energy surface. However, instead of a mostly smooth surface, there are a number of undulations across the surface. These undulations are visible across both of the free energy surfaces, which indicates that some of the simulations for determining free energies did not converge sufficiently within the 1 ns time they were run for. The GROMOS 54A7 surface also shows no well in either the α or α_L regions, though there is a shallow well just below the α_L region centred around (60°,−40°). The free energy surface generated with the parameters in figure 5.5a (figure 5.5d) shows general broad sampling across the β/P_{II} regions, as well as improved sampling in the α region. As mentioned previously, there are a large number of undulations in the surface which would require extending free energy simulations in order to smooth out. There is also no sampling within the α_L region, and the well below the region seen in the GROMOS 54A7 free energy surface is also all but gone.

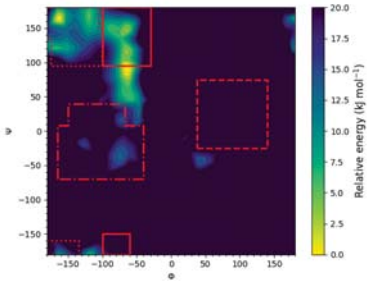
(a) Fitted glutamine parameter values

Φ			Ψ		
k (kJ mol^{-1})	δ deg	m	k (kJ mol^{-1})	δ deg	m
8.86	-30	1	11.97	180	1
6.03	30	2	3.90	-150	3
2.49	60	3	1.26	90	2
1.14	-90	6	1.20	-90	6

(b) PDB surface



(c) GROMOS 54A7 surface



(d) Fitted surface

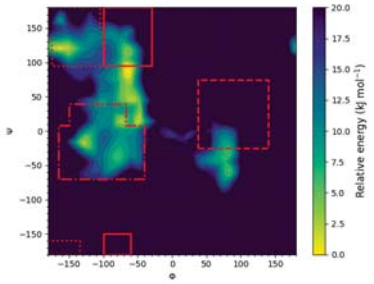


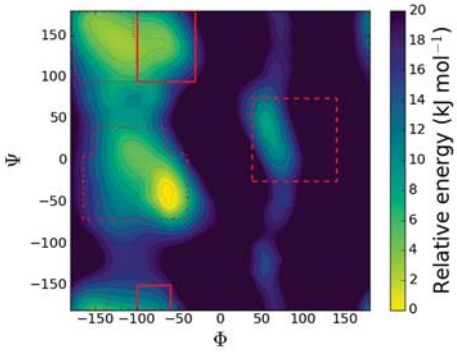
Figure 5.6: (a) Table of terms fitted to the difference between the QM potential energy surface and the local elevation molecular dynamics free energy surface for glutamine dipeptide; (b) inverse log plot of the probability distribution derived from RCSB PDB protein crystal structures; (c) the free energy surface calculated with the GROMOS 54A7 backbone parameters; (d) the free energy surface calculated with the backbone parameters given in table a. Secondary structure regions are outlined in red. Contours are placed every 2 kJ mol^{-1}.

Glutamine Fitted parameters and Φ/Ψ energy surface plots for glutamine dipeptide are given in figure 5.6. On the experimental energy surface (figure 5.6b), there is a well centred around $(-65°,-35°)$ with a broad tail allowing sampling of the α region, and a much shallower well in the α_L centred around (55°,50°). There is also a broad well spanning the β and P_{II} regions. Not much information can be gleaned from the GROMOS 54A7 (figure 5.6c) free energy surface, nor from the free energy surface generated with the parameters in figure 5.6a (figure 5.6d). This is because they show signs of interpolation artefacts, i.e. portions of the surface with very square like characteristics, which indicates the underlying free energy simulations did not converge within the simulation time period. With deficiencies in the underlying energy surface, the fitting process will be unreliable, and so any fitted terms are likely incorrect. However, the general result that the fitted free energy surface has better all round sampling than the GROMOS 54A7

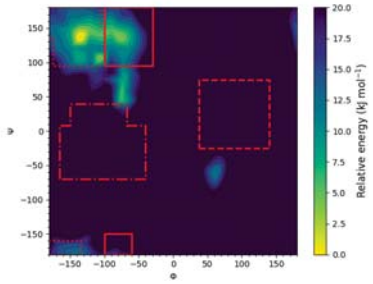
(a) Fitted phenylalanine parameter values

Φ			Ψ		
k (kJ mol^{-1})	δ deg	m	k (kJ mol^{-1})	δ deg	m
11.10	30	1	10.45	150	1
3.74	-30	3	10.33	180	2
3.43	120	2	2.89	-90	3
0.21	180	6	0.76	30	6

(b) PDB surface



(c) GROMOS 54A7 surface



(d) Fitted surface

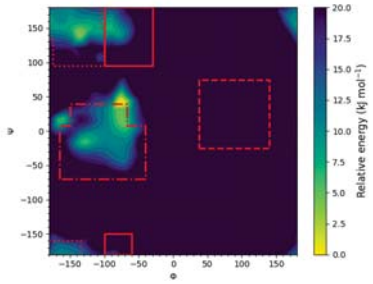


Figure 5.7: (a) Table of terms fitted to the difference between the QM potential energy surface and the local elevation molecular dynamics free energy surface for phenylalanine dipeptide; (b) inverse log plot of the probability distribution derived from RCSB PDB protein crystal structures; (c) the free energy surface calculated with the GROMOS 54A7 backbone parameters; (d) the free energy surface calculated with the backbone parameters given in table a. Secondary structure regions are outlined in red. Contours are placed every 2 kJ mol^{-1}.

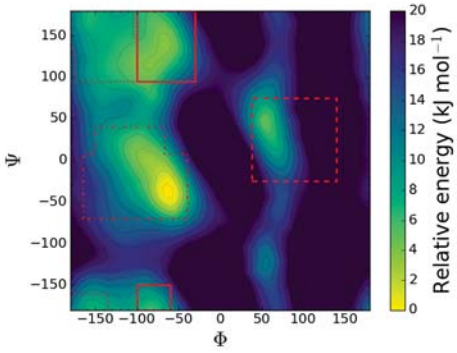
free energy surface.

Phenylalanine Fitted parameters and Φ/Ψ energy surface plots for phenylalanine dipeptide are given in figure 5.7. In the same manner as the other longer side chain amino acids, the experimental energy surface (figure 5.7b) shows a deep well in the α region, centred around (−60°,−40°), a shallow well in the α_L region centred around (50°,40°), and a broad area straddling the β and P_{II} regions, with a reasonable level of sampling between those regions and the α region. The GROMOS 54A7 free energy surface (figure 5.7c) shows only the broad, but undulating, well spanning the β and P_{II} regions. In contrast, the free energy surface generated with the parameters in figure 5.7a (figure 5.7d) shows both the broad well spanning the β and P_{II} regions, and the deep well in the α region, slightly shifted centring around (−75°,40°). This

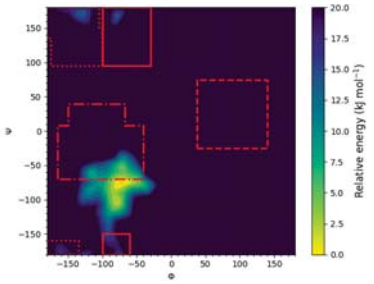
(a) Fitted aspartic acid parameter values

Φ			Ψ		
k ($\mathrm{kJ\,mol^{-1}}$)	δ deg	m	k ($\mathrm{kJ\,mol^{-1}}$)	δ deg	m
16.07	0	1	17.26	-150	1
9.33	30	2	4.86	-60	2
0.96	180	3	3.08	-150	6
0.50	-30	6	2.66	180	3

(b) PDB surface



(c) GROMOS 54A7 surface



(d) Fitted surface

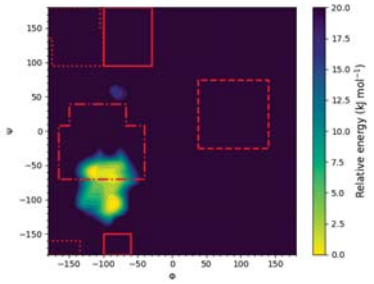


Figure 5.8: (a) Table of terms fitted to the difference between the QM potential energy surface and the local elevation molecular dynamics free energy surface for aspartic acid dipeptide; (b) inverse log plot of the probability distribution derived from RCSB PDB protein crystal structures; (c) the free energy surface calculated with the GROMOS 54A7 backbone parameters; (d) the free energy surface calculated with the backbone parameters given in table a. Secondary structure regions are outlined in red. Contours are placed every $2\,\mathrm{kJ\,mol^{-1}}$.

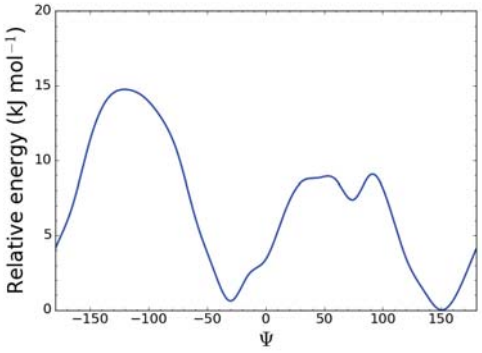
is overall in better agreement with the experimental surface, though the wells are also narrower than the GROMOS 54A7 wells.

Aspartic Acid Fitted parameters and Φ/Ψ energy surface plots for aspartic acid dipeptide are given in figure 5.8. Aspartic acid is different to all other amino acids investigated here in that it is generally found in a deprotonated, and thus negatively charged, state. The overall negative charge on the dipeptide molecule resulted in a QM potential energy surface (see figure D.2) showing large local oscillation behaviour, particularly in the 0° to 180° Φ range. The difference between the QM surface and the raw free energy surface consequently also had large local oscillation behaviour, giving the fit parameters in figure 5.8a with far larger force constants than seen in any of the other dipeptide fits, particularly in relation to the multiplicity one terms. Whereas

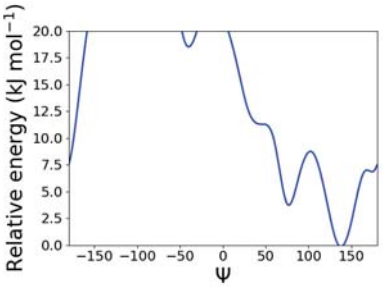
(a) Fitted proline parameter values

Ψ		
k (kJ mol^{-1})	δ deg	m
9.08	150	1
7.24	180	2
2.52	120	3
1.26	90	6

(b) PDB profile



(c) GROMOS 54A7 profile



(d) Fitted profile

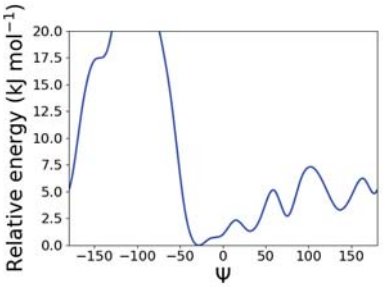


Figure 5.9: (a) Table of terms fitted to the difference between the QM potential energy surface and the local elevation molecular dynamics free energy surface for proline dipeptide; (b) inverse log plot of the probability distribution derived from RCSB PDB protein crystal structures; (c) the free energy surface calculated with the GROMOS 54A7 backbone parameters; (d) the free energy surface calculated with the backbone parameters given in table a.

the experimental energy surface (figure 5.8b) shows broad sampling across the α, β and P_{II} regions, both the GROMOS 54A7 free energy surface (figure 5.8c) and the free energy surface generated with the fitted parameters (figure 5.8d) show sampling only on the lower edge and below the α region.

Proline Fitted Ψ parameters and Ψ energy profile plots for proline dipeptide are given in figure 5.9. Like aspartic acid, proline dipeptide differs from all other amino acids investigated here as the five-membered ring of the side chain inhibits free rotation about the Φ dihedral angle. As such, only the Ψ dihedral angle was investigated here. The experimental energy profile (figure 5.9b) shows two main wells around $-30°$ and 150°, corresponding to the α and β/P_{II} regions respectively. The GROMOS 54A7 free energy profile (figure 5.9c) shows similar wells, though there is a vast energy difference between them with the well around $-30°$ having a

markedly higher relative energy. This leads to a broader low sampling zone between $-180°$ and $0°$ than in the experimental profile. A minor well around $75°$ in the experimental energy profile also becomes more pronounced. On the other hand, the free energy profile generated using the fitted parameters in figure 5.9a (figure 5.9d) has a narrower low sampling region below $0°$, but a broader sampling region above due to the large number of wells of relatively similar energies.

5.2. General Comments

Perhaps the most notable aspect of the fit parameters shown here is that there are generally one or two terms with large force constants dominating over all the other terms. This appears to be due to the large peak at (0°,180°) present in all the QM potential energy surfaces. In this conformation, the two carbonyl oxygen atoms of the backbone are aligned with one another, and so will have intense repulsive interactions. Conceivably, a slight perturbation of some of the degrees of freedom in the QM optimised structure could be enough to allow further relaxation of the oxygen–oxygen interaction, which will lower the relative energy at this point. With a lower peak, there will not be as large a difference between the QM potential energy surface and the calculated free energy surface, which should result in fitted terms having lower force constants. An alternative means to improve the QM potential energy surface would be, at each fixed point perform a conformational search involving the other molecular degrees of freedom at a lower level of theory, and then optimising the minimum conformation using a higher theory level.

A number of the free energy surfaces show signs of artefacts arising from the sampling method used to generate them, especially those for the larger amino acids. This indicates that there was potential non-convergence of the free energy at certain points on the surface. This could be eliminated through increasing the simulation time over which sampling is under taken. Additionally, multiple simulations using the same process but beginning from different conformations could also be used to alleviate the problem.

5.3. Conclusions

Overall, this method for dihedral parameter optimisation shows much promise. Small, simple amino acids, such as glycine and alanine, show much improved free energy surfaces when compared to experimentally derived surfaces then the current GROMOS 54A7 free energy surfaces do. With larger amino acids, the results are not as obvious, primarily due to a lack of convergence in the free energy simulations. Of course, any proposed backbone parameters will need to be tested through a wide variety of protein simulations, which will be the subject of future work.

Bibliography

(1) N. Schmid, A. Eichenberger, A. Choutko, S. Riniker, M. Winger, A. Mark and W. van Gunsteren, "Definition and testing of the GROMOS force-field versions 54A7 and 54B7", *European Biophysics Journal*, 2011, **40**, 843–856.

(2) C. Margreitter and C. Oostenbrink, "Optimization of Protein Backbone Dihedral Angles by Means of Hamiltonian Reweighting", *Journal of Chemical Information and Modeling*, 2016, **56**, 1823–1834.

(3) Z. Lin and W. F. van Gunsteren, "Refinement of the application of the GROMOS 54A7 force field to beta-peptides", *Journal of Computational Chemistry*, 2013, **34**, 2796–2805.

(4) M. Setz, "Large Scale Validation of GROMOS Force Fields", personal correspondence, 2016.

(5) P. J. Huber, "Robust Estimation of a Location Parameter", *The Annals of Mathematical Statistics*, 1964, **35**, 73–101.

(6) I. Mizera and C. H. Müller, "Breakdown points of Cauchy regression-scale estimators", *Statistics & Probability Letters*, 2002, **57**, 79–89.

(7) T. Simpson, *Essays on Mathematics*, London, 1740, p. 81.

(8) A. D. Becke, "Density-functional thermochemistry. III. The role of exact exchange", *The Journal of Chemical Physics*, 1993, **98**, 5648–5652.

(9) P. J. Stephens, F. J. Devlin, C. F. Chabalowski and M. J. Frisch, "Ab Initio Calculation of Vibrational Absorption and Circular Dichroism Spectra Using Density Functional Force Fields", *The Journal of Physical Chemistry*, 1994, **98**, 11623–11627.

(10) R. H. Hertwig and W. Koch, "On the parameterization of the local correlation functional. What is Becke-3-LYP?", *Chemical Physics Letters*, 1997, **268**, 345–351.

(11) S. Grimme, J. Antony, S. Ehrlich and H. Krieg, "A consistent and accurate ab initio parametrization of density functional dispersion correction (DFT-D) for the 94 elements H-Pu", *The Journal of Chemical Physics*, 2010, **132**, 154104:1–19.

(12) S. Grimme, S. Ehrlich and L. Goerigk, "Effect of the damping function in dispersion corrected density functional theory", *Journal of Computational Chemistry*, 2011, **32**, 1456–1465.

(13) H. Li, C. S. Pomelli and J. H. Jensen, "Continuum solvation of large molecules described by QM/MM: a semi-iterative implementation of the PCM/EFP interface", *Theoretical Chemistry Accounts*, 2003, **109**, 71–84.

(14) H. Li and J. H. Jensen, "Improving the efficiency and convergence of geometry optimization with the polarizable continuum model: New energy gradients and molecular surface tessellation", *Journal of Computational Chemistry*, 2004, **25**, 1449–1462.

(15) F. Weigend and R. Ahlrichs, "Balanced basis sets of split valence, triple zeta valence and quadruple zeta valence quality for H to Rn: Design and assessment of accuracy", *Physical Chemistry Chemical Physics*, 2005, **7**, 3297–3305.

(16) *Basis Set Exchange*, accessed November 2016, November 7 2016, `https://bse.pnl.gov/bse/portal`.

(17) D. Feller, "The role of databases in support of computational chemistry calculations", *Journal of Computational Chemistry*, 1996, **17**, 1571–1586.

(18) K. L. Schuchardt, B. T. Didier, T. Elsethagen, L. Sun, V. Gurumoorthi, J. Chase, J. Li and T. L. Windus, "Basis Set Exchange: A Community Database for Computational Sciences", *Journal of Chemical Information and Modeling*, 2007, **47**, 1045–1052.

(19) M. W. Schmidt, K. K. Baldridge, J. A. Boatz, S. T. Elbert, M. S. Gordon, J. H. Jensen, S. Koseki, N. Matsunaga, K. A. Nguyen, S. Su, T. L. Windus, M. Dupuis and J. A. Montgomery, "General atomic and molecular electronic structure system", *Journal of Computational Chemistry*, 1993, **14**, 1347–1363.

(20) M. S. Gordon and M. W. Schmidt, in, ed. C. E. Dykstra, G. Frenking, K. S. Kim and G. E. Scuseria, Elsevier, 2005, ch. Advances in electronic structure theory: GAMESS a decade later, pp. 1167–1189.

(21) E. I. Izgorodina, C. Yeh Lin and M. L. Coote, "Energy-directed tree search: an efficient systematic algorithm for finding the lowest energy conformation of molecules", *Physical Chemistry Chemical Physics*, 2007, **9**, 2507–2516.

(22) R. Longtin, "A Forgotten Debate: Is Selenocysteine the 21st Amino Acid?", *Journal of the National Cancer Institute*, 2004, **96**, 504–505.

(23) B. Hao, W. Gong, T. K. Ferguson, C. M. James, J. A. Krzycki and M. K. Chan, "A New UAG-Encoded Residue in the Structure of a Methanogen Methyltransferase", *Science*, 2002, **296**, 1462–1466.

(24) G. Srinivasan, C. M. James and J. A. Krzycki, "Pyrrolysine Encoded by UAG in Archaea: Charging of a UAG-Decoding Specialized tRNA", *Science*, 2002, **296**, 1459–1462.

(25) G. Ramachandran, C. Ramakrishnan and V. Sasisekharan, "Stereochemistry of polypeptide chain configurations", *Journal of Molecular Biology*, 1963, **7**, 95–99.

(26) *RCSB Protein Data Bank*, accessed April 2016, `http://www.rcsb.org`.

(27) H. M. Berman, J. Westbrook, Z. Feng, G. Gilliland, T. N. Bhat, H. Weissig, I. N. Shindyalov and P. E. Bourne, "The Protein Data Bank", *Nucleic Acids Research*, 2000, **28**, 235–242.

(28) W. van Gunsteren, "Constrained dynamics of flexible molecules", *Molecular Physics*, 1980, **40**, 1015–1019.

(29) I. G. Tironi, R. Sperb, P. E. Smith and W. F. van Gunsteren, "A generalized reaction field method for molecular dynamics simulations", *The Journal of Chemical Physics*, 1995, **102**, 5451–5459.

(30) W. F. van Gunsteren, S. Billeter, A. Eising, P. Hünenberger, P. Krüger, A. Mark, W. Scott and I. Tironi, *Biomolecular simulation: the GROMOS96 manual and user guide*, Vdf Hochschulverlag AG an der ETH Zürich, Switzerland, 1996.

(31) W. R. P. Scott, P. H. Hünenberger, I. G. Tironi, A. E. Mark, S. R. Billeter, J. Fennen, A. E. Torda, T. Huber, P. Krüger and W. F. van Gunsteren, "The GROMOS Biomolecular Simulation Program Package", *The Journal of Physical Chemistry A*, 1999, **103**, 3596–3607.

(32) M. Christen, P. H. Hünenberger, D. Bakowies, R. Baron, R. Bürgi, D. P. Geerke, T. N. Heinz, M. A. Kastenholz, V. Kräutler, C. Oostenbrink, C. Peter, D. Trzesniak and W. F. van Gunsteren, "The GROMOS software for biomolecular simulation: GROMOS05", *Journal of Computational Chemistry*, 2005, **26**, 1719–1751.

(33) E. Jones, T. Oliphant, P. Peterson *et al.*, *SciPy: Open source scientific tools for Python*, http://www.scipy.org, 2001–2017.

(34) S. van der Walt, S. C. Colbert and G. Varoquaux, "The NumPy Array: A Structure for Efficient Numerical Computation", *Computing in Science & Engineering*, 2011, **13**, 22–30.

Part II.

CherryPicker

6. Introduction

CherryPicker is a program developed for the automated parametrisation of large, biochemical molecules. The parametrisation method utilises a library of previously parametrised small molecules and subgraph isomorphism mappings to locate and combine overlapping fragments of the library molecules to produce parameters for the target molecule. Following a brief introduction to automated parametrisation methods and molecular fragmentation, this part of the thesis covers the development and testing of the CherryPicker parametrisation scheme.

6.1. Automated Molecular Parametrisation

Biomolecular force fields are parameterised for specific types of molecules, such as lipids, proteins or nucleic acids. Unless specifically designed as such, force fields are in general not compatible with each other. If one wishes to simulate a system incorporating novel compounds, for example to investigate how a certain drug molecule interacts with a protein embedded in a membrane, a force field would have to be used that contains parameters for all the required molecules. However, the number of possible novel compounds is near infinite. Having a force field contain parameters for all possible molecules is implausible. Therefore parameters need to be prepared for any compounds not already defined within the force field.

More general force fields are available, such as UFF[1] or MMFF94,[2] which are not specialised for a specific purpose. Such general force fields could be used to skirt the issues involved with simulating novel compounds to an extent. However, changing from a specific to a general force field can have undesirable outcomes as the majority of the system will generally be better described with a more specific force field. With the desire to continue using a specific force field, the most obvious approach to generating molecular parameters is to manually assign parameters to atoms or groups of atoms based on parameters assigned to similar groups already defined within the force field. This is a slow, error-prone and tedious process, requiring a detailed knowledge of both the chemistry of the target molecule, and the underlying philosophy used to develop the original force field, and may not always generate valid results. Consequently, various automated methods have been developed.

6.1.1. Rule Based Approaches

The simplest means of automatically generating parameters for novel molecules is to follow a purely rule based approach. One example of this approach is MKTOP, which generates molecular topologies consistent with the OPLS-AA force field.[3,4] Cartesian coordinates, and optionally a file containing the atomic charges, are supplied to the program. Bonding is determined based on a distance cutoff comparison between pairs of atoms. Higher forms of connectivity, i.e. bond angles and dihedrals, are determined from the generated bonding table. Atom types are determined based on lists of connectivity criteria which are able to distinguish between a number of different functional groups. Atom types are then used to assign force constants to each of the bonds, angles and dihedrals based on parameters already defined within the force field.

Other programs that follow a rule based approach are Antechamber, which generates topologies for the Generalised AMBER force field (GAFF) based on input coordinates,[5,6] PRODRG, discussed in more detail below, which has been created for generation of topologies consistent with the GROMOS force field, again following a similar rule based approach,[7] and STaGE which is more an automated pipeline for calculating solvation free energies, though does have some rule based parameterisation included.[8] As no quantum chemical calculations need to be performed, these methods have the advantage of being incredibly fast. However, if the force field a molecule is being parameterised within does not contain adequate parameters to describe the molecule well enough, the topology generated with a rule based method has the potential to be utterly wrong.

6.1.2. Quantum Mechanics Based Approaches

At the other end of automated procedures to generate molecular parameters are the purely QM based methods, such as the Automated Topology Builder (ATB).[9–11] Like PRODRG, the ATB generates force field descriptions of novel molecules consistent with the GROMOS force field. However, it is a vast improvement over PRODRG. Lemkul *et al.* mentioned a number of serious limitations with PRODRG.[12] As with most other automated systems, ligand protonation or tautomeric states are unable to be user defined, assignment of 1–4 exclusions are occasionally inappropriate, and point charges and charge groups are not assigned in a manner consistent with the GROMOS force field. ATB runs a number of QM calculations at increasingly high levels of theory. From the results of these calculations, charges (and charge groups) are assigned to the atoms. Bond stretch and angle bend terms are assigned from those already defined within the force field, or new ones are generated from the Hessian. This method is much more rigorous than that of PRODRG, but comes with its own limitations. As the ATB is provided as a free web based service, there are limits on the molecule size that can be parameterised. High level

QM results, referenced as QM2, are only produced for molecules with fewer than forty atoms, and involve optimisation at the B3LYP/6-31G(d) level of theory along with calculation of the Hessian. With fewer than fifty atoms in the molecule, the QM1 level is used which is the same as QM2 except there is no Hessian calculation. Finally, if there are more than fifty atoms in the molecule, the QM0 level is used, which is an optimisation using the AM1 semi empirical functional.

As mentioned in section 1.2, assignment of charges is heavily influenced by conformation and symmetry considerations. Symmetry considerations, and to a lesser extent conformation, also influence the equilibrium bond and angle values and their respective force constants. These effects are mitigated by accounting for local symmetry to ensure symmetrically equivalent atoms are assigned the same parameters, and by providing possible alternative terms if parameter assignment is slightly ambiguous. Additionally, currently the torsional terms are not derived from QM calculations, rather they are assigned based on atom typing, so are considered rough estimates.

Paramfit is another QM based program for automated parameterisation.[13] It is designed to determine and optimise torsional parameters in parallel within the AMBER force field using a hybrid genetic algorithm to minimise the number of QM calculations that are required to be performed.

Several programs exist part way between a purely rule based approach, and a purely QM derived approach. Generally, a rule based approach is used to assign the bonded parameters and atom types, and charges are derived from QM calculations. Examples include GENRTF which generates topologies for the CHARMM force field,[14] and R.E.D. Server which generates topologies for the AMBER force field.[15]

6.1.3. Novel Force Field Generation Approaches

Outside of the popular biochemical force fields, there are a few fully automated procedures that generate complete stand-alone force fields. Naturally, these force fields are not compatible with others, though because they are generated in a self-consistent manner, a repository of a number of automatically generated parameter sets could be created.

Wang *et al.* created the ForceBalance method to automatically derive accurate force field parameters referenced to a flexible combination of experimental and theoretical data.[16] The procedure was used to optimise parameters for the popular TIP3P and TIP4P water models, and work is on going to apply it to more complex systems such as organic molecules and proteins.[17]

Grimme's Quantum Mechanically Derived Force Field (QMDFF) is a black-box type procedure that can generate a classical force field from only QM computed input data, for almost

any arbitrary structure.[18] Though not recommended for biochemical simulations, instead being more suited to single molecule simulations or self-solvation, QMDFF has one feature of particular interest. Unlike the other force fields mentioned here, it is designed to allow smooth transitions from covalently bound to nonbonded states, i.e. atomisation. This is a step on the way to improving one of the major limitations of classical force fields: bonds cannot be created or removed, only conformational changes can be investigated. Conceivably, this ability to atomise a molecule within the force field could be harnessed to allow a force field to make and break bonds within a system, bringing more chemistry into biochemical investigations.

Automated procedures for molecule parameterisation are not a one stop shop for the lazy chemist. Each method has pitfalls that must be understood and weighed up against the benefits. Parameters generated should always be checked to ensure consistency and that they are sensible. Further, rigorous testing should be undertaken, generating simulation results which could be compared with available experimental, or if necessary theoretical data. If nothing else, automated parameterisation procedures provide a reasonable starting point for further optimisation of molecular topologies.

6.2. Molecular Fragmentation

Molecular fragmentation is the breaking of a large molecule into smaller fragments to increase the tractability of computations on the molecule. By dividing a molecule into fragments, properties of each fragment can be calculated independently, and the results combined to predict the same properties for the whole molecule. A thorough review of fragmentation methods within QM calculations, where computational time scales as poorly as the seventh power of the number of basis functions, was undertaken by Gordon *et al.*[19] Such an undertaking is outside of the scope of this thesis, but the fact that fragmentation is used in QM calculations shows that it provides a valid means to increase the effective size of molecules which can be studied.

At the molecular dynamics level, molecular fragmentation is used heavily. Force fields used for protein and nucleic acid simulations are produced in a fragmented or modular way.[20–23] Instead of parameterising entire proteins or nucleic acids, the building blocks of each are parameterised individually, often times using smaller model compounds themselves. Amino acids and nucleotides are then joined in sequence to create the desired macromolecule. The modular approach undertaken is very efficient; obtaining parameters for the twenty amino acids allows the researcher to simulate any protein, and simply introduce point mutations if they so desire.

Another type of biomolecule that would be well suited to a modular parameterisation approach is the lipid. As shown in figure 6.1, lipids are easily broken up into distinct fragments. Currently, the majority of lipid parameters are developed in isolation, focusing on only a few

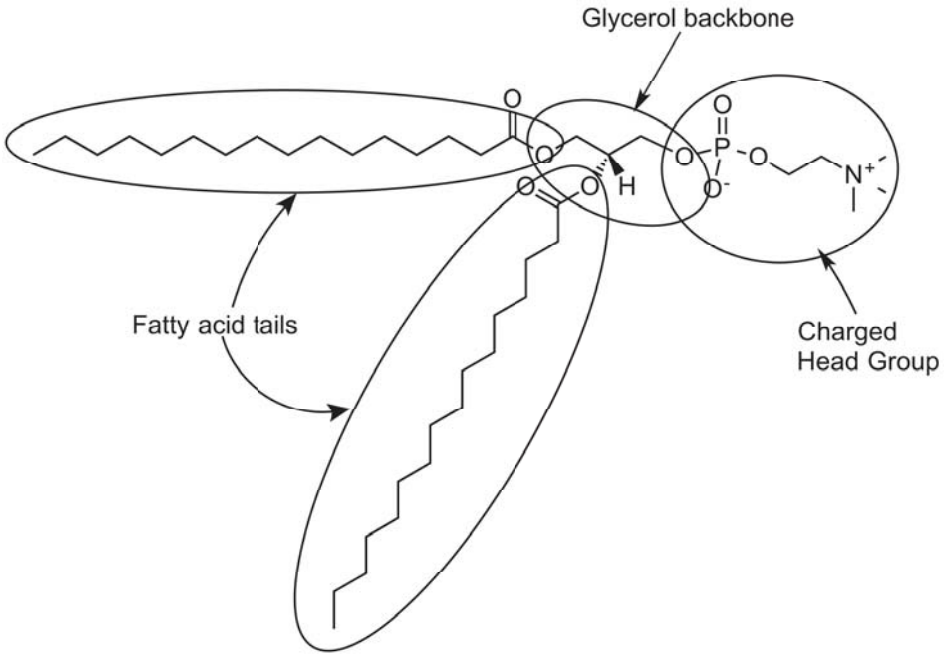


Figure 6.1: A phospholipid showing the three major components of a lipid: a glycerol backbone with a polar head group at the *sn*-3 position, and two fatty acid tails ester-linked at the other two positions.

of the potentially thousands of biologically relevant lipids.[24–27] Several groups have shown the transferability of lipid parameters between different lipids indicating that a modular approach is valid.[28–30] In essence, lipids are already parameterised in a modular manner, with properties of liquid hydrocarbons used as target data for the fatty acid tails[30–33] and small model molecules used to parameterise the head groups.[29,30] Based on these arguments, Skjevik *et al.* developed the LIPID11 parameter set, derived mainly from the GAFF parameters and compatible with the AMBER force field.[34] Success of this parameter set, and its extended LIPID14 sibling,[35] shows the potential of a modular approach when applied to the parameterisation of, at least, phospholipids.

7. Algorithmic Design and Theory

CherryPicker provides an automated means for parametrising large biochemical molecules. Cartesian coordinates for the target molecule to be parameterised with the element type of each atom, the connectivity of the molecule, and an overall molecular charge are required as input. Currently, support is only provided for PDB formatted input coordinates. It is assumed that although the input coordinates may not be optimal, they are sensible. This means that, for instance, aromatic rings are planar, carbon-carbon double bonds are not rotated, four-coordinate carbon centres are close to tetrahedral and so on. Currently, only hydrogen, carbon, nitrogen, oxygen, phosphorous, sulfur, fluorine, chlorine and bromine are supported elements, but this set can easily be extended. With the information supplied as input, a condensed molecular graph (section 7.2) is produced containing all pertinent information, such as the formal charge on atoms and bond orders (chapter 9). Fragments from a library of previously parametrised small molecules (section 7.4) are matched with the condensed molecular graph using subgraph isomorphism testing (section 7.5). The parameters of these matched library fragments are used as the basis of parameters for the target molecule (section 7.6). Parameters are output in either a GROMACS or GROMOS molecular dynamics simulation engine compatible format, utilising the GROMOS force field, in either its AA or UA form. A schematic representation of the algorithm is given in figure 7.1. Support for other force fields and simulation engines can be provided by simply extending the input–output methods.

7.1. Mathematical Concepts

Throughout this part, discrete mathematical concepts relating to the fields of set theory and graph theory are heavily utilised. To aid the reader, a number of basic definitions are given here. Set definitions are as per Gallier,[36] and graph theory definitions are as per Diestel.[37] Additional definitions can be found there as required.

7.1.1. Set Definitions

Definition 7.1. A *set* A is an unordered collection of objects, without duplicates. A *multiset* $\mathcal{A}$ is an unordered collection of objects, with duplicates. A set is regarded as a single object. If

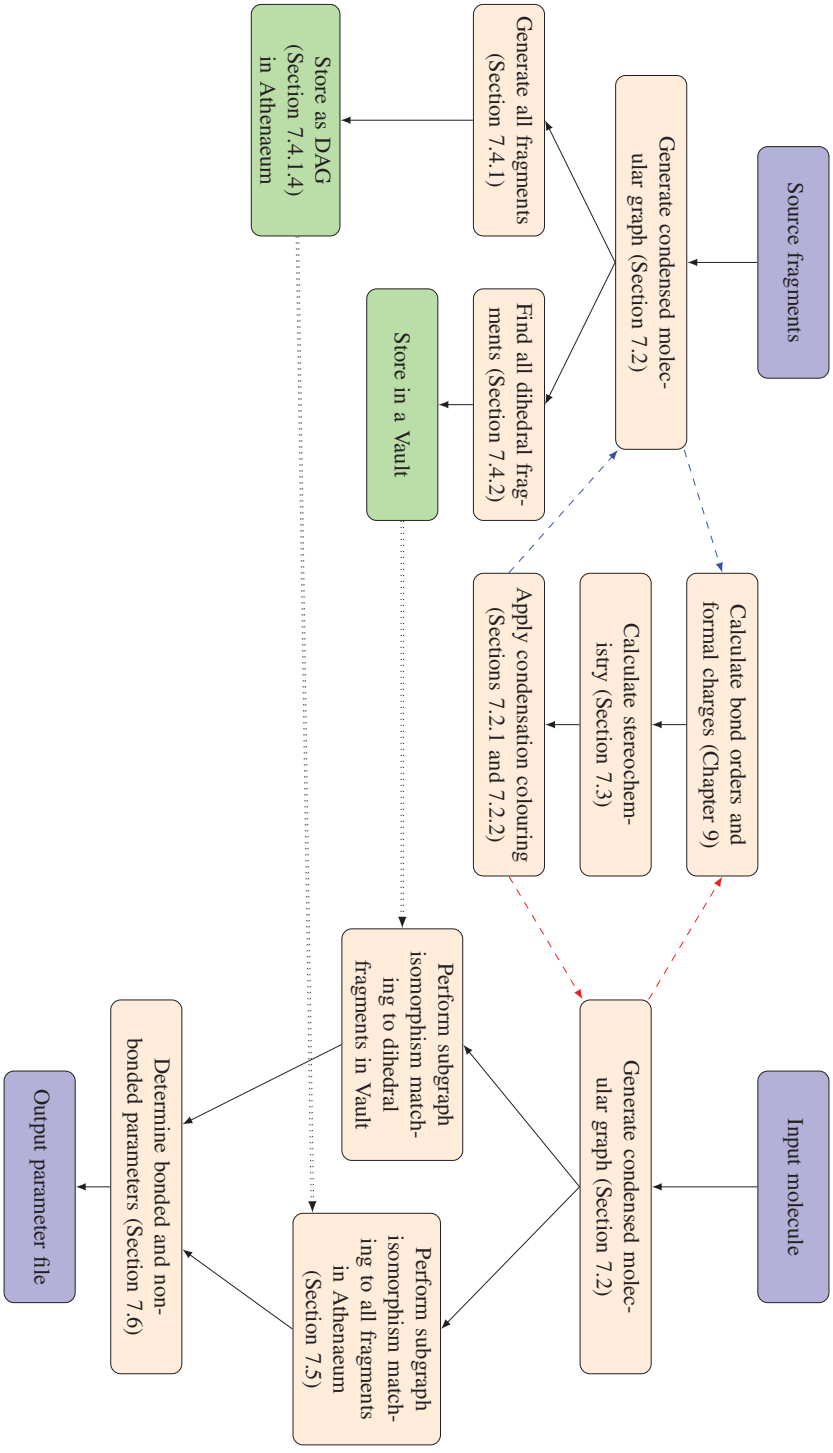


Figure 7.1: Schematic representation of the CherryPicker algorithm.

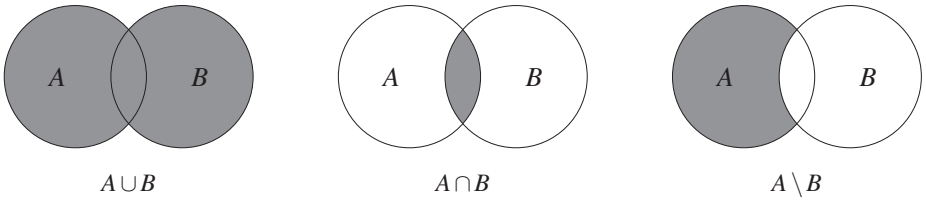


Figure 7.2: Venn diagrams of the set operations, union (left), intersection (centre) and difference (right). The shaded regions show the result of the set operation.

some object a is a *member* of (belongs to) A, it is written as $a \in A$. If a is not a member of A, it is written as $a \notin A$. A set can be defined explicitly by listing its members within curly braces ({ and }) or as a collection of objects satisfying a certain property. For example the set C consisting of the colours red, blue and green is given by $C = \{\text{red}, \text{blue}, \text{green}\}$.

Definition 7.2. Let A and B be sets. A and B are *equal* if and only if they have exactly the same members; that is, every member of A is a member of B and vice versa. There is a special set having no members at all, the *empty set*, denoted $\emptyset$. A is a *subset* of B, denoted $A \subseteq B$, if and only if every member of A is also a member of B. A is a *proper subset* of B if and only if $A \subseteq B$ and $A \neq B$. This implies that there is some $b \in B$ with $b \notin A$. This is usually written as $A \subset B$.

Definition 7.3. If a set A has a finite number of members, then this number is called the *cardinality* of the set and is denoted by $|A|$. The cardinality of the empty set is 0. The *multiplicity* of a member $a \in \mathcal{A}$ is the number of duplicates of a and is written as $\text{mult}(\mathcal{A}, a)$.

Definition 7.4. Let A and B be sets. The *union* of A and B, written as $A \cup B$, is defined such that x is a member of $A \cup B$ if $x \in A$ or $x \in B$. The *intersection* of A and B, written as $A \cap B$, is defined such that x is a member of $A \cap B$ if $x \in A$ and $x \in B$. The *set difference* of A and B, written as $A \setminus B$, is defined such that x is a member of $A \setminus B$ if $x \in A$ and $x \notin B$. These concepts are illustrated through the use of Venn diagrams in figure 7.2. Two sets are *disjoint* if $A \cap B = \emptyset$.

Definition 7.5. Let A and B be two sets. The set of all ordered pairs (a, b), with $a \in A$ and $b \in B$, is a set denoted by $A \times B$ and called the *cartesian product* of A and B. This can be generalised to the *n-ary product*. Given n sets $A_1, \ldots, A_n$ with $n \geq 2$, the members of $A_1 \times A_2 \times \cdots \times A_n$ are the n-tuples $(a_1, a_2, \ldots, a_n)$ with $a_1 \in A_1$, $a_2 \in A_2 \ldots$ $a_n \in A_n$.

7.1.2. Graph Theoretic Definitions

Definition 7.6. A *graph* is a pair $G = (V, E)$ of sets such that $E \subseteq [V]^2$. The elements of V are the *vertices* of the graph G; the elements of E are its *edges*. Figure 7.3 shows a drawing of

a graph. How the vertices and edges are drawn is considered irrelevant; all that matters is the information of which pairs of vertices form an edge and which do not.

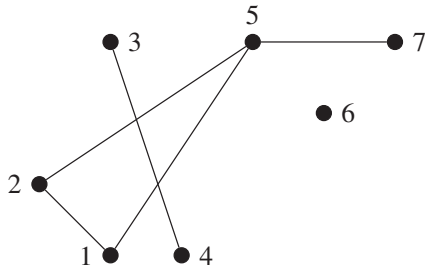


Figure 7.3: The graph on $V = \{1,\dots,7\}$ with edge set $E = \big\{\{1,2\},\{1,5\},\{2,5\},\{3,4\},\{5,7\}\big\}$.

Definition 7.7. Let $G = (V,E)$ be a graph. Two vertices x,y of G are *neighbours* if $\{x,y\}$ is an edge of G. Two edges $e \neq f$ are *adjacent* if they share a vertex. The set of neighbours of a vertex v in G is denoted by $N(v)$. The *degree* $d(v)$ of a vertex v is the number of neighbours of v.

Definition 7.8. Let $G = (V,E)$ and $G' = (V',E')$ be two graphs. If $V' \subseteq V$ and $E' \subseteq E$, then G' is a *subgraph* of G, written as $G' \subseteq G$. If $G' \subseteq G$ and $G' \neq G$, then G' is a *proper subgraph* of G. If $G' \subseteq G$ and G' contains all the edges $xy \in E$ with $x,y \in V'$, then G' is an *induced subgraph* of G and written as $G' \coloneqq G[V']$ (figure 7.4).

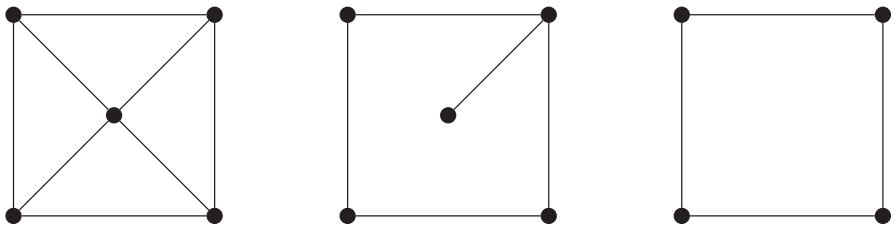


Figure 7.4: A graph G with subgraphs G' and G'': G'' is an induced subgraph of G, but G' is not.

Definition 7.9. A *path* is a non-empty graph $P = (V,E)$ of the form

$$V = \{x_0,x_1,\dots,x_k\} \qquad E = \{x_0x_1,x_1x_2,\dots,x_{k-1}x_k\},$$

where the x_i are all distinct. The number of edges of a path is its *length*, and the path of length k is denoted by P^k (figure 7.5). A path is often referred to by the natural sequence of its vertices:

$P = x_0x_1 \dots x_k$. For $0 \leq i \leq j \leq k$ subpaths of P can be written as

$$Px_i \coloneqq x_0 \dots x_i$$
$$x_iP \coloneqq x_i \dots x_k$$
$$x_iPx_j \coloneqq x_i \dots x_j$$

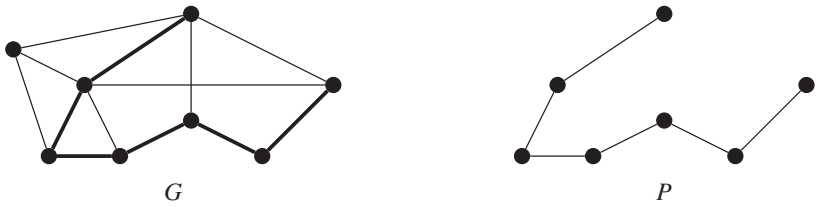


Figure 7.5: A path $P = P^6$ in G.

Definition 7.10. Let $P = x_0x_1 \dots x_{k-1}$ be a path. Then if $k \geq 3$, the graph $C \coloneqq P + x_{k-1}x_0$ is called a *cycle*. As with paths, a cycle is often denoted by its (cyclic) sequence of vertices; $x_0 \dots x_{k-1}x_0$. The *length* of a cycle is its number of edges and is denoted by C^k. An edge which joins two vertices of a cycle but is not itself an edge of the cycle is a *chord* of that cycle. An *induced cycle* in G, a cycle in G forming an induced subgraph, is one that has no chords (figure 7.6).

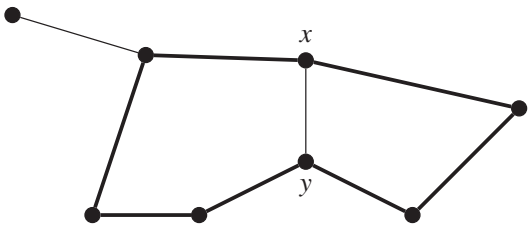


Figure 7.6: A cycle C^8 with chord xy, and induced cycles C^5,C^4.

Definition 7.11. A graph $G = (V,E)$ is called *connected* if it is non-empty and any two of its vertices are linked by a path in G. A *maximal connected subgraph* of G, $G' = (V',E')$, is a graph which is connected and for all vertices u such that $u \in V$ and $u \notin V'$, there is no vertex $v \in V$ for which $uv \in E$. A maximal connected subgraph of G is a *component* of G (figure 7.7).

Definition 7.12. An *acylic* graph, one not containing any cycles, is called a *forest*. A connected forest is called a *tree*. Thus a forest is a graph whose components are trees. The vertices of degree 1 in a tree are its *leaves*. Sometimes it is convenient to consider one vertex of a tree as

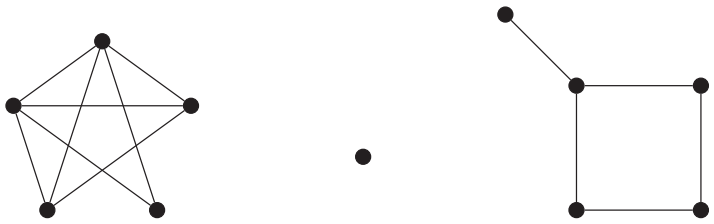


Figure 7.7: A three component graph

special; such a vertex is then called the *root* of this tree. The root of a tree is never called a leaf, even if it has degree 1. A tree T with a fixed root r is a *rooted tree* (figure 7.8).

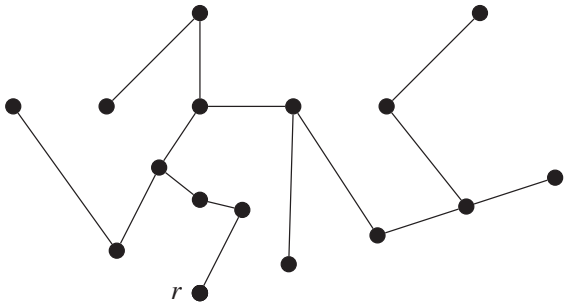


Figure 7.8: A tree with root r.

Definition 7.13. A *directed graph* is a pair (V,E) of disjoint sets (of vertices and edges) together with two maps $\text{init} : E \to V$ and $\text{ter} : E \to V$ assigning to every edge e an *initial vertex* $\text{init}(e)$ and a *terminal vertex* $\text{ter}(e)$. The edge e is said to be *directed from* $\text{init}(e)$ to $\text{ter}(e)$. A directed graph may have several edges between the same two vertices x,y called *multiple edges*. If $\text{init}(e) = \text{ter}(e)$, the edge e is called a *loop*. An *oriented graph* is a directed graph without loops or multiple edges.

Definition 7.14. A *vertex colouring* of a graph $G = (V,E)$ is a map $c : V \to S$ such that $c(v) \neq c(w)$ whenever v and w are adjacent. The elements of the set S are called the available *colours*. An *edge colouring* of G is a map $c : E \to S$ with $c(e) \neq c(f)$ for any adjacent edges e, f. A weakening of the vertex and edge colouring definitions by allowing $c(v) = c(w)$ and $c(e) = c(f)$ respectively leads to the *molecular graph*. A molecular graph $M = (A,B)$ has a weak vertex colouring $c_v : A \to P$ where P is the set of atomic numbers for elements in the periodic table, and a weak edge colouring $c_e : B \to R$ where R is the set of allowable bond orders. This definition comes from Nic *et al.*[38]

Definition 7.15. A *plane graph* is a pair (V,E) of finite sets with the following properties:

1. $V \subseteq \mathbb{R}^2$;

2. every edge is an arc between two vertices;

3. different edges have different sets of endpoints;

4. the interior of an edge contains no vertex and no point of any other edge.

When G is a plane graph, we call the regions of $\mathbb{R}^2 \setminus G$ the *faces* of G. These are open subsets of $\mathbb{R}^2$ and hence have their frontiers in G. Since G is bounded–i.e., lies inside some sufficiently large disc D–exactly one of its faces is unbounded, the face that contains $\mathbb{R}^2 \setminus D$. This face is the *outer face* of G; the other faces are its *inner faces*. We denote the set of faces of G by $F(G)$. A *planar graph* is a graph which can be embedded in $\mathbb{R}^2$ to form a plane graph.

Definition 7.16. Let $G = (V,E)$, $G' = (V',E')$ be two graphs. A map $\varphi : V \to V'$ is a *homomorphism* from G to G' if it preserves the adjacency of vertices, that is, if $\{\varphi(x), \varphi(y)\} \in E'$ whenever $\{x,y\} \in E$. If φ is bijective and its inverse φ^{-1} is also a homomorphism (so that $xy \in E \Leftrightarrow \varphi(x)\varphi(y) \in E'$ for all $x,y \in V$), we call φ an *isomorphism*, say that G and G' are *isomorphic*, and write $G \simeq G'$. An isomorphism from G to itself is an *automorphism* of G. Let $G'' = (V'',E'')$ be a subgraph of G. If a homomorphism bijective map $\varphi : V' \to V''$ exists from G' to G'', then G' is *subgraph isomorphic* with G, that is G' is isomorphic with a subgraph of G.

7.2. Condensed Molecular Graph

The vertex and edge colourings of a molecular graph (definition 7.14) contain insufficient molecular information to be used for the parametrisation process, primarily a lack of stereochemical information and formal charge information on the atoms. Additionally, they contain a large number of leaves which can cause exponential explosions in the number of (sub)graph isomorphic mappings between two molecular graphs. To bypass these issues, the concept of a condensed molecular graph is introduced.

Definition 7.17. Let $M = (A,B)$ be a molecular graph, with weak vertex colouring $c_{v_M} : A \to P$ and weak edge colouring $c_{e_M} : B \to R$ as per definition 7.14. A *condensed molecular graph* $G = (V,E)$ is then the subgraph $G = M[A']$ where $A' \subseteq A$ is the set of vertices $a \in A$ with $d_M(a) \neq 1$ unless $d_M(a) = 1$ and $c_{e_M}(\{a\} \cup N_M(a)) \neq 1$ or $c_{v_G}(a) \wedge 7 \neq 0$. A condensed molecular graph has a vertex colouring $c_{v_G} : V \to S_v$ where S_v is the set of vertex colours as per section 7.2.1, and an edge colouring $c_{e_G} : E \to S_e$ where S_e is the set of edge colours as per section 7.2.2. A vertex which is in the molecular graph but not in the condensed molecular graph has been

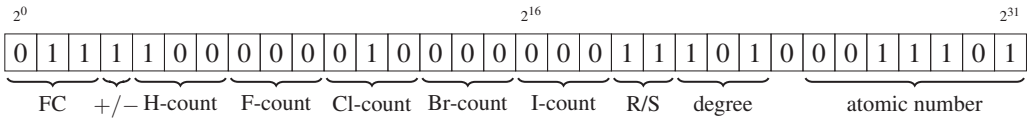


Figure 7.9: Bit string representation of the vertex colour 32-bit integer. This string has the integer value 3,099,068,446 and corresponds to the (unrealistic) situation of a uranium atom with a formal charge of −6, one condensed hydrogen atom, two condensed chlorine atoms, S stereo chemistry and a degree of 5 in the normal molecular graph.

incorporated into its parent vertex and is known as a *condensed* vertex. The vertex colouring contains information on the condensed vertices.

To aid in packing large amounts of information into as little memory as possible, the notion of bit manipulation is utilised in determining the colours available to the vertex and edge colourings. Here bit manipulation is used to compress multiple types of information into a single integer by assigning the different types of information to unique index ranges within the bit representation of the integer.

7.2.1. Vertex Colours

Vertex colours utilise 31 of the available 32 bits of the 32-bit integer used for each available colour. These 31 bits contain ten distinct pieces of information, as shown in figure 7.9 and detailed below, from least significant bit to most significant bit.

$2^0 \rightarrow 2^2$ These two bits give the formal charge magnitude on the atom, as determined by the FPT algorithm described in section 9.4.4. The use of three bits allows for formal charge magnitudes of between zero and seven, which should cover all reasonable formal charge values. The binary value of the formal charge is used to populate the three indices.

2^3 This bit gives the sign of the formal charge. It is set to `0` if the formal charge is zero or positive and `1` if it is negative.

$2^4 \rightarrow 2^6$, $2^7 \rightarrow 2^9$, $2^{10} \rightarrow 2^{12}$, $2^{13} \rightarrow 2^{15}$ and $2^{16} \rightarrow 2^{18}$ These five groups of three bits represent the counts of the different elements which could have been condensed into a vertex in the transition from a molecular graph to a condensed molecular graph. The rules governing which vertices may be condensed into their parent vertex (definition 7.17) effectively limit the elements that could be condensed to hydrogen and the halogens. Though it is unlikely that there would be a situation where more than three vertices of the same molecular graph colour need

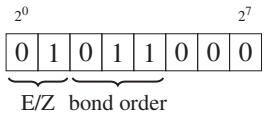


Figure 7.10: Bit string representation of the edge colour 8-bit integer. This string has the integer value 26 and corresponds to the (unrealistic) situation of a bond with E stereo chemistry and an order of 6.

to be condensed into the same parent vertex, there are some simple situations, such as methane, where it is possible, so three bits are required rather than just two. Again, the values of the bits are set by the binary value of the integer count of each element.

$2^{19} \rightarrow 2^{20}$ These two bits are used to represent any chirality associated with an atom. Using the Cahn–Ingold–Prelog priority rules (CIP) rules and the cartesian coordinates from which a molecular graph is generated, an estimate of the stereo chemistry (section 7.3) of an atom with four unique neighbours is made.[39,40] An estimate is made, as opposed to an exact calculation, as the stereo chemistry of an atomic centre could change depending on the size of the fragment that it is in. To make this estimate, only atoms within a path of length k from the centre of interest are included in the calculation. For the purposes of this description, k is assumed to be set to two. There are three possible states that the chirality of a centre can take, achiral, represented by `00`, R, represented by `01` and S, represented by `11`.

$2^{21} \rightarrow 2^{23}$ The degree of the vertex within the molecular graph is given by these three bits. This information is somewhat redundant, but it could be useful in some matching situations. The three bits are set to match the binary value of the integer degree.

$2^{25} \rightarrow 2^{31}$ These final seven bits represent the atomic number of the element associated with the vertex. Currently, there are 118 elements in the periodic table, meaning they can all be represented by the seven bits available. Naturally, the binary value of the atomic number is used to fill the seven bits.

7.2.2. Edge Colours

Edge colours utilise five of eight bits in an 8-bit integer. These five bits contain two types of information, as shown in figure 7.10 and detailed below.

$2^0 \rightarrow 2^1$ These two bits contain information about the stereochemistry of double bonds. As is the case for the stereochemistry of atom centres, this only gives an estimate due to the potential

for the result to change depending on the size of the fragment. If a bond is not a double bond, or the double bond is symmetric, a value of `00` is used. If a double bond has *E* stereochemistry, a value of `01` is used, and a value of `11` is used for *Z* stereochemistry.

$2^2 \rightarrow 2^4$ These three bits contain the bond order as produced by the FPT algorithm described in section 9.4.4. The bits are set to match the integer value of the order, unless the bond is aromatic, in which case the bits are set to `111`. An aromatic bond is determined using the bond orders obtained by the FPT algorithm and the aromaticity perception algorithm in OpenBabel.[41]

7.3. Stereochemical Determination

The CIP are a standard process developed by Cahn *et al.* used to completely and unequivocally assign an R or S descriptor to each stereocenter and an E or Z descriptor to each double bond of a molecule.[39,40] The general idea is that each substituent of a possible stereocenter or double bond is assigned a priority based on the atomic number of the substituent. The order in which the substituents are arranged in space, as given by their relative priorities, dictate the descriptor assigned.

7.3.1. CIP String Generation

A CIP string is the means used to determine the relative priorities of each substituent. It is a nested list of lists where each of the inner lists is an ordered list of the atomic number of all atoms with a path length of k, and the position of the inner list in the outer list, from the centre or bond of interest.

7.3.2. R/S Chirality Determination

Given that a possible stereocentre has four substituents, each with a different CIP string, chirality determination proceeds as follows. Coordinates of the stereocentre and its four neighbours are extracted and geometrically manipulated in order to determine the chirality. First, the submolecule is centred on the possible stereocentre. Two rotations are then applied. The first rotates the submolecule such that the neighbour with the lowest priority lies along the z-axis, in the negative direction. The second rotation rotates about the z-axis so that the neighbour with the highest priority lies in the yz-plane. The sign of the x position of the second highest priority neighbour then defines the chirality. A positive x position gives an R stereocentre, and a negative x position gives an S stereocentre.

7.3.3. E/Z Geometric Isomerisation Determination

Given a double bond where each atom of the bond has two substituents with different CIP strings, geometric isomerisation determination proceeds as follows. Coordinates of the two atoms of the bond and their neighbours are extracted and geometrically manipulated. The submolecule is centred on the midpoint of the bond. A rotation is applied to rotate the bond so that it lies along the z-axis, followed by a rotation about the z axis to align the submolecule in the xz-plane. If the highest priority substituents on each atom are on the same side of the z-axis, the bond is labeled as a Z ismoer, otherwise it is an E isomer.

7.4. Athenaeum

The CherryPicker algorithm requires a collection of already parameterised molecules and fragments derived from them. An Athenaeum is a repository of parameterised small molecules and the set of all possible fragments that can be produced from them. There are three important concepts associated with an Athenaeum: how a fragment is defined, how a fragment is stored and how the set of all fragments of a molecule is stored. Each of these areas will be addressed in turn.

7.4.1. Fragment definition

A fragment is a connected induced subgraph of a condensed molecular graph. A fragment of a molecule consists of two parts, the retained fragment section and an overlap region. The overlap region helps to ensure that fragments are matched to areas which have a similar chemical environment to that of their source molecule.

7.4.1.1. Fragment generation

To produce all possible fragments of a given molecule, all the connected subgraphs of a condensed molecular graph need to be checked for validity (section 7.4.1.2), which means that all the connected subgraphs need to be produced. A given graph $G = (V, E)$ has $2^{|V|}$ induced subgraphs. A naive approach to generating all connected subgraphs would be to check each of those induced subgraphs for connectivity. A more feasible approach is given in algorithm 1 where only subgraphs which are connected can be generated. Use of the condensed molecular graph has the benefit of not only containing more information than the molecular graph, but also generation of all connected subgraphs is more computationally efficient, due to the smaller number of vertices.

Algorithm 1 Generate all connected subgraphs of a graph

```
1:  procedure ConnectSub(G, B, H, S)   ▷ Generate all connected subgraphs of G = (V,E)
2:      B ← V                          ▷ Remaining vertices that could be part of subgraph
3:      H ← ∅                                                          ▷ Current subgraph
4:      S ← ∅                                                  ▷ Neighbours of vertices in H
5:      append (B,H,S) to stack
6:      while length stack > 0 do
7:          pop B, H, S from stack
8:          if H = ∅ then
9:              R ← B
10:         else
11:             R ← B ∩ S
12:         end if
13:         if R = ∅ and H ≠ ∅ and SizeCheck(H) then
14:             yield G[H]
15:         else if R ≠ ∅ then
16:             v ← min(R)
17:             if MaxSizeCheck(H) then
18:                 append (B \ {v},H,S) to stack
19:                 append (B \ {v},H ∪ {v},S ∪ N(v)) to stack
20:             else if SizeCheck(H) then
21:                 yield G[H]
22:             end if
23:         end if                     ▷ No need to return the empty set subgraph
24:     end while
25: end procedure
```

SizeCheck and MaxSizeCheck are functions used to check that the number of vertices in a subgraph fall within some upper and lower bounds. MaxSizeCheck checks only that the subgraph has fewer vertices than the upper bound, whereas SizeCheck checks both upper and lower bounds. When generating subgraphs of a condensed molecular graph, the number of condensed vertices in each vertex can be included in the vertex count of the subgraph.

7.4.1.2. Fragment Validity

Each subgraph generated is checked for validity, determining whether or not it can be used as a fragment. There are four rules used to determine if a subgraph can be used as a fragment, each described below. Throughout these descriptions, let $G = (V,E)$ be a condensed molecular graph, $G' = (V',E')$ be the induced subgraph of G being checked for validity, and $G'' = (V'',E'')$ be the induced subgraph of G such that $V' \cup V'' = V$.

Algorithm 2 Generate the set of minimal cycles of a graph

```
1:  procedure MINIMALCYLES(G)
2:      function ALLCYCLES(G)                          ▷ Generate all cycles in G
3:          S ← CYCLEBASIS(G)
4:          for k ∈ {1,2,...,|S|,|S|+1} do
5:              for combo ∈ (S choose k) do            ▷ all k combinations of S
6:                  yield ⨁ combo
7:              end for
8:          end for
9:      end function
10:     C ← ALLCYCLES(G)
11:     ascending sort C by size
12:     s ← min(size(c ∈ C))
13:     for c ∈ C do
14:         if size(c) > s and all edges seen then      ▷ check all cycles of same size
15:             break
16:         else if size(c) > s then
17:             s ← size(c)
18:         end if
19:         if s > 6 and G is non-planar then
20:             break                                  ▷ only give cycles up to size 6 if non-planar
21:         end if
22:         assert all edges only seen max once
23:         assert c has unseen edges
24:         yield c
25:     end for
26: end procedure
```

Cycles Cycles, or ring structures, are an important part of biochemical molecules, so fragments containing cycles or parts thereof are given special treatment. For the purposes of fragment validation, if an induced cycle has a length greater than eight, it is not treated as a cycle and only the other three fragment validation rules apply. In order to treat cycles within a condensed molecular graph differently, they must be identified.

Algorithm 2 lays out an algorithm which generates the set of minimal cycles of a planar graph. Minimal cycles are the set of cycles which cover all cyclic edges, i.e. those edges which are in a cycle, within a graph. In essence, these are the faces of a planar graph, however there are some situations where faces will be missed. The outer face will always be missed, and if there is a cycle which shares all its edges with other cycles, that cycle will be missed in the case where its length is greater than the length of the largest cycle it shares an edge with. This algorithm can be used to find the faces of a planar graph (as opposed to a plane graph) as generally in chemistry, the

smaller the cycle, the more interesting it is. In the unlikely case where a condensed molecular graph is not planar*, algorithm 2 returns only cycles with up to six vertices. This allows for identifying some cycles while likely avoiding issues caused by the non planarity. CYCLEBASIS is the algorithm to determine a cycle basis described in Paton[42] as implemented in the NetworkX graph library.[43] A cycle basis of a graph is a minimal collection of cycles such that any cycle in the graph can be written as a sum of cycles in the basis.

A fragment containing cycles is deemed valid if there are no partial cycles in $G[V']$, or there are no partial cycles in $G[V'']$. A partial cycle is a cycle C^k in G where $C^k \cap V' \neq \emptyset$ and $C^k \cap V'' \neq \emptyset$.

Bond Order Any edge e incident on G', i.e. $e \in E \setminus (E' \cup E'')$, must be either aromatic or have a bond order of one. This is determined through the condition $c_{e_G}(e) \wedge k_7 \in \{k_1, k_7\}$, where k_x is the integer value of a bond order of x when bit shifted to match with the correct position in the edge colour bit string. In combination with the cycles rule, this allows for partial fused aromatic ring systems to be used to match with much larger systems.

Hetero Bonds Any edge $e = u, v$ incident on G', i.e. $e \in E \setminus (E' \cup E'')$, must have at least one carbon atom as an end point. This is determined through the condition $\{c_{v_G}(u) \wedge k_{127}, c_{v_G}(v) \wedge k_{127}\} \supseteq \{k_6\}$ where k_x is the integer value of atomic number x when bit shifted to match with the correct position in the vertex colour bit string.

Overlap All fragments require an overlap region, that is the subgraph needs to be able to be extended from all edges incident on G' by a path of length at least k. Thus, given an edge $e = u, v$ with $u \in V'$ and $v \in V''$, the overlap vertices from that edge are all those vertices $y \in V''$ with a path $P = x_0 \ldots x_i$ that has length $i \leq k$ in G and $V_P \cap V' = \{u\}$, where $x_0 = u$ and $x_i = y$. A separate path length k_c is utilised for any edge $e = u, v$ with $u \in V'$ and $v \in V''$ where u is in a cycle. Figure 7.11 shows an example fragment with overlap regions.

7.4.1.3. Fragment Storage

Once a fragment has been generated and validated, it must be stored in a memory efficient manner. The storage method must include a means for distinguishing between the retained fragment section, and the overlap region. Parameters associated with the fragment need not be stored as they can easily be obtained from the subgraph $G[V']$ of the condensed molecular graph, where V' is the retained fragment vertex set. As a fragment only needs to be created once, additional expensive to calculate information about that fragment could conceivably be stored with the

*none of the 17499 molecules used as test cases in chapter 9, nor any of the 9064 molecules in SRC9064 have non-planar molecular graphs

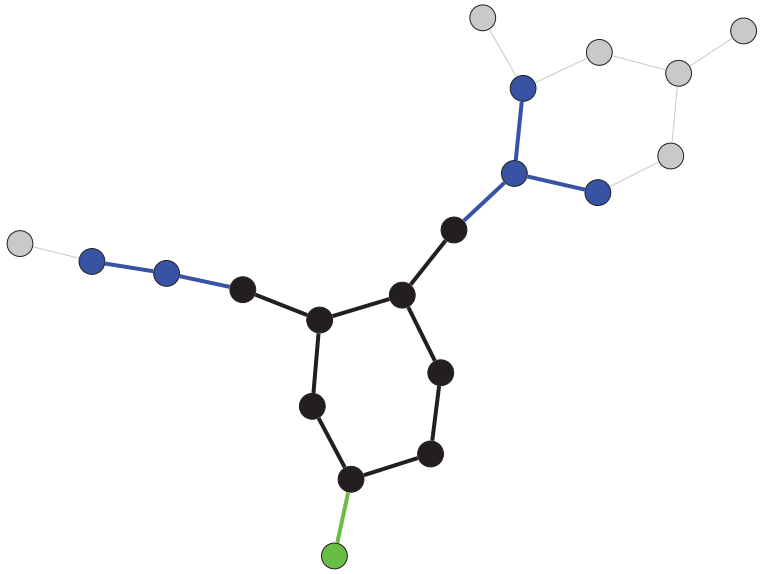


Figure 7.11: Fragment of a graph with overlap regions. The subgraph under consideration as a fragment is given in black. The blue vertices and edges are the overlap region from edges not incident to a cycle, with $k = 2$. The green vertex and edge are the overlap region from edges incident to a cycle, with $k_c = 1$. Vertices and edges not part of the fragment and overlap region are in grey.

fragment. Given these conditions, the storage method used is the tuple $(x, -1, y_1, -2, y_2, \dots)$ which provides the additional information of how far a vertex in the overlap region is from the retained fragment. Here, x is the vertex set of the retained fragments V', y_1 is the set of vertices in the overlap region with a path length of 1 away from the retained vertices, y_2 is the set of vertices in the overlap region with a path length of 2 away from the retained vertices, and so on. Negative integers are used as the divider between vertex sets as vertices in a condensed molecular graph are always labelled with positive integers. The magnitude of the negative indices gives the path length of all vertices in the proceeding set to a vertex in V'. Vertices in the overlap region which are incident to a cycle vertex in the retained fragment are treated independently to those not; the magnitude of the preceding negative integer is equal to the path length plus nine. This implies an upper bound on the value of the overlap path length of 9. With fused cycles, it is possible for a vertex to have two or more possible paths to a vertex in the retained fragment. In this case, the vertex is only placed in the vertex set with the shortest path length.

7.4.1.4. Molecular Fragment Set Storage

The set of fragments of a molecule with condensed molecular graph $G = (V, E)$ are stored as a directed acyclic graph (DAG) $D = (U, L)$. The vertices $u \in U$ are mapped to fragments of

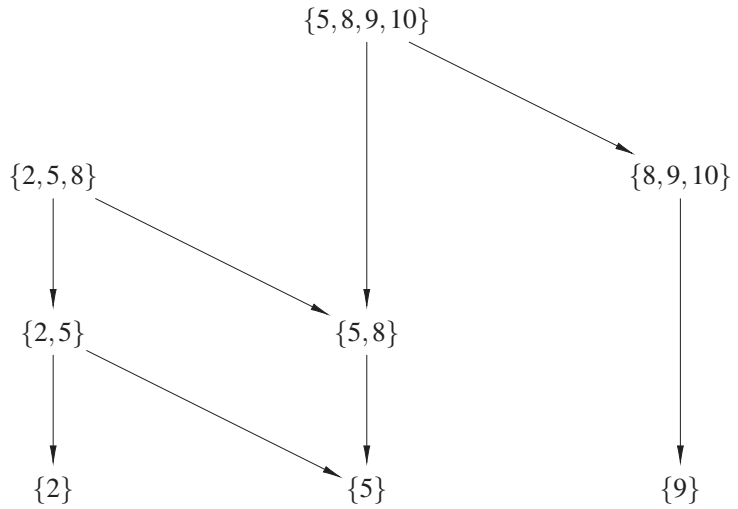


Figure 7.12: Example of the DAG produced from the fragment set of an arbitrary molecule. The sets at each vertex represent the vertex indices of vertices within the condensed molecular graph which are contained within a fragment. They are non-consecutive as they match the indices in the regular molecular graph. Vertices part of the overlap region of a fragment are not shown. An arrow from one set to another indicates that the target set is a subset of the source set.

G through an index set T. An edge is added from $u_1 \in U$ to $u_2 \in U$ if, given $T(u_1) = V'$ and $T(u_2) = V''$, $G[V''] \subseteq G[V']$. This gives a directed graph which has at least one path from any vertex to any other vertex which is a subgraph of that vertex. An edge $e = \{x,y\} \in L$ is removed from the graph if there is a path $P = x \ldots y$ in the graph $H = (V, E \setminus \{x,y\})$. Removing edges this way gives a directed graph which has a unique path between any two vertices where one is a subgraph of the other. An example of a final DAG is given in figure 7.12. The structure of this DAG plays an important role in optimisation of the subgraph isomorphism testing procedure, allowing for fewer subgraph isomorphism test to be performed.

7.4.2. Dihedral Fragments

As part of an Athenaeum, dihedral fragments are generated and stored separately (in a Vault) to the fragments defined previously. A dihedral is defined for every edge $\{x,y\}$ of a graph when both vertices of the edge have degree larger than one. A dihedral fragment of a graph $G = (V,E)$ is then the subgraph $G' = G[V']$ where $V' = \{x,y\} \cup N(x) \cup N(y)$. During the generation of fragments from a condensed molecular graph, if the evaporated fragment graph (i.e. the corresponding subgraph of the molecular graph) is a dihedral fragment, it is added to the Vault. The Vault stores dihedral fragments independently of the source molecule they were obtained

from. As such, there will be duplicates. Duplicates are identified through subgraph isomorphism testing when adding a dihedral fragment. As part of the storage of dihedral fragments, the parameters associated with each duplicate fragment are stored in a list. In future, it is hoped to combine this dihedral fragment storage with the SpinningTop program (part I) to calculate dihedral parameters for each dihedral fragment identified.

7.5. Subgraph Isomorphism Testing

A target molecule is passed through the same preparation pipeline as used to prepare molecules for fragmentation, before an Athenaeum is used to generate parameters. The general scheme for parameterisation is that all fragments, including the overlap region, of all molecules in the Athenaeum are checked for subgraph isomorphism with the target condensed molecular graph. If a subgraph isomorphism exists, the map of the retained fragment region is stored. Each fragment can have multiple subgraph isomorphisms with the target condensed molecular graph, and they are all stored. Section 7.6 details how this set of stored subgraph isomorphisms is translated into parameters.

The VF2 algorithm[44] as implemented in the NetworkX Python library[43] is used to identify subgraph isomorphisms. This algorithm has two mapping feasibility criteria for determining isomorphisms, syntactic feasibility and semantic feasibility. Syntactic feasibility depends only on the structure of the graphs, that is the connectivity of the graphs. Semantic feasibility depends on the attributes of the graph. Here, the attributes refer to the vertex and edge colourings of the condensed molecular graph. As the colourings are integers containing multiple pieces of information, for a given target molecule, different pieces of information can be used in the semantic feasibility test by performing a logical `AND` between the colour and the integer representing the pieces of information that are desired to be matched. For example, if only the element type is to be used in the semantic feasibility test, the colour of a vertex, c, can be `AND`'d with 4,261,412,864, the value of $\sum_{i=25}^{31} 2^i$, and the result used to check semantic feasibility. Only if both syntactic and semantic feasibility checks return true is there a subgraph isomorphism between the two graphs.

Checking all fragments of a molecule for subgraph isomorphism can be computationally expensive. To reduce this cost, the structure of the DAG which a set of fragments is stored as is exploited. It is clear that if a fragment does not have a subgraph isomorphism with the target condensed molecular graph, any fragments which contain that fragment as a subgraph will also not have a subgraph isomorphism. Given that the DAG contains a unique path from any vertex representing a fragment to all other vertices which represent subgraphs of that fragment, if a fragment does not have a subgraph isomorphism with the condensed molecular graph, any ver-

tices which have a path to that fragment will also not have a subgraph isomorphism. By checking the smaller fragments first, i.e. the leaves of the DAG, the number of fragments that need to be checked can potentially be reduced. In order for the algorithm to perform this, a topological sort of the vertices of the DAG is used to create the order fragments should be checked in. A topological sort is a non-unique permutation of the vertices of a DAG such that an edge from u to v implies that u appears before v in the topological sort order. As the leaves of the DAG are desired to be checked first, a reverse topological sort is used.

7.6. Parameterisation

From the mappings provided by the subgraph isomorphism testing procedure, parameters for the target molecule are determined. This proceeds in three parallel steps, determination of non-bonded terms, determination of bond and angle terms, and determination of proper and improper dihedral terms. The bonded terms are split into two groups as it allows for future extension of the parameterisation of the dihedrals through an automated dihedral parameterisation scheme as outlined in part I.

The parameterisation algorithm allows for two Athenaeums to be utilised, one containing a small set of manually parameterised fragments and the other a much larger set of automatically parameterised fragments. The manually parameterised Athenaeum is designed to be used as the basis for the parameterisation. For example, if the molecule to be parameterised is a penta-peptide with four natural amino acids and one unnatural amino acid, the first Athenaeum would provide parameters for the natural amino acids, and the second would be used to parameterise the unnatural amino acid. To this end, the first Athenaeum is assumed to be consistent, that is any fragments which have isomorphic subgraphs are assumed to have the same parameters for the isomorphic subgraphs. Any vertices and edges mapped by the first Athenaeum are ignored when determining the parameters from the second Athenaeum.

7.6.1. Condensed Atoms

Because of the use of condensed molecular graphs, the graphs used for subgraph isomorphism testing are incomplete. Thus, to be able to parameterise the complete molecule, any vertices which have been condensed into their parent vertex need to be evaporated out again. The general rule for evaporating out a vertex is that if a mapping between vertex a and vertex b exists, then all evaporated vertices of a map to all evaporated vertices of b of the same element. The proceeding descriptions of parameterisation assume evaporation of condensed vertices, either as an explicit step external to the parameterisation or implicitly as part of the parameterisation step.

7.6.2. Symmetry

Another consideration for parameterisation of molecules is that of symmetry. For each term in a parameterised molecule, the set of that term and all its symmetric terms should have identical values. For example, the charges on the hydrogen atoms of a methyl group should all be identical, as too should the bonds between the hydrogen and carbon atoms, and all the hydrogen-carbon-hydrogen angles. This does not need to be explicitly incorporated into the subgraph isomorphism testing as if a fragment maps to an atom with symmetry partners, it should also map to all the symmetry partners. As a safety net, symmetry is explicitly accounted for in the parameterisation stage.

Symmetric vertices can be easily identified through performing a graph automorphism test on the condensed molecular graph. In the condensed molecular graph, any vertices which have been condensed into their parent vertex are considered to be symmetric with all other vertices of the same element condensed into the same parent vertex, and any vertices condensed into the symmetric partners of the parent vertex. If two vertices are symmetric, then any bonds involving those vertices are also symmetric. Symmetric angles can be identified as those angles where either both bonds of the angle are symmetric, or the central and one of the terminal atoms of the angle are identical and the other atoms are symmetric. A similar argument can be made to identify symmetric dihedral terms. To explicitly account for the symmetry of a molecule while parameterising, all lists of potential parameters produced are assumed to be produced from the mapped fragments of all symmetric partners.

7.6.3. Non-bonded Terms

The non-bonded terms consist of van der Waals and electrostatic charge terms, that is an atom type and a point charge. For each atom in the target molecule, from the mappings provided by the subgraph isomorphism testing procedure, lists of all atom types and point charges mapped to it are created. The GROMOS force field uses atom types to keep track of the van der Waals parameters assigned to each atom. The atom type is a discrete value and so the atom type of the atom in the target molecule is set to the mode of the list of atom types. Point charges are more complex as they are continuous values. Experimentation showed that the distribution of mapped partial charges on a specific atom tend towards a narrow, unimodal distribution with the mean and median of the data set being approximately equal, when a sufficiently large number of fragments are mapped. As such, the mean of the mapped partial charges is used as the point charge on an atom. As the actual distribution of mapped partial charges is generally hidden from the user, the median, standard deviation, and sample size are provided in the output to allow the user some means of determining the reliability of each value.

Due to the nature of the subgraph isomorphism testing, it is possible for an atom from a molecule in the Athenaeum to be mapped to a single atom in the target condensed molecular graph multiple times. Conceivably, this could result in a situation where the mean of the partial charges is distorted due to the large number of identical partial charges from a single fragment atom. This case is not taken into consideration here, but there are a number of ways that it could be. One method would be in the subgraph isomorphism testing phase. In this case, given a vertex v in multiple fragments of a molecule, if v maps to a vertex in the target condensed molecular graph, all mappings except the largest (or smallest) fragment are discarded. In this way, the fragments mapped will be as large (or small) as possible, thereby reducing the number of multiple maps of a single fragment atom by a reduction in the number of fragments mapped. Another method would be to apply a diminishing returns to the count for each atom with multiple maps. That is, each additional mapping of an atom to a single vertex in the target molecular graph could be given a lesser weight. This could be treated in either a continuous or discrete manner. In the continuous manner, each additional mapping after the initial adds, for instance, some fraction of the partial charge of the atom to the sum of mapped partial charges. Of course, the count of mapped atoms will need to be increased by the same fraction. A discrete manner would increment the effective count of mapped atoms at certain actual count values. For instance, an implementation may require one mapped atom for the first effective count, four for the second, nine for the third and so on. In the limit, this can be seen as only counting each mapped atom once.

7.6.3.1. Charge Group Determination

The GROMOS force field makes use of the charge group concept, therefore it is necessary to arrange atoms into small groups based on the parameters produced by CherryPicker. When the partial charges of a group of atoms add up to exactly zero, the leading term of the electrostatic interaction between two such groups of atoms is of dipolar character. The sum of the $1/r$ monopole contributions of the various atom pairs in the group-group interaction will be zero. Use of charge groups in calculating the electrostatic energy increases performance as fewer calculations are required. Additionally, use of charge groups reduces errors and discontinuities in the energy as atoms move in and out of the cutoff radius. As such, it is desirable that the atoms of a molecule be organised into groups such that the sum of the group partial charges add up to, ideally, zero, or integer values otherwise. In their paper describing an algorithm for partitioning a molecule into charge groups, Canzar *et al.* describe why charge groups should not be too large, and that they should be connected.[10] Namely, the effective cutoff distance of an individual atom in a given charge group is given by the cutoff distance minus the distance to the centre of geometry of the

charge group. If the distance of an atom to the centre of geometry becomes large, the effective cutoff becomes small, leading to an undesirable increase in errors and discontinuities. Imposing the restriction that charge groups should be connected aids in this as connectivity imposes spatial proximity. A similar approach to the algorithm described by Canzar *et al.* is undertaken here to determine an optimal charge group assignment.

The algorithm described here is designed to subdivide a graph G into N connected subgraphs which minimises some penalty function. The subgraphs of G are all disjoint. That is, given any pair of subgraphs $G_1 = (V_1, E_1)$ and $G_2 = (V_2, E_2)$, $V_1 \cap V_2 = \emptyset$. The penalty function is calculated for individual subgraphs, and summed together to give an overall penalty for the subdivision. Following the condition that charge groups should not be too large, the subgraphs have a soft maximum weight w. The weight of a vertex is the number of vertices contracted into a vertex, including the vertex itself, after a leaf contraction of the molecular graph is undertaken. A leaf contraction is a contraction of leaf vertices, i.e. those vertices with degree 1, into their parent vertex. This is primarily undertaken so as to ensure that hydrogen atoms are never in their own charge group. The soft nature of w means that a given a graph with maximum degree of 4, a subgraph will have an upper weight limit of $w+3$. That is, during the optimisation process, a subgraph with total weight $w-1$ could have a vertex with weight 4 (the vertex plus three leaves contracted into it) added to it, but a subgraph with total weight w will not have any more vertices added. The penalty function used to determine optimal charge group partitioning is:

$$\rho(G) = \sum_{g \in S(G)} |\sum_{i \in g} q_i - \sum_{i \in g} \mathrm{FC}(i)| \tag{7.1}$$

where $\rho(G)$ is the total penalty score for a subdivision $S(G)$ of graph G, g is a subgraph of G, q_i is the partial atomic charge on atom i and $\mathrm{FC}(i)$ is the formal charge of atom i as determined by the algorithm described in chapter 9. As part of the leaf contraction, the formal charge and partial charge of leaves that are contracted are added to those of their parent vertex.

Let $G = (V, E)$ be a graph, with subgraphs $G_1 = G[V']$ and $G_2 = G[V \setminus V']$. It should be obvious that the minimum penalty charge group partition of G, $\mathrm{CGP}(G)$ can be given as

$$\min(\mathrm{CGP}(G)) = \min(\mathrm{CGP}(G_1)) + \min(\mathrm{CGP}(G_2)). \tag{7.2}$$

That is, a graph can be split into two subgraphs and the minimum penalty charge group partitioning of the graph can be obtained from the minimum penalty charge group partitioning of the two subgraphs. This fact lends itself to solving the optimisation problem using dynamic programming. By applying this fact recursively to the subgraphs G_1 and G_2, the optimisation problem can be solved in a top down manner utilising memoisation. The algorithm to perform

Algorithm 3 Generate the optimal charge group partitioning of a graph

```
1:  procedure ChargeGroupPartition(G)
2:      function CGP(B)
3:          if B ≡ ∅ then                          ▷ terminate recursion when bag is empty
4:              return [], 0
5:          else if B has been seen then
6:              return seen_subgraphs[B]
7:          end if
8:          v ← min(B)                                          ▷ choose a vertex from B
9:          for all G' = (V', E') ∈ ConnectSub(B \ {v}, {v}, N(v)) do
10:             best_part, best_pen ← CGP(B \ V')
11:             g_pen ← ρ(G')
12:             if best_pen + g_pen < seen_subgraphs[B] then
13:                 seen_subgraphs[B] ← [V'] + best_part, best_pen + g_pen
14:             end if
15:         end for
16:         return seen_subgraphs[B]
17:     end function
18:     seen_subgraphs = dict()                ▷ associative array to store seen subgraphs
19:     return CGP(V)
20: end procedure
```

this is given in algorithm 3. Though this algorithm is guaranteed to give an optimal charge group partition, the optimal charge group partition is not guaranteed to contain only charge groups with integer values.

To further improve the performance of the algorithm, an upper limit on the size of a graph to subdivide can be provided. Then, if a graph is larger than this size, it is recursively split into approximately even sized subgraphs until the subgraph size is less than this upper limit. In such a case, the charge group partition will not necessarily be optimal, but assuming the upper limit is much larger than w, it will be a good approximation.

7.6.3.2. Charge Redistribution

Adding partial atomic charges from different sources or, as is the case here, adding means of lists of partial atomic charges is likely to result in non-integer total molecular charge once all atoms in the target molecule have been processed. There are two ways that this could be handled, a global scaling of all partial atomic charges, or more localised scaling of specific partial atomic

charges. A global scaling method would scale each partial atomic charge by the value:

$$q_{\text{scaled}} = \frac{q_{\text{raw}} \cdot Q}{\sum_{i \in G} q_i} \tag{7.3}$$

where q_{scaled} is the scaled partial atomic charge of an atom, q_{raw} is the initial partial atomic charge of an atom, Q is the target molecular charge and $\sum_{i \in G} q_i$ is the total molecular charge as given by the sum of initial partial atomic charges of all atoms. Because partial atomic charges are given to only a few decimal places (between three and five), when there is a small difference between the target molecular charge and the initial total molecular charge, the scaling of each individual partial charge may not be sufficient for changes to be noticeable within the number of decimal points output. As such, a more localised charge redistribution scheme is utilised here.

The charge redistribution scheme developed here applies a number of rules to individual atoms and small groups of atoms. These rules generate a matrix equation of the form $\mathbf{A}x = b$ where x is the partial charge changes that need to be applied to each atom, which can then be solved for x using linear least squares. Each atom in the target molecule has a column in A, and the rows are the results of applying the rules to the molecule. If a certain rule does not apply for an atom, it is given a value of `0` in that row, otherwise the value is that determined by the rule. The rules used are formalised in equations 7.4 to 7.11 and are described below.

$$q_{\text{req}} = Q - \sum_{i \in G} q_i \tag{7.4}$$

$$q_{\text{CG}_f} - q_{\text{CG}_0} = 0 \quad \text{if } q_{\text{CG}_0} \in \mathbb{Z} \tag{7.5}$$

$$\mathfrak{p}(\text{CG}_f) = \mathfrak{p}(\text{CG}_0) \tag{7.6}$$

$$\Delta q_i = 0 \quad \text{if } i \text{ is pre-mapped} \tag{7.7}$$

$$\Delta q_i = 0 \quad \text{if } q_i = 0 \tag{7.8}$$

$$\Delta q_i = 0 \quad \text{if } (\chi_i - \max_{n \in N(i)} (\chi_n) \geq 0.5) \wedge (q_{\text{req}} < 0) \tag{7.9}$$

$$\Delta q_i = 0 \quad \text{if } (\chi_i - \max_{n \in N(i)} (\chi_n) \leq 0.5) \wedge (q_{\text{req}} > 0) \tag{7.10}$$

$$\Delta q_i = \Delta q_j \quad \text{if } j \in \text{symm}(i) \tag{7.11}$$

The rules can be divided up into global rules, charge group rules, and atomic rules.

Global Rules Possibly the most important rule is that the amount of additional charge that should be added to the molecule, q_{req}, needs to be equal to the difference between the target molecular charge, Q, and the total initial charge $\sum q_i$, as given by equation 7.4. If this rule is not met, the entire reason for redistributing charge becomes void and the universe is sucked into a

black hole.

Charge Group Rules An initial charge group partitioning can be performed using the initial partial atomic charges. The charge groups given by this partitioning can then be used to apply charge group based rules. Two such rules are developed, equations 7.5 and 7.6, though only the first is generally used.

Equation 7.5 shows that if a given charge group has an integer total charge, after the redistribution of charge on the molecule, it should retain its integer charge. This rule does not preclude the addition of charge to atoms within this charge group as a small partial charge added to one atom can be balanced by subtracting the same magnitude partial charge from another in the group.

Equation 7.6 shows that the dipole of a charge group, $\mathfrak{p}(\mathrm{CG})$, should remain constant with charge redistribution. As a dipole is conformationally dependent, this rule is not generally utilised.

Atomic Rules Atomic rules apply to individual or pairs of vertices of the molecular graph. Equation 7.7 shows that if a vertex i has been mapped using the first Athenaeum, its charge, q_i, should remain constant through charge redistribution. This rule follows from the spirit of the first Athenaeum as a source of well tested parameters. Allowing charge to be added to such a pre-mapped vertex would mean that the parameters would no longer match those of the source fragments.

Equation 7.8 shows that atoms with zero total partial charge should not have additional charge added to them. Generally this would be unnecessary as an explicit rule as any changes would be minor. However, it is provided for as sometimes it may be desired.

Equations 7.9 and 7.10 describe the situation when an atom is involved in a polar bond, defined as a bond where the magnitude of the difference in electronegativities, χ, of the two atoms making up the bond is larger than 0.5. If a total positive additional charge is required, additional charge is only added to the least electronegative atom of a polar bond, and if a total negative additional charge is required, additional charge is only added to the most electronegative atom of a polar bond.

The final rule is given by equation 7.11 which shows that if two vertices are determined to be symmetric partners of one another, the total charge added to one of the vertices must be equal to the total charge added to the other. This ensures that the symmetry of the molecule is maintained.

Finalising Charges and Charge Groups The matrix formed by applying the rules described above is solved using linear least squares. Once solved, this gives the amount of additional

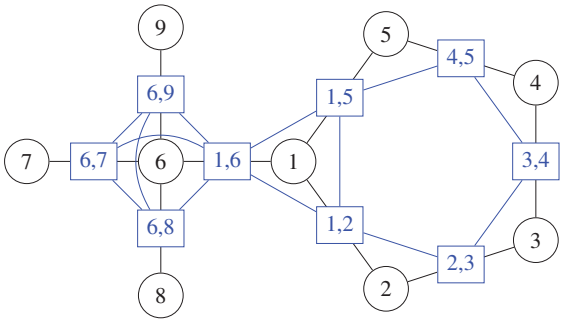


Figure 7.13: Example of a line graph. The graph G is drawn in black with the line graph $L(G)$ superimposed in blue.

charge that needs to be added to each atom in the molecule. Due to rounding errors, adding these additional partial charges to their atoms may result in very minor differences between the target charge and the total charge. To deal with this, any remaining charge once additional charge has been added and the new partial charges rounded to the correct number of decimal points is added to either the single atom with largest signed partial charge, or the set of symmetric partners. In this way, the symmetry of the molecule is retained. With final charges determined, the charge groups can be recalculated.

7.6.4. Bond and Angle Terms

Determining parameters for the bond and angle terms is a fairly straight forward process. Each fragment that is mapped to the target condensed molecular graph is evaporated and the mapping translated to the target molecular graph. The edges of the fragment molecular graph define the bonds which have been mapped to the target molecular graph. The set of angles that have been mapped can be determined by utilising the line graph of a graph:

Definition 7.18. Let $G = (V, E)$ be a graph. The *line graph* $L(G)$ of G is the graph on E in which $x, y \in E$ are adjacent as vertices if and only if they are adjacent as edges in G. An example of a line graph is given in figure 7.13.

It should be obvious that the edges of a line graph represent the angles of the molecule, and, by definition, the vertices of a line graph represent the bonds of the molecule. As such the line graph can be used to identify both bonds and angles. For ease of description, the molecular graph is used to identify bonds within a molecule, and the line graph of the molecular graph is used to identify angles.

The list of all mapped bond types for a given bond is constructed as follows. An edge is considered to be mapped by a fragment if each vertex of the edge is mapped by the fragment.

By definition, the fragment molecular graph of a mapped fragment has been entirely mapped to the target molecular graph. Thus, every edge in the fragment molecular graph maps to a bond in the target molecule. For each bond in the target molecule, all mapped bond types are compiled into the list. As bond types are discrete values, the bond type of the target molecule's bond is set to the mode of this list. A similar process is used for determining the angle types of all mapped angles in the target molecule, except the edges of the line graph of the fragment molecular graph are used to identify the mapped angles.

7.6.5. Proper and Improper Dihedral Terms

The four-body bonded terms are treated separately to the bond and angle terms. Theoretically, they can be determined in a similar manner to that of the bond and angle terms, however there are some practical considerations which mean it is simpler to treat them independently. Improper dihedrals are highly dependent on the order of atoms in their definition. This makes it difficult to determine the correct order of atoms for their definition based solely on the fragments that are mapped to the atoms. This becomes more apparent if the stereo chemistry of an atom is not included as part of the subgraph isomorphism testing. However, they are fairly simple to identify based on the input coordinates, assuming that although the input coordinates may not be ideal, they should not be unreasonable. Treating proper dihedrals independently from the bond and angle terms means that in future it is easy to extend their determination by calculating them in a manner such as that described in part I.

7.6.5.1. Improper Dihedrals

An improper dihedral is generally used to keep a three-coordinate atom in a planar state, or in the case of united atoms, to keep a tetrahedral carbon with one hydrogen in the correct tetrahedral conformation. In the GROMOS force field, improper dihedrals are also used to keep aromatic rings planar. For the purposes of identifying positions for improper dihedrals to be placed, a dihedral $ijkl$ is defined as being planar if the angle between the planes ijk and jkl is either 0 or $180 \pm \texttt{TOL}°$. `TOL` is the angular tolerance for planarity and is set to 5° by default. Based on these use cases, a planar improper dihedral (i.e. an improper dihedral with $\xi_0 = 0$ or 180°) is added in the following situations: if a carbon or nitrogen atom a has three neighbours i, j, k and the dihedral $aijk$ is planar, if the proper dihedral $ijkl$ is in an aromatic ring, and if the bond jk is a carbon-carbon double bond a planar improper is added to $ijkl$. A tetrahedral improper dihedral is added to any carbon atom a with three non-hydrogen neighbours i, j, k and one hydrogen neighbour.

7.6.5.2. Proper Dihedrals

Proper dihedrals are parameterised based on subgraph isomorphism mapping to the dihedral fragments of an Athenaeum's Vault. All dihedral fragments of the target molecule are extracted and mapped with those in the Vault. When a match is found, the parameters for the dihedral in the target molecule are set to the mode of the list of possible parameters stored with the dihedral fragment in the Vault.

7.6.6. Unmapped regions

In case a target molecule is unable to be mapped completely by the fragments of the Athenaeum, the relevant unmapped regions are left blank in the output parameter file, and a warning message is issued to the user. In future, unmapped regions could be extracted and parameterised through the ATB[9–11] before being incorporated into the target molecule.

8. Testing and Discussion

This chapter discusses the outcomes of testing the CherryPicker algorithm described in the previous chapter. Various choices that need to be made, such as the amount of fragment overlap and the minimum fragment size, are investigated to determine sensible values. All the tests were undertaken using two sets of molecules. The first set are source molecules from which an Athenaeum is built, containing 9064 small molecules, representing all available molecules at the time, obtained from the ATB parameterised using its QM2 level and is hereafter referred to as SRC9064. The molecule ID's are given in table F.1. Chosen due to availability of NMR reference data, the second set of molecules is of 23 larger molecules to be parameterised and is hereafter referred to as CPT023. Structures for these molecules are given in figure 8.1.

8.1. Algorithm Optimisation

The CherryPicker algorithm contains a number of internal parameters and choices which need to be optimised or made. In lieu of a full multi-dimensional optimisation scheme, which would be infeasible, a discussion of the choices made for some of the parameters is given below.

8.1.1. Charge Group Partitioning

As outlined in section 7.6.3.1, the charge group determination algorithm implemented here has a *w* parameter to provide a soft upper limit for the size of charge groups. The ATB utilises a more in depth and robust algorithm for determining charge groups which has undergone heavy testing. As a means to determine an optimal value of *w* to utilise within CherryPicker, charge groups obtained using various values of *w* are compared to charge groups obtained by the ATB. The molecules in SRC9064 had charge groups calculated with *w* values ranging from 1 to 10. Three metrics were measured for each charge group obtained: the number of atoms in the charge group, the charge of the charge group, and the diameter of the sphere which would just encompass all the atoms of the charge group. These metrics were also calculated for the charge groups assigned by the ATB. An optimal *w* value would then provide the best overall reproduction of the metrics.

Figure 8.2 shows the distributions of charge group sizes, i.e. the number of atoms in the charge group, for each *w* value. When *w* is less than 4, charge group sizes tend towards much smaller

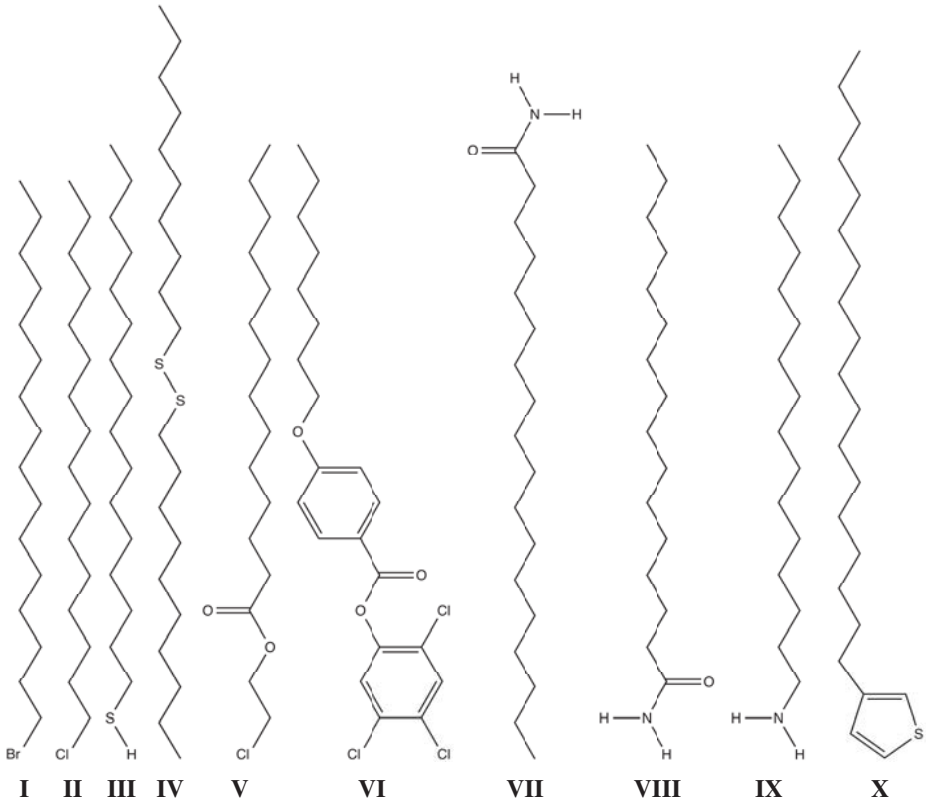


Figure 8.1: Structural diagrams of the 23 molecules within CPT023 used to test the CherryPicker parameterisation algorithm. The molecules are: (I) 1-bromohexadecane, SDBS No.: 1743, ATB MolID: 193517; (II) 1-chlorohexadecane, SDBS No.: 5998, ATB MolID: 193519; (III) 1-hexadecanethiol, SDBS No.: 7774, ATB MolID: 193577; (IV) didecyl disulfide, SDBS No.: 7589, ATB MolID: 193574; (V) 2-chloroethyl tetradecanoate, SDBS No.: 5009, ATB MolID: 193544; (VI) 2,4,5-trichlorophenyl *p*-(octyloxy)benzoate, SDBS No.: 19281, ATB MolID: 193539; (VII) stearamide, SDBS No.: 7918, ATB MolID: 193566; (VIII) palmitamide, SDBS No.: 7746, ATB MolID: 193569; (IX) hexadecylamine, SDBS No.: 1602, ATB MolID: 193552; (X) 3-octadecylthiophene, SDBS No.: 19372, ATB MolID: 193576; (XI) tetrabutylurea, SDBS No.: 22628, ATB MolID: 193564; (XII) 4-bromo-4-heptylbipheny, SDBS No.: 18679, ATB MolID: 194901; (XIII) tris(2-ethylhexyl)amine, SDBS No.: 18489, ATB MolID: 193545; (XIV) ditetradecyl sulfide, SDBS No.: 7756, ATB MolID: 193634; (XV) 2,2′(*o*-phenylenebis(methylenethio))dinaphthalene, SDBS No.: 16336, ATB MolID: 224564; (XVI) 7-tetradecanol, SDBS No.: 41018, ATB MolID: 193571; (XVII) *p*-(dodecyloxy)nitrobenzene, SDBS No.: 9469, ATB MolID: 193558; (XVIII) retinol, SDBS No.: 22561, ATB MolID: 193555; (XIX) N-lauroyl-N-methylglycine, SDBS No.: 15483, ATB MolID: 193567; (XX) 1,3-bis(1-(2-hydroxyethyl)-4-piperidyl)propane, SDBS No.: 7827, ATB MolID: 193560; (XXI) N,N,N′,N″-tetrabutyldiethylenetriamine, SDBS No.: 41223, ATB MolID: 193547; (XXII) N-dodecylaniline, SDBS No.: 7904, ATB MolID: 193548; (XXIII) N-(4-hydroxy-3-methoxybenzyl)nonanamide, SDBS No.: 53218, ATB MolID: 193562.

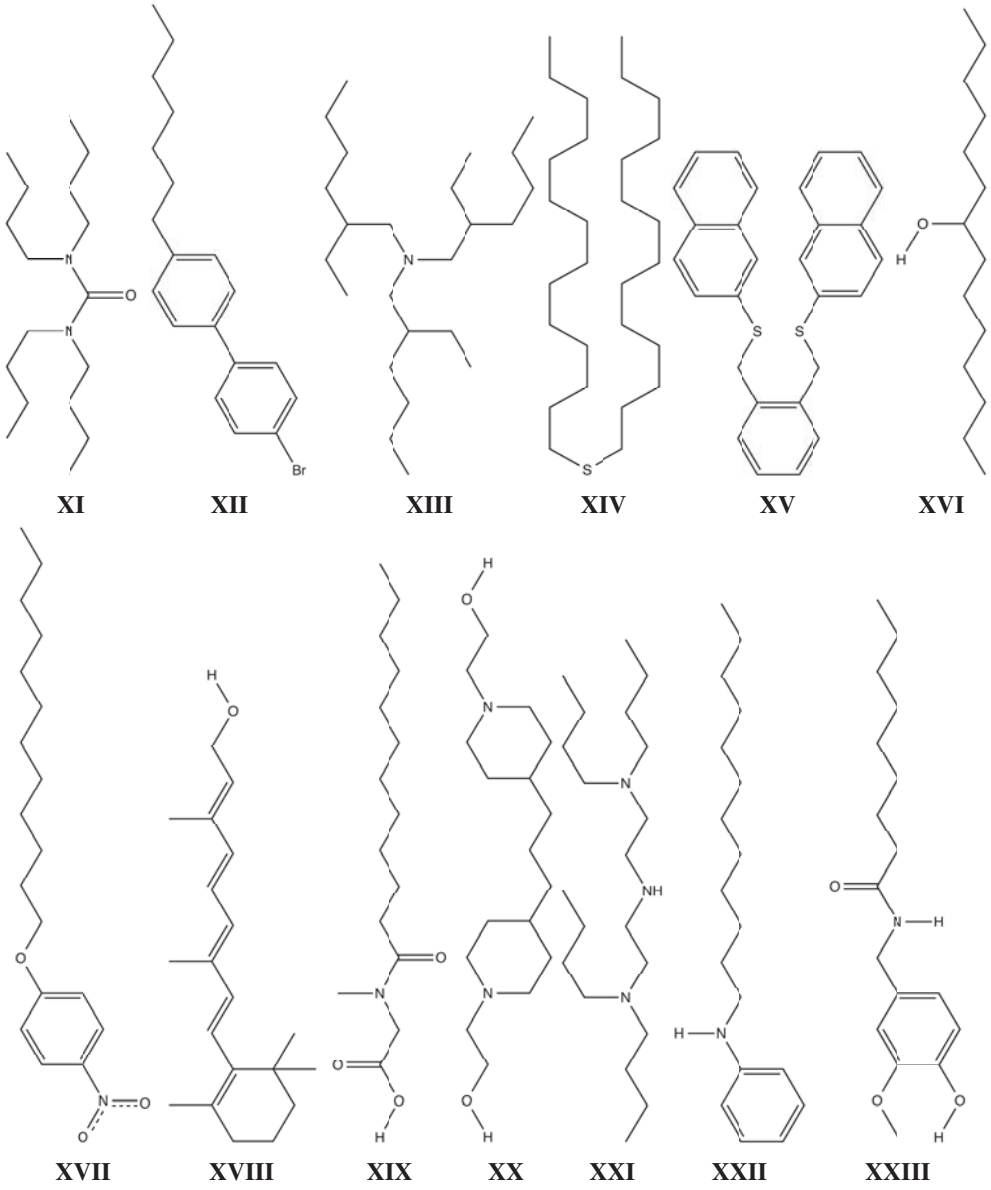


Figure 8.1 ***continued***

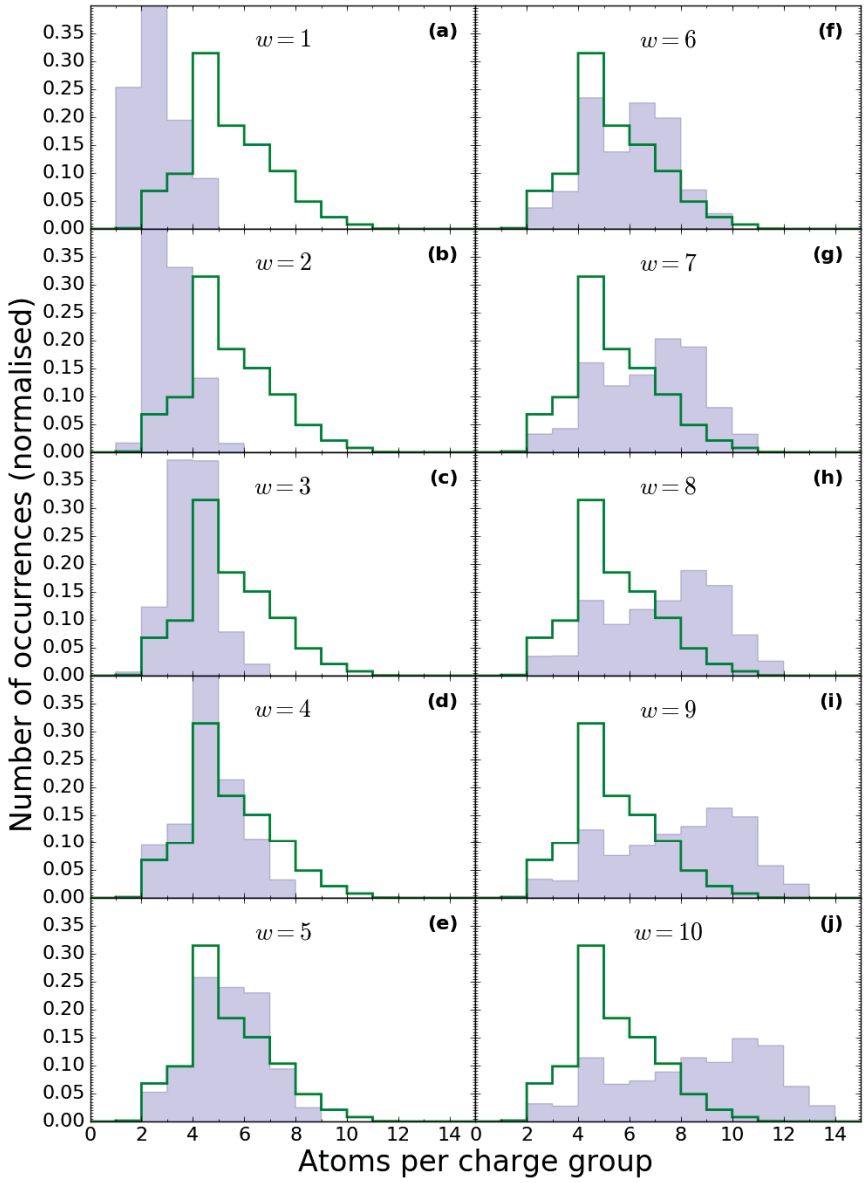


Figure 8.2: Distributions of the charge group size, the number of atoms in a charge group, as determined with w values from (a) 1 to (j) 10. The filled blue histograms show the size distributions produced for each w value and the green outlines show the reference ATB distribution. The axes are scaled to fit the reference values vertically and the largest calculated value horizontally.

sizes than the reference sizes. Values of between 4 and 6 do a reasonable job of reproducing the reference distribution as they show similar normalised counts to the reference distribution across a broad range of group sizes, whereas values larger than 6 result in a much broader size

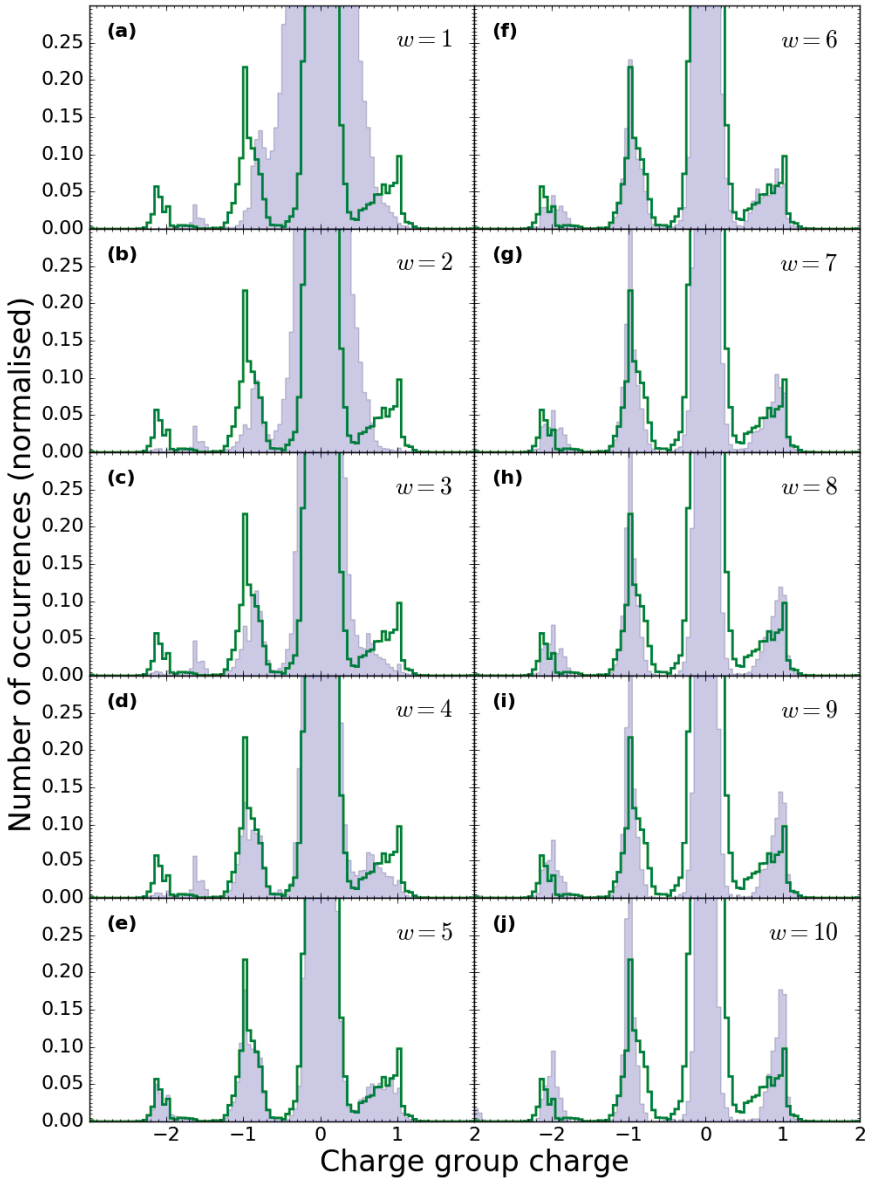


Figure 8.3: Distributions of the charge of charge groups as determined with w values from (a) 1 to (j) 10. The filled blue histograms show the charge group charge distributions produced by each w value, and the green outlines show the reference ATB distribution. The axes are scaled to show peaks in the distribution at -2, -1 and 1 which would otherwise be overwhelmed by the large peak at 0.

distribution. These results are not entirely unexpected. A large w limit would tend to favour large charge groups as they can more easily obtain integer charges.

The increased ability to obtain charge groups with integer charges as w increases is evident in

figure 8.3. Low w values show a broad peak around a charge group charge of 0 with peaks at -2, -1 and 1 not all being distinguishable until $w = 4$. With $w = 5$ or 6, the calculated distributions match the ATB reference distribution very well, though $w = 6$ has slightly narrower peaks and the peak at -2 is shifted slightly towards 0 relative to the corresponding reference peak. Larger values of w result in narrower peaks than the ATB reference as charge groups tend more towards the expected integer charges.

Even though sharper peaks would be desired in general, as they indicate charge groups closer to the optimum integer charge, they come at a cost of increased charge group diameter, as shown in figure 8.4. Generally speaking, the ATB charge group diameter distribution has a sharp drop off at the upper size limit, whereas the distributions produced by the method described in algorithm 3 tend to gradually reduce. This is due to the ATB algorithm including an explicit upper limit to the diameter of a charge group as part of the optimisation method, whereas the algorithm presented here only limits the size of charge groups. The lower limits are identical due to natural constraints imposed by bonded atoms. With a w value less than or equal to 3, the calculated distributions have 99.9% of the distribution below the reference upper limit. With $w = 4$, this drops to 95%, $w = 5$ is 84.6% and it continues to rapidly decrease until $w = 10$ has only 40.7% of charge group diameters below the reference upper limit.

Taking all three metrics into consideration, a value of $w = 5$ provides the best reproduction of the ATB reference charge groups. The charge group size distribution has a slightly larger proportion of medium sized charge groups, and less of the larger sizes. The charge group charges distribution is the best match to the reference overall, narrowly better than $w = 6$ due to the slightly wider peaks, and the location of the peak at -2, and the diameter distribution matches the reference at low diameters before petering off past the upper limit. As such, all parameterisations discussed here were performed with a $w = 5$ value.

8.1.2. Fragment Generation

How fragments are generated can have an effect on the ability of a given Athenaeum to parameterise certain molecules. Namely, situations may arise whereby even though all atoms of a molecule are mapped to at least one fragment in the Athenaeum, some bonds and angles do not have corresponding mapped bonds or angles in any fragments of the Athenaeum. Due to the rules introduced in section 7.4.1.2 for checking the validity of fragments containing cycles or portions of cycles, it is possible for rotatable bonds between two cycles to not have any mapped fragments unless both cycles are fully contained within one fragment. Such a case occurs with molecule XII. As a means to work around this limitation, an additional `FULLY_FRAGMENT` switch was created which removes the cyclic rule when generating an Athenaeum. This allows for par-

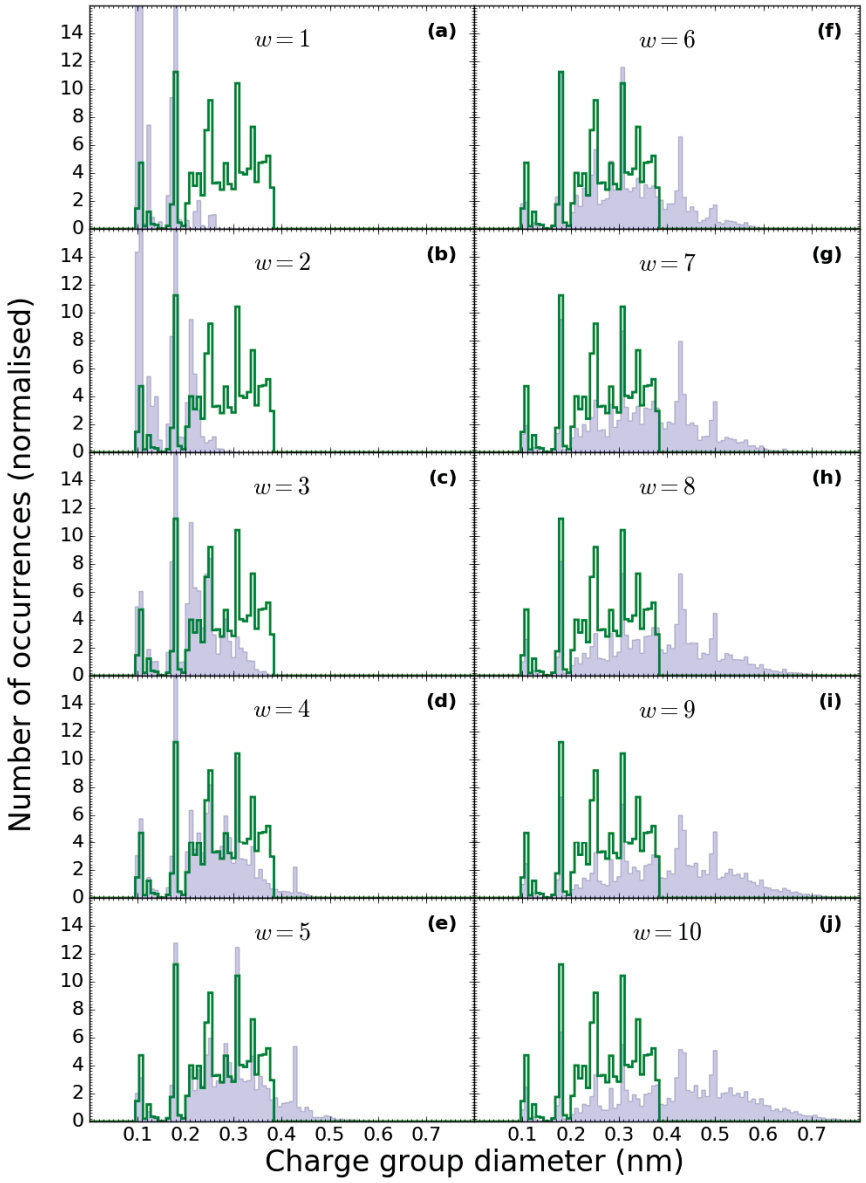


Figure 8.4: Distributions of the diameter of charge groups as determined with w values from (a) 1 to (j) 10. The filled blue histograms show the charge group diameter distribution produced for each w value and the green outlines show the reference ATB distribution. The axes are scaled to fit the reference values vertically and the largest calculated value horizontally. Single atom charge groups are excluded due to their diameter being zero.

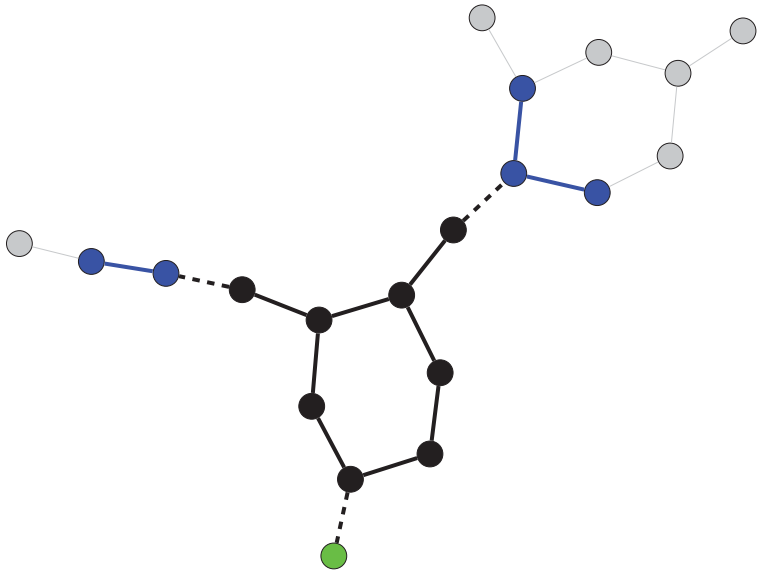


Figure 8.5: Edge terminated version of the fragment in figure 7.11. The subgraph under consideration as a fragment is given in black. The blue vertices and edges are the overlap region from edges not incident to a cycle, with $k = 2$. The green vertex and edge are the overlap region from edges incident to a cycle, with $k_c = 1$. Vertices and edges not part of the fragment and overlap region are in grey.Dashed edges are the terminal edges of the fragment.

tial cycles which will provide parameters for the rotatable bonds. A mid-point between following the validity rules of cycles laid out in section 7.4.1.2 and allowing full fragmentation of cycles would be to only allow full fragmentation of cycles when there is at least one complete cycle in the fragment.

Another potential means of dealing with this issue is through modifying the way in which the relationship between fragments and their overlap region is described. Instead of being vertex terminated at the boundary with their overlap region, fragments could be edge terminated. An example of such a fragment is given in figure 8.5. Effectively, this would mean that bonded parameters associated with the edges between a fragment and its overlap region are kept as part of the fragment mapping. The parameter inheritance scheme described in section 7.6 requires that for a given bonded term, all vertices between which the term is defined must be contained within the fragment. By switching to an edge terminated fragment description, the requirement would change to allowing one of the vertices to be in the overlap region. Of course, the vertex in the overlap region would not contribute to the non-bonded parameters. For the tests undertaken here, this method has not been implemented, with the issue dealt with through allowing full cycle fragmentation.

8.1.3. Fragment Size

The size of a fragment can be described as either a strict count on the number of atoms in the fragment, or as a determination of the molecular mass of a fragment. The strict counting scheme is used here as a count scheme is more desirable due to its inherit ability to ensure all fragments contain bonded terms. Limits to fragment sizes come in two forms: a lower size limit and an upper size limit. With the use of a counting scheme, a lower size limit is effectively enforced by the requirement for bonded parameters to be mapped. Though dihedral terms are mapped independently of the Athenaeum fragments, it may be desirable for the fragments to be large enough, on average, to define a dihedral term. Effectively, this means each fragment should have a path between two atoms with a length of at least four. Excluding hydrogen, the atoms of molecules within SRC9064 have an average degree of 2.889. Rounding this up to 3, it can be seen that, on average, the smallest fragment containing a given atom with a path length of at least four would require a minimum size of five. Of course, this value can be increased as desired; five is just the absolute smallest recommended fragment size.

There is no obvious driving force towards providing an upper fragment size limit. A given Athenaeum has an implicit upper size limit based on a combination of source molecule sizes and overlap length. The upper size limit would be redundant if it was larger than the size of the largest source molecule, and the overlap length reduces this limit as any fragments must have an overlap. As there is no fundamental reason for a fragment size upper limit, tests presented have no upper size limit enforced.

8.1.4. Overlap

The length of the overlap used when generating an Athenaeum plays a major role in the ability of an Athenaeum to map to a target molecule, and the reliability of the parameters produced in the end. To gain insight into the effect of overlap length on the applicability of parameters produced by the CherryPicker algorithm, both the bonded and non-bonded terms are independently investigated.

8.1.4.1. Bonded Terms

To investigate the applicability of bonded terms, four Athenaeums with different overlap lengths were generated using the SRC9064 molecules. Full fragmentation of cycles was allowed in all cases. Overlap lengths used were between zero and three inclusive. Each Athenaeum was mapped to all molecules in CPT023, as well as to an additional ten large molecules*, with a

*cholesteryl chloroformate, *cis*-13-docosenoyl chloride, 9*Z*,12*Z*-octadecadienoyl chloride, *N*,*N*,*N*′,*N*′-tetrakis(*p*-tolyl)benzidine, heptadecanenitrile, pyridoxine 3,4-dipalmitate, folic acid, methacrylic acid 11-(4-(4-

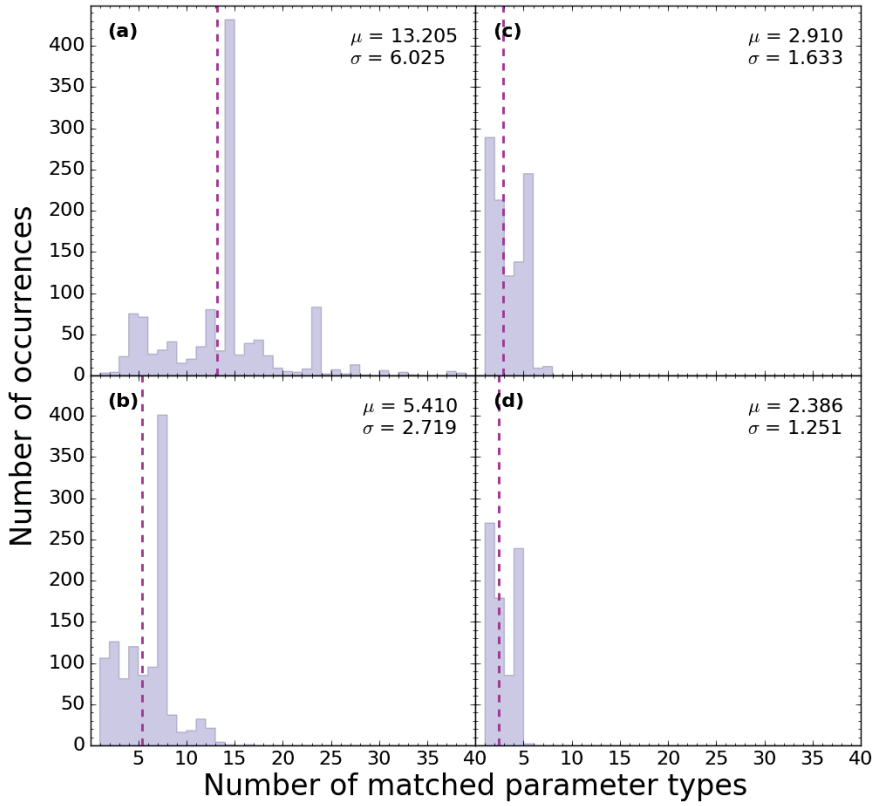


Figure 8.6: Distributions of the number of potential bond parameters mapped to all molecules within CPT023 using Athenaeums with overlap lengths of between (a) 0 and (d) 3. Athenaeums were generated using SRC9064 with full cyclic fragmentation enabled, and the mapped UA parameters counted. The vertical magenta line shows the mean number of matched parameter types for each Athenaeum. Numerical values for the mean (μ) and standard deviation (σ) of the distributions are provided.

bit-mask of 4261937151[†] for vertex mapping and 28[‡] for edge mapping, and a lower fragment size limit of five. The number of potential bond and angle parameters for each bond and angle were then counted and distributions plotted. These distributions are shown in figure 8.6 for the bond parameters and figure 8.7 for the angle parameters.

Both of these figures show the same trends, though the statistics of bond parameter distributions are always noticeably lower than the corresponding angle parameter distribution. With an overlap of zero, the distribution is very broad with a high mean value. With an overlap of one,

butylphenylazo)phenoxy)undecyl ester, pentacontane, and palytoxin. Except for pentacontane, these molecules are not a part of CPT023 as SRC9064 does not produce fragments which can fully map them. Pentacontane is a straight chain alkane, which is well represented in the other molecules, and was used purely for initial development purposes.

[†]all available information except stereo chemistry and vertex degree.

[‡]only bond order.

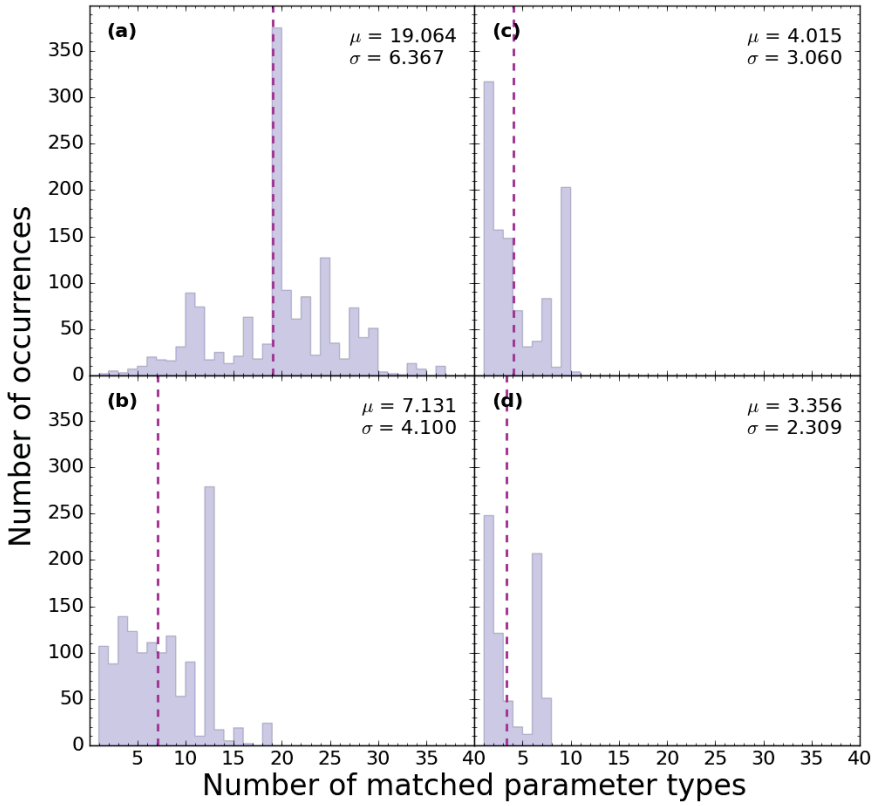


Figure 8.7: Distributions of the number of potential angle parameters mapped to all molecules within CPT023 using Athenaeums with overlap lengths of between (a) 0 and (d) 3. Athenaeums were generated using SRC9064 with full cyclic fragmentation enabled, and the mapped UA parameters counted. The vertical magenta line shows the mean number of matched parameter types for each Athenaeum. Numerical values for the mean (μ) and standard deviation (σ) of the distributions are provided.

the distribution becomes concentrated at lower values, but still has a long tail of high numbers of matched bonded parameters. Overlaps of two and three continue this trend with increased concentration of the distribution at lower values, and a diminishing of the tail. Generally speaking, a large number of potential bonded parameters is undesirable as such a situation indicates that fragments are mapped which poorly match the surrounding environment. An overlap of zero clearly shows this, as there is no accounting for the environment around a fragment in the mapping process. The long tail of the distributions with overlap one show that even though having an overlap of one is a vast improvement on having no overlap, a larger overlap should generally be desired.

8.1.4.2. Non-bonded Terms

To investigate the applicability of non-bonded terms, namely the atomic point charges, a slightly different approach was taken. Instead of generating an Athenaeum and performing mapping to a target molecule, three single atom fragments, with overlap lengths of zero, one, and two, were chosen and mapped to all the molecules in SRC9064. The use of a lower size limit means that a single atom would never be mapped in a general mapping procedure, however it does provide a simple means to investigate the effect of increasing overlap length. These mappings were performed with only element and formal charge for vertices, and bond order for edges. The atom fragments chosen were: the hydrogen of a methyl group bonded to another carbon; a carbon within an aromatic carbon ring, with a hydrogen substituent only; and the neutral oxygen of a negatively charged carboxylic acid functional group, bonded to another carbon. For each fragment and overlap combination, the point charges for all matching atoms within SRC9064 were plotted in a distribution. These distributions are given in figure 8.8.

With an overlap of zero we see that the methyl hydrogen has a bimodal distribution (figure 8.8a) due to the inability of a fragment with zero overlap to be able to distinguish between polar and non-polar hydrogens. The aromatic carbon has a distribution skewed towards negative partial atomic charges (figure 8.8d), and the carboxylic oxygen has a broad base with a sharp central peak (figure 8.8g). All distributions have a reasonably large difference between their mean and median. With an overlap of one, the methyl hydrogen distribution becomes unimodal as polar and non-polar hydrogen atoms are now able to be distinguished, however the carboxylic oxygen distribution becomes bimodal. This bimodality comes about due to being unable to distinguish between oxygens of protonated carboxylic acids and deprotonated carboxylic acids. Additionally, the distributions are narrower and means and medians are a lot closer together. Finally, with an overlap of two, all distributions have a relatively small standard deviation and almost identical mean and median. In combination with the bond parameter results, this indicates that an acceptable overlap length to use is at least two.

8.2. Parameterisation Tests

In order to evaluate the parameter sets returned by the CherryPicker algorithm, several testing methods were devised: a static test to determine how well a QM minimised structure is maintained (section 8.2.1), a dynamic test to determine deviations of bond lengths and angles from a QM minimised structure (section 8.2.2), and a test comparing with experimental NMR data (section 8.2.3).

To perform these tests, five Athenaeums were generated from SRC9064 with overlaps of

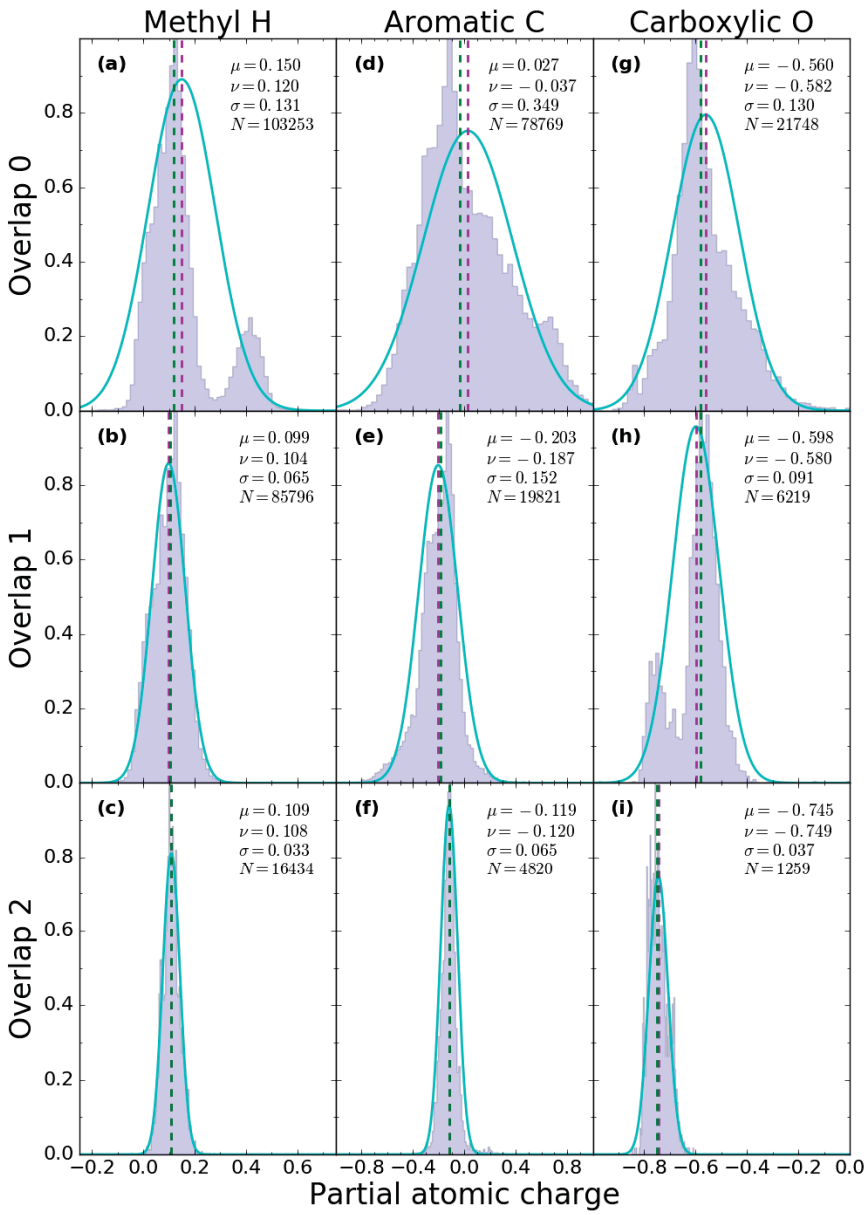


Figure 8.8: Distributions of partial atomic charges mapped to single atom target fragments with varying degrees of overlap. The central atom of each fragment was (a) to (c) a methyl hydrogen, (d) to (f) an aromatic carbon and (g) to (i) a carboxylic acid oxygen. Each charge distribution obtained by mapping the single atom fragment to the molecules in SRC9064 is plotted as a blue histogram, with the cyan curves being the gaussian formed with the distributions' mean (μ) and standard deviation (σ). Vertical lines are given in magenta for the mean, and green for the median (ν) of the distribution. The counts (y-axis) have been normalised so that the largest count becomes 1.0, with the actual counts (N) given.

Table 8.1: Molecules successfully parameterised by each Athenaeum.

Athenaeum	Parameterised molecules
OLP0	I, II, III, IV, V, VI, VII, VIII, IX, X, XI, XII, XIII, XIV, XV, XVI, XVII, XVIII, XIX, XX, XXI, XXII, XXIII
OLP1	I, II, III, IV, VII, VIII, IX, X, XII, XIV, XVI, XVII, XX, XXI, XXII, XXIII
OLP2	I, III, VII, VIII, IX, X, XVI, XXII
OLP3	I, III, VII, VIII, IX, XVI
OLP4	I, III, VII, VIII, IX, XVI

between zero and four (OLP0 to OLP4) inclusive. Full cyclic fragmentation was allowed in the fragment generation stage. Parameterisations were then performed by mapping each molecule in CPT023 with each Athenaeum. Additionally, each molecule was submitted to the ATB for parameterisation at either its QM0 or QM1 level, depending on the number of atoms in the molecule, so as to provide comparative parameters. Due to the higher level of theory used for generation of the parameters for molecules in SRC9064, it is expected that the CherryPicker parameters will out perform the ATB source parameters.

Most Athenaeums were unable to fully parameterise all of the molecules, due to the set of molecules in SRC9064 being what was available rather than specifically designed for fragment based parameterisation. The molecules each Athenaeum successfully parameterised are given in table 8.1. In all of the tests outlined below, the molecular mechanics calculations were performed using the GROMOS molecular dynamics engine.

8.2.1. Structural Minimisation

A simple test of a parameter set is to determine if, when the structure is energy minimised using the assigned parameter set, the minimised structure is chemically reasonable.

Method To provide a reference, chemically reasonable, structure, each molecule in CPT023 was QM structurally optimised at the B3LYP/6-31G(d) level of theory using GAMESS-US. With the optimised structure as the initial conformation, molecular mechanics energy minimisation was performed using the parameter set assigned from each Athenaeum, and the resulting minimised structure rotated to align with the QM optimised structure. A RMSD between the two conformations was calculated. Results are given in table 8.2. As a comparison, for each

Table 8.2: RMSD between QM optimised structure and molecular mechanics optimised structure. Values are in nm.

	CherryPicker			ATB		
Athenaeum	Min	Mean	Max	Min	Mean	Max
OLP0	0.0069	0.0282	0.1412	0.0091	0.0347	0.1604
OLP1	0.0072	0.0246	0.1136	0.0114	0.0263	0.0550
OLP2	0.0133	0.0233	0.0378	0.0172	0.0258	0.0389
OLP3	0.0131	0.0222	0.0406	0.0172	0.0242	0.0330
OLP4	0.0149	0.0239	0.0450	0.0172	0.0242	0.0330

Athenaeum, the RMSD values obtained when minimising the subset of molecules that was parameterised by that Athenaeum but using the parameters obtained from the ATB are also given.

Results The smaller the RMSD, the closer the molecular mechanics minimised structure matches the reference QM structure. As such, a small RMSD shows that, the parameters assigned using CherryPicker obtain a local minimum conformation which is similar to the QM optimised structure. The results in table 8.2 show that, even with an Athenaeum generated with an overlap of zero or one, the CherryPicker parametrisation algorithm generates parameters which are better at maintaining the QM optimised geometry than the corresponding ATB parameters. Though the sample sizes are small, this result does bode well for the CherryPicker algorithm as it produces parameters which rival the performance of ATB generated parameters at a fraction of the computational cost, especially when the molecule has a large number of atoms. For example, a parametrisation of palytoxin by the ATB takes nearly 13 hours, whereas the CherryPicker algorithm can complete the parametrisation in approximately 10 minutes. The results also reinforce the requirement that an Athenaeum should be generated with an overlap length of at least two. Though OLP0 and OLP1 both have molecules with very low RMSD values, they also have molecules with much larger RMSD values than the Athenaeums with longer overlaps. It is more desirable to use an Athenaeum which produces reasonable parameters for molecules in general, rather than good parameters for a few types of molecules and bad parameters for the rest.

8.2.2. Dynamic Stability

Static stability, such as that given by the structural minimisation, is a useful indicator that parameters are at least reasonable. However, they should also act reasonably during a simulation.

Method To investigate this, seven of the eight molecules successfully parameterised using the OLP2 Athenaeum were simulated for 50 ns, using both the OLP2 parameters and the ATB

parameters. Though molecule III was completely parameterised using the OLP2 Athenaeum, simulations using both that parameter set and the ATB parameter set were unable to be undertaken as there were issues with the SHAKE algorithm applying constraints to some solvent molecules. Each molecule was solvated in a cubic box of chloroform with at least 15 Å between the box walls and the molecule. Initial velocities were generated from a Maxwell distribution at 60 K. The temperature was increased to 300 K in steps of 60 K, with a short 10 ps simulation undertaken at each intermediate step. Temperature was maintained by coupling to a Berendsen thermostat. No pressure coupling was involved. A twin range cut-off scheme was used, with a short range cut-off of 8 Å calculated every time step, and a long range cut-off of 14 Å calculated every five steps along with the pairlist update. Bond length constraints were applied to only the solvent molecules using SHAKE. As such, a time step of 0.5 fs was used. Conformations were printed out every 0.5 ps The resulting simulations are analysed in this and the proceeding section.

Results Here, we look at the variation in bond length and angle over the course of the simulation. For each frame of the simulation, the percentage deviation of every bond and angle from the QM optimised structure was calculated. For each bond and angle, a mean percentage deviation was calculated. An overall molecular mean was calculated from these values. These results are given in table 8.3. Over the course of a simulation, bond lengths and angles are expected to vibrate around an equilibrium value, which is here referenced as the corresponding value in the QM optimised structure. As such, smaller average percentage deviations imply, though do not guarantee, that the assigned parameter sets are appropriate. Use of a percentage deviation, as opposed to absolute deviation, accounts for bonds or angles with weaker force constants having larger absolute deviations as they would generally have longer equilibrium values, which means large absolute deviations are a smaller percentage deviation.

Generally, the results between the OLP2 parameters and the ATB parameters are fairly similar. Occasionally, the ATB results have less variation than the OLP2 results, and vice versa. The mean bond length deviation tends to sit between 1.5 % and 2.0 %. The ATB parameters stray outside these bounds with molecules IX and X. The most notable deviation is found with the OLP2 parameters and molecule XVI. In this case, the mean deviation is noticeably raised by large deviations, of 13.1 % and 12.9 %, in only two bonds, marked in red in figure 8.9. Through analysis of the parameters assigned, it was determined that the reason for these bonds having such large deviations was due to them being assigned relatively weak force constants. This could indicate that either there were inadequate fragments in the Athenaeum, or that those bonds should indeed be weaker than the other carbon–carbon bonds.

Table 8.3: Overall molecular bond length and angle deviations, averaged over all frames of the simulation. Deviations are referenced with respect to the QM optimised structure. All values are percentages.

		Bonds			Angles		
Molecule		Min	Mean	Max	Min	Mean	Max
I	CP	1.4532	1.5628	2.8692	3.0830	3.2931	3.8390
	ATB	1.4476	1.5608	2.8487	3.0702	3.2440	3.3160
VII	CP	1.4314	1.6046	2.9942	2.3457	3.1841	3.6760
	ATB	1.3876	1.8698	2.8113	2.3510	3.1764	4.2427
VIII	CP	1.4429	1.6107	2.7792	2.3633	3.3165	4.2467
	ATB	1.4243	1.8855	2.7780	2.3723	3.1588	3.9031
IX	CP	1.4108	1.5532	2.2518	3.1584	3.6293	5.5426
	ATB	1.4245	2.2423	6.7625	3.1739	3.7342	5.5971
X	CP	1.4444	1.6093	2.3391	1.9167	2.8583	3.3407
	ATB	1.6814	2.1381	4.8270	1.5043	2.9458	3.7322
XVI	CP	1.4547	3.2088	13.0974	2.9574	3.7886	8.3558
	ATB	1.4291	1.9209	6.1950	1.6891	2.9991	3.7464
XXII	CP	1.4329	1.7002	2.5194	1.3195	2.8933	7.6950
	ATB	1.4311	1.6875	2.2573	1.7769	2.6617	3.4603

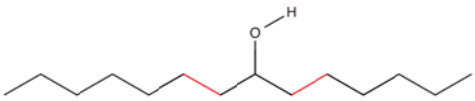


Figure 8.9: Molecule XVI with soft bonds marked in red

8.2.3. Nuclear Magnetic Resonance

Ideally, the parameters assigned to each molecule would be validated through comparison of the thermodynamic properties obtained from a molecular dynamics simulation with experimental data. However, finding experimental data to validate computational results is a difficult proposition. The GROMOS force field was validated using solvation free energy calculations,[45] as was the ATB.[9,11] Such an undertaking could not be achieved here as experimental solvation free energy values tend to only be available for molecules with low molecular weights. NMR experiments are ubiquitous throughout chemistry, particularly within organic chemistry. An NMR spectrum provides a time averaged measure of the chemical shifts of NMR active elements within the molecule. Chemical shifts provide structural information. Thus, an NMR spectrum can be produced from a molecular dynamics simulation by calculating chemical shifts at each frame of the simulation, and averaging across all frames. As such, an alternative validation means, utilising NMR spectra was thus undertaken. Experimental NMR spectra were obtained from the Spectral Database for Organic Compunds (SDBS) which provides a large database of

hydrogen and carbon NMR spectra,[46] and all molecules in CPT023 are present in the database.

Method In order to calculate the NMR spectra, the CHARGE program was utilised,[47] which is a semi-empirical algorithm which determines ^{1}H NMR chemical shifts for a given structural geometry, taking into account a number of short and long range effects. Though CHARGE is capable of determining splitting due to coupling, calculations were performed without this capability enabled due to the reference data not providing coupling constants. Simulations were run in UA form, meaning that in order to calculate shifts correctly, the positions of non-polar hydrogen atoms needed to be generated. For the molecules simulated here, all the hydrogen atoms to generate were on tetrahedral carbons. Three types of carbon atoms required bonded hydrogen atoms to be placed: CH_1, CH_2 and, CH_3. In each case, the hydrogen atoms were placed on the points of a tetrahedron, centred on the central carbon atom and defined by the other substituents, assuming perfect tetrahedral geometry. In the case of CH_3, an initial hydrogen atom was placed randomly with respect to the fourth substituent on the central carbon in order to fix the orientation of the tetrahedron. Hydrogen atoms of the CH_1 group where placed with a bond length of 108.83 pm, while the CH_2 and CH_3 hydrogen atoms were placed with a bond length of 109.45 pm. These bond lengths came from determining the mean of all the similar bond lengths for molecules within SRC9064, which had been QM optimised by the ATB as part of the parameter generation method. Further, a small amount of noise was added to each position, based on the standard deviation of the set of bond lengths, in order to simulate the vibration of the CH bond. To do so, a (r, θ, ψ) triplet was randomly generated. A value for r was selected from a normal distribution with $\mu = 0$ and $\sigma = 2.399$ pm for CH_1 hydrogen atoms, and 1.601 pm for CH_2 and CH_3 hydrogen atoms; θ was selected from the uniform interval $[0, \pi]$; and ψ was selected from the uniform interval $[0, 2\pi)$. The polar vector defined by this random triplet was then added to the previously calculated hydrogen atom position.

For every frame, ^{1}H NMR shifts were calculated using CHARGE. For each atom, the mean shift was determined and plotted as a Lorentz distribution with width at half height of 0.5 Hz. The experimental spectra are given in figure 8.10 and the calculated spectra are given in figures 8.11 to 8.17. Based on the shift values provided with the experimental spectra, reference spectra were also plotted using the same parameters.

Results Both the simulations with ATB and CherryPicker generated parameters tend to replicate the reference spectrum well. The majority of peaks in the 0 to 2 ppm range are due to hydrogen atoms on alkyl chains, with several bonds to the nearest hetero atom. The experimental data had poor separation of these peaks, due to their very similar chemical environments. Large numbers of chemically similar hydrogen atoms were grouped into the same reported shift,

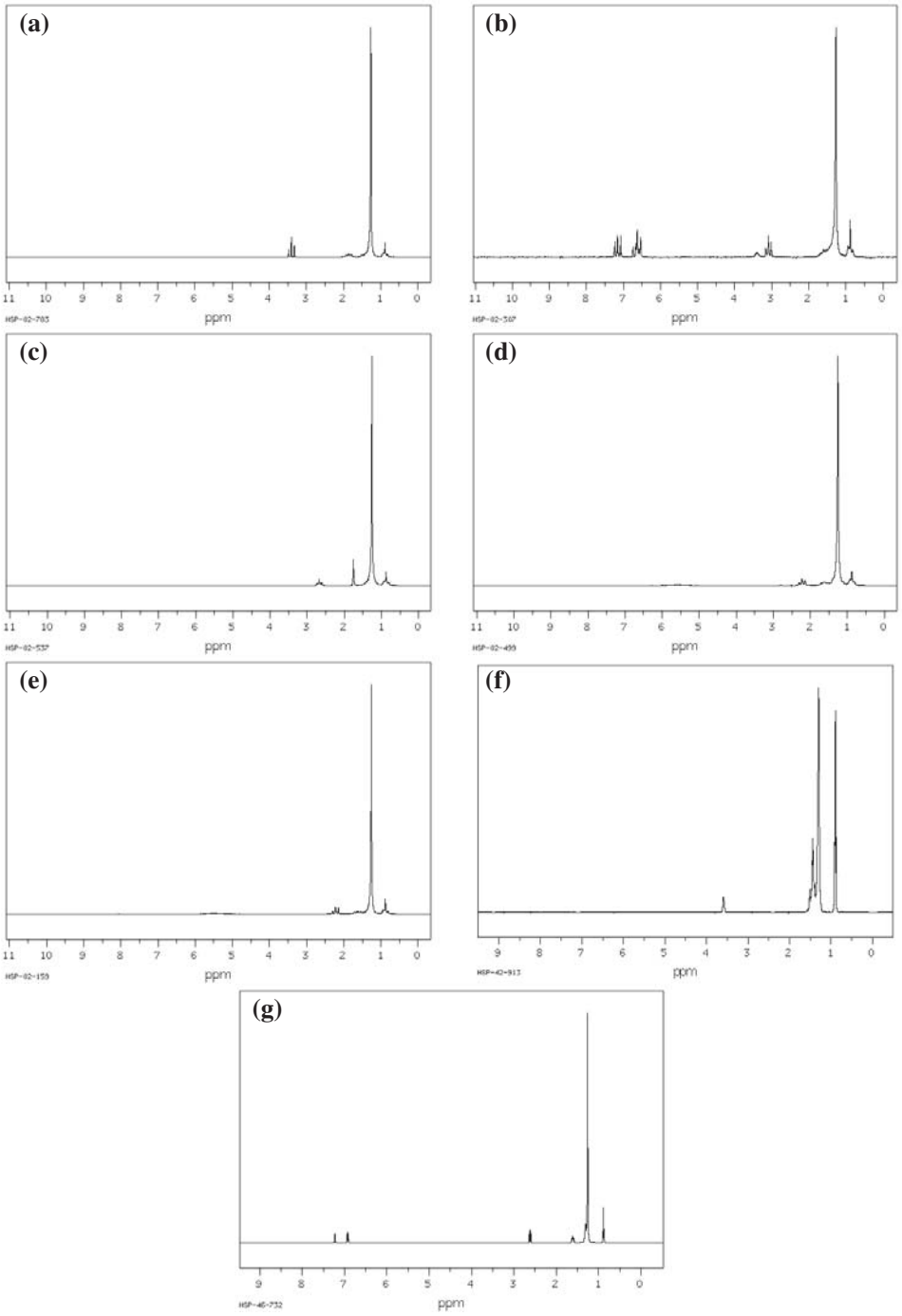


Figure 8.10: Experimental ^{1}H NMR spectra for (a) molecule I, (b) molecule XXII, (c) molecule IX, (d) molecule VII, (e) molecule VIII, (f) molecule XVI and (g) molecule X. Spectra were obtained from the SDBS.[46]

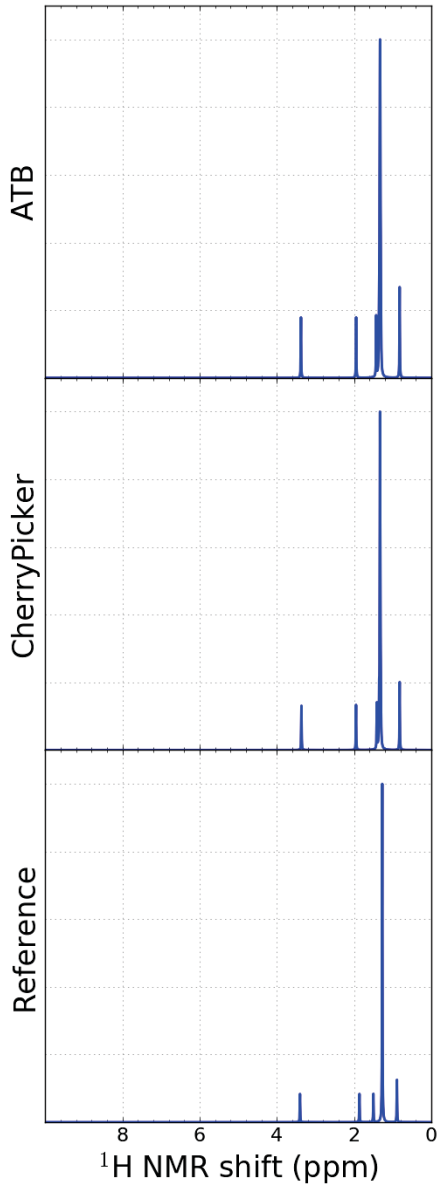


Figure 8.11: Calculated ^{1}H NMR spectra of molecule I.

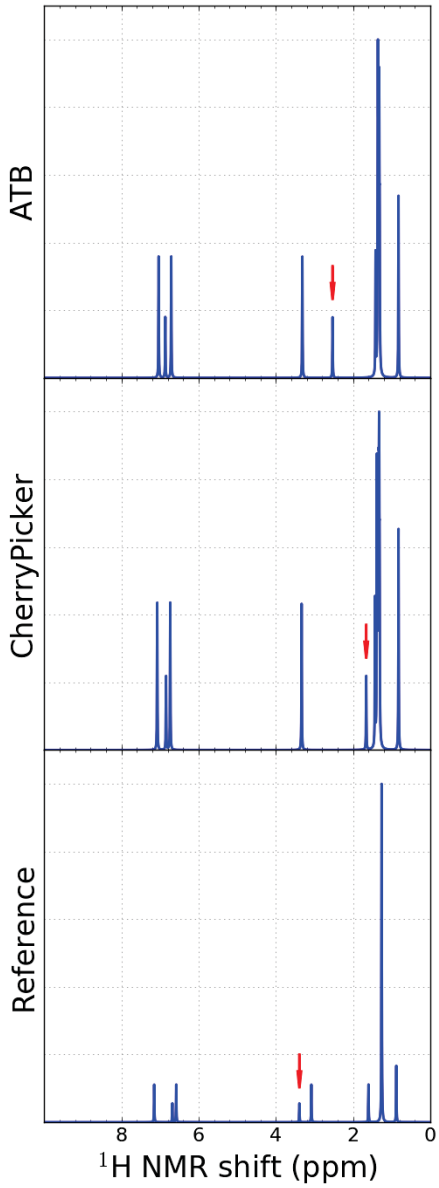


Figure 8.12: Calculated ^{1}H NMR spectra of molecule XXII. The amine hydrogen shifts, labelled with red arrows, are at 2.53 ppm for the ATB simulations, 1.66 ppm for the CherryPicker simulations and 3.39 ppm for the reference data.

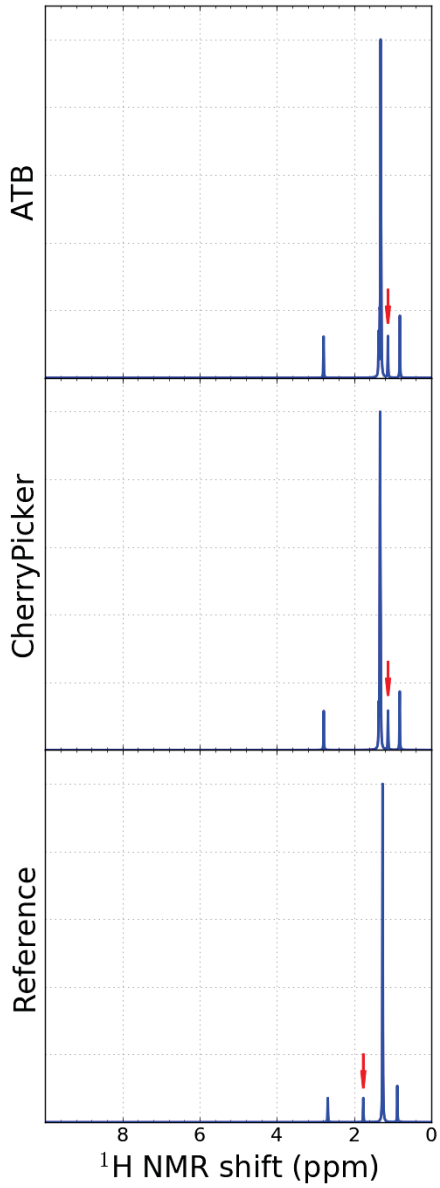


Figure 8.13: Calculated ^{1}H NMR spectra of molecule IX. The amine hydrogen shifts, labelled with red arrows, are at 1.12 ppm for both the ATB and CherryPicker simulations and at 1.76 ppm for the reference data.

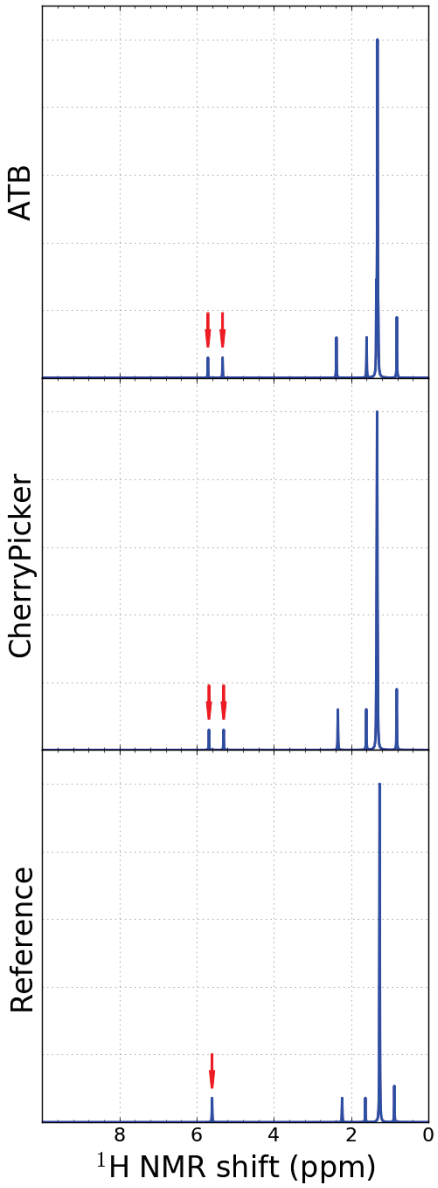


Figure 8.14: Calculated ^{1}H NMR spectra of molecule VII. The split hydrogen amide peaks, labelled with red arrows, are at 5.71 ppm and 5.33 ppm for the ATB simulations, and 5.68 ppm and 5.30 ppm for the CherryPicker. The single reference peak is at 5.60 ppm.

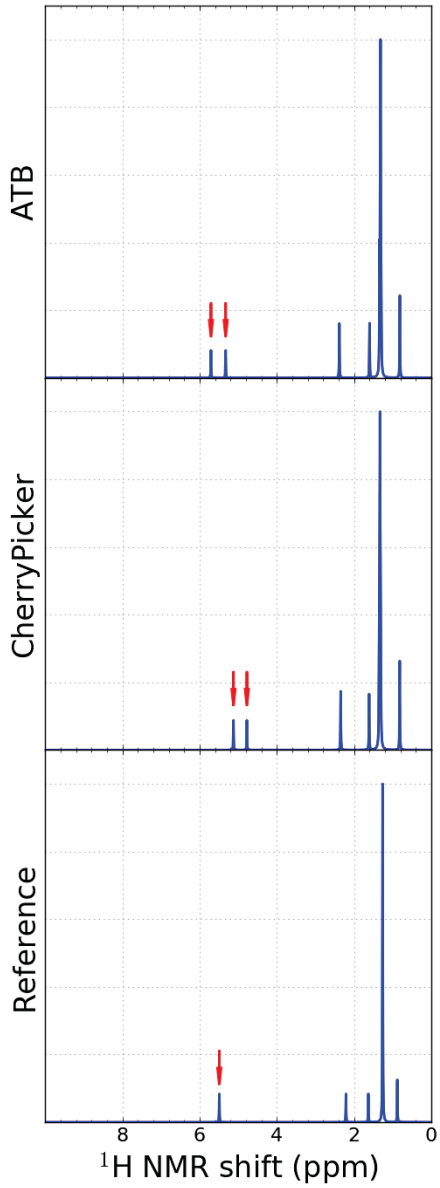


Figure 8.15: Calculated ^{1}H NMR spectra of molecule VIII. The split hydrogen amide peaks, labelled with red arrows, are at 5.71 ppm and 5.33 ppm for the ATB simulations, and 5.12 ppm and 4.78 ppm for the CherryPicker. The single reference peak is at 5.49 ppm.

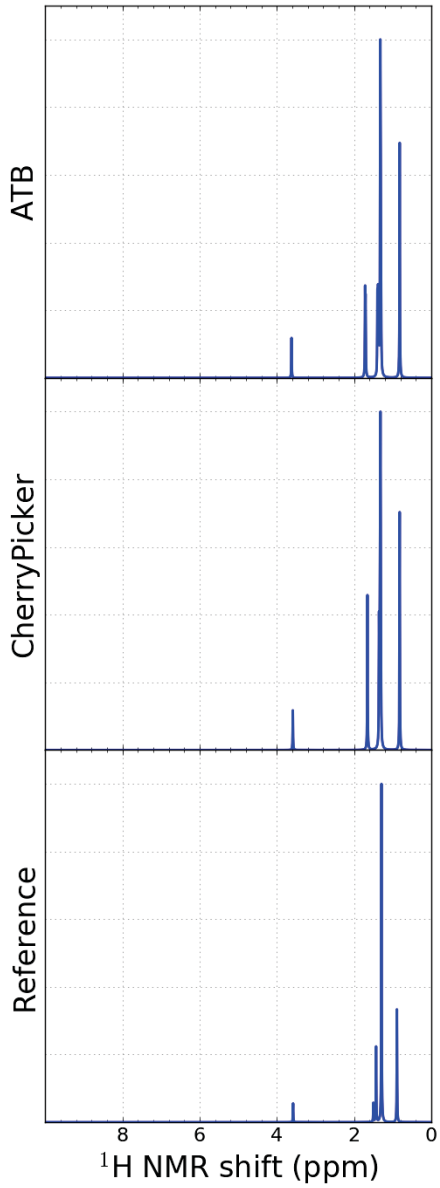


Figure 8.16: Calculated ^{1}H NMR spectra of molecule XVI.

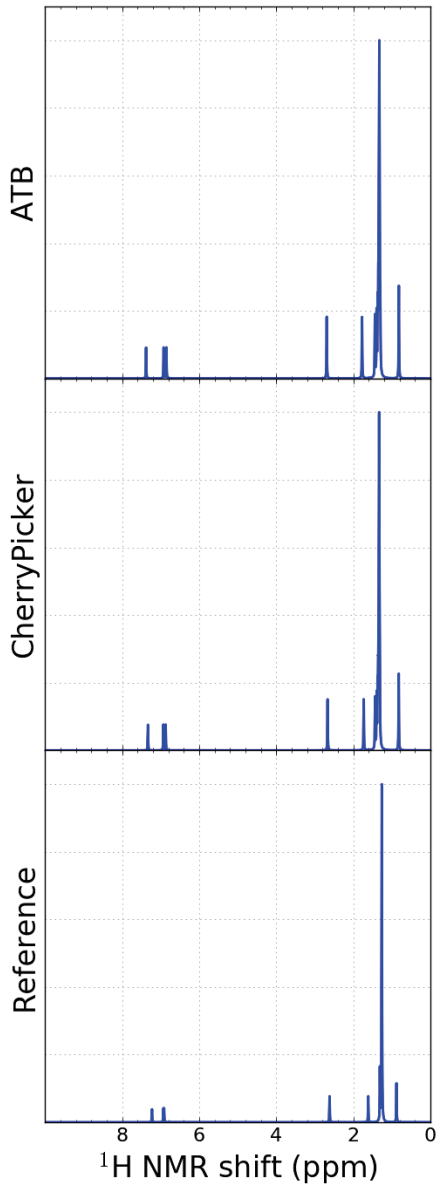


Figure 8.17: Calculated ^{1}H NMR spectra of molecule X.

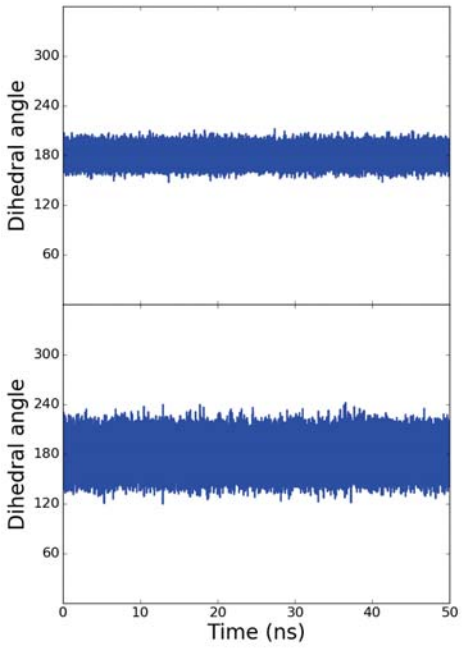


Figure 8.18: Time series of the H-N-C-O dihedral in the ATB parameter set (top) and CherryPicker parameter set (bottom) simulations of VII. They show fluctuation around 180°, indicating a lack of free rotation about the carbon–nitrogen bond.

which when plotted results in sharper peaks in the reference spectrum than the corresponding experimental spectrum had.

Molecules VII and VIII are very similar amides, with the only difference being that the carbon chain is two carbon atoms longer for molecule VII. They also both show a splitting of the amide hydrogen atom shifts which is not apparent in the reference spectra. In both cases, the experimental spectrum, figure 8.10d and e respectively, shows a very broad, low peak, spanning as much as 1 ppm. As this is the case, the splitting may in fact be represented in the experimental spectrum. The cause of the splitting is due to the intra-molecular hydrogen bond between one of the amide hydrogen atoms and the amide oxygen atom. This can be seen in a time series analysis plot of the dihedral defined by the hydrogen atom that is not part of the hydrogen bond, the nitrogen atom of the amide, the carbon atom of the amide and the oxygen atom of the amide. Such a time series is given for the simulation of molecule VII using both the ATB and CherryPicker parameter sets in figure 8.18. The time series shows fluctuation around a dihedral angle of 180°, which indicates that free rotation about the carbon–nitrogen bond is hindered by the intra-molecular hydrogen bond.

Molecule XXII (figure 8.12) is a benzalkyl amine. Both ATB and CherryPicker give similar shifts for the hydrogen atoms of the aromatic ring system, with the CherryPicker results being slightly closer to the reference data. However, both perform poorly for the amine hydrogen atom shift, with large upfield movement relative to the reference data, 2.53 ppm for the ATB simulation and 1.66 ppm for the CherryPicker simulation compared with 3.39 ppm for the reference data. Though the experimental peak seen in figure 8.10b is fairly broad, it is not broad enough to account for such drastic differences. This potentially indicates an over shielding from the aromatic system in both cases. A similar issue arises with molecule IX (figure 8.13), though not to nearly the same extent. Here, the amine hydrogen atoms have a slight upfield shift, 1.12 ppm for both simulations, relative to the reference data of 1.76 ppm.

8.3. Conclusion

This section of work presented the CherryPicker algorithm, which is designed to quickly parameterise large, biochemical molecules. The algorithm utilises a library of previously parameterised small molecules, from which fragments are produced, and sub graph isomorphism mapping to locate and combine overlapping fragments, producing parameters for the input molecule. Proof of principle tests were carried out through parameterising a small number of molecules with both the CherryPicker and the ATB as a reference. They show that the CherryPicker algorithm derived parameters perform similarly to the ATB parameters, at a fraction of the computational time to produce, for static and dynamic stability, and NMR generation.

9. Bond Order and Formal Charge Assignment

Bond orders and formal charges play an important role in the CherryPicker algorithm. They aid in determining how to fragment source molecules, as well as perform a critical role in the subgraph isomorphism matching process to ensure correctness of parameters. As such, it is imperative that they can be quickly and reliably assigned. This chapter details the development of methods which can be used to attain such a goal.

9.1. Introduction

Lewis structures show how the valence electrons of a molecule are arranged amongst the atoms and bonds of the molecule. The major driving force behind Lewis structures is the octet rule; that atoms will lose, gain or share electrons with one another in order to obtain a filled valence shell of eight electrons, matching the nearest noble gas electron configuration. Even though a Lewis structure is a crude representation of the electronic structure of a molecule, they do have some merit, particularly in organic chemistry. From Lewis structures, bond orders and formal charges can easily be deduced. A formal charge is the charge assigned to an atom in a molecule assuming that electrons in all chemical bonds are shared equally between atoms, and the bond order of a bond is the number of chemical bonds between a pair of atoms. Thus to determine bond orders and formal charges, Lewis structures need to be determined.

With the advent of large databases of organic molecules, such as the various PDB databases and the Cambridge Structural Database (CSD), the need to have an automated scheme to determine information, such as bond orders, about the various structures became apparent. As such, over the last few decades, a number of such schemes have been developed. The COBRA program of Leach *et al.* uses a backtracking search algorithm to automatically assign bond orders.[48] IDATM from Meng and Lewis can be used to determine connectivity and hybridisation state of atoms based on input three dimensional coordinates.[49] Baber and Hodgkin follow a similar scheme, but can also assign bond orders.[50] Lang *et al.* assign bond orders based on characteristic bond lengths, bond angles and torsion angles,[51] as do Hendlich *et al.* who also include small

functional group identification to help avoid wrong assignments due to erroneous or ambiguous geometrical data.[52]

Within the CherryPicker paradigm*, none of these methods are acceptable as they require accurate three dimensional information. Methods of Froeyen and Herdewijn[53] and Labute[54] could theoretically be used on structures with only atom type and connectivity information, though they have been developed more for use when three dimensional information is available for only the heavy atoms.

Wang *et al.* developed a heuristic method to determine bond orders based on arbitrary penalty scores.[5] Dehof *et al.* extended this work using the same penalty scores, developing three exact solvers guaranteed to find a bond order assignment with minimum total penalty score, and also allowing enumeration of all possible bond order assignments with minimum total penalty score.[55] These methods only require element type and atom connectivity information when all hydrogen atoms are included.

Dehof *et al.*'s method fits well within the CherryPicker paradigm, with one exception. As formal charges and bond orders are somewhat co-dependent, the absence of formal charges in the input molecule means that incorrect atom types could be perceived, leading to incorrect bond order assignments. Theoretically, formal charges can be back calculated from bond order assignments (if they are correct), however there can be situations where ambiguous formal charge assignments are possible, and there is no guarantee that the calculated formal charges will match the required total molecular charge, particularly with large magnitude molecular charges. As such, a new method for the simultaneous assignment of formal charges and bond orders has been developed. In essence, formal charges and bond orders are descriptions of the positions of valence electrons within a molecule. By minimising some function of the electron positions, a formal charge and bond order assignment is able to be generated. Given that electron positions are involved, the obvious choice of function would be something derived from high level QM calculations. This chapter gives a description of the function which is minimised and the development of various optimisation methods. Following this, testing and comparisons of the various optimisation methods with each other and the current state of the art A* method of Dehof *et al.*, here after referred to as the BALL method, are undertaken. Reference data for these tests comes from three databases containing structures with bond order and formal charge assignments. The databases used are the MMFF94 validation suite,[2] the KEGG Drug Database,[56] and ZINC15.[57–59]

*the CherryPicker paradigm is that three dimensional coordinates of all atoms are provided, though they cannot be guaranteed to be accurate, along with element type information, atom connectivity, and an overall total charge for the molecule.

9.2. General Problem Information

There are three globally relevant questions which must be answered before the process of optimising an electron distribution is undertaken:

1) what is the function which is to be optimised and how does it relate to electrons;

2) how many electrons need to be placed (section 9.2.2);

3) and where can electrons be placed (section 9.2.3).

9.2.1. Optimisation function

Definition 9.1. Let $G = (V,E)$ be a molecular graph. An *electron distribution* is a map $c_p : V \cup E \to S$ where $c_p(x)$ is the total number of electrons placed on member $x \in V \cup E$. The *energy* of the electron distribution is then given as

$$E = \sum_{x \in V \cup E} f(x) \tag{9.1}$$

where $f(x)$ is a function which performs lookup in the lookup tables LUT_{fc} and LUT_{bo} (see section 9.3.4) depending on if x is a vertex or an edge.

In the case where x is a vertex, LUT_{fc} requires an atomic number and a formal charge. The atomic number can determined with the map c_v from definition 7.14. The formal charge, $g(x)$ is calculated as

$$g(x) = \mathrm{val}(c_v(x)) - c_p(x) - \sum_{y \in N(x)} \frac{c_p(xy)}{2} \tag{9.2}$$

where $\mathrm{val}(c_v(x))$ is the *valence* of atomic number $c_v(x)$.

If x is an edge yz, LUT_{bo} requires two atomic numbers, the signs of their respective formal charges and a bond order. Atomic numbers are given as $c_v(y)$ and $c_v(z)$, and the bond order as $c_p(x)/2$. Formal charges are calculated as $g(y)$ and $g(z)$ as per equation 9.2. If $\sum c_p(x)$ for all $x \in V \cup E$ is not equal to e_T, then the energy of the electron distribution is set to ∞ as it is an invalid electron distribution. This is also the case if for some member $x \in V \cup E$ there is no valid entry in the lookup tables or for some vertex $v \in V$ the valence state of v (given by the sum of $c_p(e)$ for all $e \in E$ that are incident on v and $c_p(v)$) is larger than the target octet or hypervalent state of atomic number $c_v(v)$.

9.2.2. Electron Count

The answer to the second question is fairly simple. Given that the optimisation problem is focused on arranging the valence electrons about the molecule, the total number of electrons to position is thus

$$e_T = -q_T + \sum_{i=1}^{N} v_i \tag{9.3}$$

where e_T is the total number of electrons to place, q_T is the total molecular charge, N is the number of atoms in the molecule and v_i is number of valence electrons from atom i.

9.2.3. Electron Positioning

Not all electrons to be placed are involved in the optimisation process. There is a requirement that each bond in the molecule must have a bond order of at least one, meaning at least two electrons per bond. As this 'backbone' bonding electron count is always present, the electrons involved do not need to be part of the optimisation process. Further electrons can be positionally fixed by pre-filling lone pairs on some atoms. With data obtained from the KEGG LIGAND database[56] and MMFF94 database,[2] the probability of each type of atom[†] having between zero and three lone pairs is tabulated (see table H.1). For each atom, the number of electron lone pairs to add is given by the largest lone pair count with a probability of 1.0. For example, an oxygen atom with one bond will have two lone pairs added. These lone pairs are only added if the probabilities were calculated from more than 1000 data points. Generally, pre-filling of lone pairs only noticeably improves the performance of optimisation methods if there is a large concentration of atoms which will require lone pairs, such as high oxygen content molecules, and so is not utilised in the proceeding tests.

Definition 9.2. Let $G = (V,E)$ be a molecular graph. The *electron positions* are then given by the multiset $\mathcal{P}$. Using the octet rule, a vertex $v \in V$ is added to $\mathcal{P}$ such that mult$(\mathcal{P},v)$ is given by the difference between the target octet value of the element with atomic number $c_v(v)$ and $2 \cdot N(v)$. The target octet value is set at two for hydrogen and eight for all other elements. There is a `HYPERVALENT` switch which allows for some atoms to exceed the octet by becoming hypervalent. Here, this is limited to phosphorus and sulfur atoms with coordination number greater than two. They can have a valency of up to ten or twelve respectively. An edge $e \in E$ is added to $\mathcal{P}$ such the mult$(\mathcal{P},e)$ is the maximum value which will cause both vertices $u,v \in e$ to not exceed their target octet or hypervalent state.

[†]Here an atom type is determined from the number of bonds the atom is involved in, and its element.

9.3. Energy Calculations

The general idea of the bond order and formal charge assignment scheme developed here is to generate an optimum distribution of electrons around a molecule by minimising the energy of the molecule, i.e. produce a Lewis structure. The "best" way to produce a Lewis structure would be to calculate the actual electronic density distribution and then use the Natural Bond Orbital (NBO) method to obtain bond orders and formal charges.[60] However, this approach can be computationally expensive and so a different scheme is developed utilising an approximation to the molecular energy. A molecule's energy is determined as the sum of energy contributions from the individual atoms and bonds which make up the molecule. Working with closed shell systems is the main use case for this optimisation scheme, so electrons are treated as electron pairs as opposed to individual electrons. The nature of this optimisation scheme means that there is no real algorithmic difference between treating electrons as individuals or pairs; the optimisation methods should return the same results. However, individual electrons increase the size of the search space and so are computationally more expensive. As such, even though the code developed has the implemented capability of perform bond order and formal charge optimisation using single electrons, all tests reported here are performed with electron pairs. Beyond checking for ability to run, use of single electrons remains untested.

Energies are provided by the results of high level post Hartree-Fock calculations utilising the CR-CCL method,[61,62] and where appropriate the counterpoise correction method has been used to account for the basis set superposition error (BSSE).[63] Details of the calculations are given in sections 9.3.1 and 9.3.2. As the energies are directly derived from QM calculations, if molecules containing elements or bonds not already described within the energy tables are desired to have bond order and formal charge assignments made, it is a simple matter of performing some calculations to obtain new energies consistent with those already produced. There are two types of energies: those relating to bond order of bonds and those for formal charges on atoms. Each type is discussed in detail below.

9.3.1. Formal Charge Energies

Formal charges are attached to atoms, so naturally the energies associated with formal charges are derived from atomic energies. In a crude sense, atoms with formal charges can be described as ions with a charge equal to the formal charge. Taking this description, atomic/ionic energies of all possible valence states for an element are calculated. For example, carbon can have valence states ranging from $+4$ to -4, thus energies for C^{4+}, C^{3+}, C^{2+}, C^{+}, C^{0}, C^{-}, C^{2-}, C^{3-}, and C^{4-} are all calculated and provide the energies for the various possible formal charges of carbon. In normal molecules, it is highly unlikely that the majority of these formal charges are viable. Still,

the values are present in the penalty tables for completeness and to help guide the optimisation methods away from unrealistic solutions.

There are additional factors to be concerned with for the formal charge penalty scores. First is the spin state of the atom/ion. Looking at carbon again, there are four valence electrons in an electron configuration of $1s^2 2s^2 2p^2$. It is known that the lowest energy spin state is the triplet state, with the two electrons in the 2p shell being unpaired in degenerate orbitals. Both singlet and quintet states are conceivable, though they are higher energy states. To be consistent between elements, all energies are taken as the lowest energy spin state for the element with a given formal charge.

The second concern is large numbers. Elements with large atomic charges can have very large energies, which when summed together naturally lead to large overall molecular energies. Computationally comparing large values can run into issues with precision, especially when the comparison is looking for small differences between the two numbers, as is the case here when minimising the molecular energy. Given that across an entire molecule, the largest contributor to the molecular energy are the atoms, instead of using the absolute calculated energies as formal charge energies, relative energies are used. A reference energy for each element is calculated as the energy of the neutral atom in either the singlet or doublet state, depending on the number of electrons. All other calculated energies are subtracted from this reference value to give the final energy for that combination of element and formal charge.

Here, available elements have been limited to hydrogen, carbon, nitrogen, oxygen, fluorine, phosphorus, sulfur, chlorine, and bromine. These elements cover the majority of biochemical molecules. Energies for all possible formal charge states have been calculated using the CR-CCL method,[61,62] with the def2-SVPD and def2-TZVPPD basis sets.[64] Calculations were performed with GAMESS-US version 18 AUG 2016 (R1) software.[65,66] Relative energies are given in table 9.1.

Table 9.1: Calculated energies for all possible formal charge states of the available elements. Energies are calculated with two different basis sets and are relative to the lowest spin state of the neutral atom. All values are in E_h.

	def2-SVPD	def2-TZVPPD
H^-	0.02013	−0.00264
H^0	0.00000	0.00000
H^+	0.49928	0.49981
C^{4-}	1.45670	1.10560
C^{3-}	0.66591	0.48600

Continued on next page

Table 9.1 – *Continued from previous page*

	def2-SVPD	def2-TZVPPD
C^{2-}	0.20712	0.12642
C^{-}	−0.07677	−0.09020
C^{0}	−0.05938	−0.05385
C^{+}	0.34948	0.35759
C^{2+}	1.24107	1.24978
C^{3+}	2.98409	3.00539
C^{4+}	5.32103	5.37111
N^{3-}	0.90604	0.70853
N^{2-}	0.33772	0.20738
N^{-}	−0.02650	−0.06710
N^{0}	−0.10641	−0.10055
N^{+}	0.42292	0.43125
N^{2+}	1.50554	1.51559
N^{3+}	3.24192	3.25361
N^{4+}	6.06052	6.09490
N^{5+}	9.61128	9.68634
O^{2-}	0.11254	0.09397
O^{-}	−0.12968	−0.12915
O^{0}	−0.08409	−0.08232
O^{+}	0.40147	0.40942
O^{2+}	1.68570	1.69748
O^{3+}	3.69651	3.71155
O^{4+}	6.52685	6.54831
O^{5+}	10.67027	10.72593
O^{6+}	15.68525	15.79333
F^{-}	−0.12113	−0.11934
F^{0}	0.00000	0.00000
F^{+}	0.63130	0.63230
F^{2+}	1.89498	1.90703
F^{3+}	4.19052	4.20886
F^{4+}	7.38363	7.40644
F^{5+}	11.55460	11.59413

Continued on next page

Table 9.1 – *Continued from previous page*

	def2-SVPD	def2-TZVPPD
F^{6+}	17.27203	17.35880
F^{7+}	24.00196	24.15246
P^{3-}	0.67413	0.47698
P^{2-}	0.21734	0.11858
P^{-}	−0.04018	−0.07367
P^{0}	−0.06960	−0.06395
P^{+}	0.30853	0.32033
P^{2+}	1.02731	1.04397
P^{3+}	2.12335	2.14834
P^{4+}	3.99512	4.03146
P^{5+}	6.35594	6.41082
S^{2-}	0.03364	0.02470
S^{-}	−0.11824	−0.12008
S^{0}	−0.05233	−0.05080
S^{+}	0.30977	0.32100
S^{2+}	1.16010	1.17760
S^{3+}	2.42590	2.45359
S^{4+}	4.14584	4.18644
S^{5+}	6.78903	6.84375
S^{6+}	9.98929	10.06529
Cl^{-}	−0.12658	−0.12662
Cl^{0}	0.00000	0.00000
Cl^{+}	0.46359	0.46802
Cl^{2+}	1.31675	1.33168
Cl^{3+}	2.76026	2.78608
Cl^{4+}	4.69375	4.73544
Cl^{5+}	7.15408	7.21469
Cl^{6+}	10.68369	10.76125
Cl^{7+}	14.83826	14.93864
Br^{-}	−0.12165	−0.12263
Br^{0}	0.00000	0.00000
Br^{+}	0.42202	0.42725

Continued on next page

Table 9.1 – *Continued from previous page*

	def2-SVPD	def2-TZVPPD
Br^{2+}	1.18501	1.20276
Br^{3+}	2.46653	2.49427
Br^{4+}	4.16950	4.20793
Br^{5+}	6.32020	6.36814
Br^{6+}	9.45716	9.51383
Br^{7+}	13.14507	13.21067

9.3.2. Bond Order Energies

Bond dissociation energies are a natural source of energies for bond orders. A bond dissociation energy is defined as the change in enthalpy when a bond is homolytically cleaved. For example, the bond dissociation energy of the C−O bond in methanol is given by the enthalpy change associated with the reaction

$$H_3C{-}OH \xrightarrow{\Delta H} H_3C^{\cdot} + HO^{\cdot}$$

Computationally, this value can be determined by calculating the energy difference between the molecule containing the bond of interest and the two fragments produced by homolytic cleavage of the bond. Formally, bond dissociation energies are positive values though here the negative of the values are used.

For a given bond type, the process for determining the bond dissociation energy is as follows. The standard valence of each atom involved in the bond is filled by the addition of hydrogen atoms. This structure is geometry optimised at the MP2 level of theory[67] and a final energy calculated at the CR-CCL level. The fragments produced by homolytic cleavage are also optimised at the MP2 level before a final energy is calculated using CR-CCL. The negative bond dissociation energy is then given as

$$\Delta E = E(AB) - E(A) - E(B) \tag{9.4}$$

Due to the potential for structural distortion when comparing the individual fragments with their geometry in the complete molecule, and the use of finite basis sets, equation 9.4 can be corrected for the effect of the BSSE via the counterpoise correction method. With this method, the negative bond dissociation energy is given as

$$\Delta E = E_{AB}^{AB}(AB) - E_{AB}^{AB}(A) - E_{AB}^{AB}(B) + E_{AB}^{A}(A) + E_{AB}^{B}(B) - E_{A}^{A}(A) - E_{B}^{B}(B) \tag{9.5}$$

where $E_X^Y(Z)$ is the energy of component Z in the geometry of X with the basis set of Y. To allow for correct optimisation of the electron distribution, each higher order bond must have all lower order bonds between the same two elements calculated as well.

For the set of elements defined earlier, all possible bonds attainable with standard valencies were calculated with two different basis sets: def2-SVPD and def2-TZVPPD. Standard valencies limit bonds involving hydrogen and the halogens to a maximum bond order of one, bonds involving carbon, nitrogen or phosphorus to a maximum bond order of three, and bonds involving the chalcogens to a maximum bond order of two. The energies obtained are given in table 9.2.

Table 9.2: Calculated negative bond dissociation energies for the homolytic cleavage of bonds in neutral molecules. Energies are calculated with two different basis sets. The ‡ mark indicates energies account for the BSSE. All values are in E_h.

	def2-SVPD	def2-TZVPPD	def2-SVPD‡	def2-TZVPPD‡
Br−Br	−0.06626	−0.07774	−0.05798	−0.07253
Br−C	−0.11578	−0.12095	−0.10523	−0.11532
Br−Cl	−0.06827	−0.08190	−0.06122	−0.07827
Br−F	−0.08855	−0.09598	−0.07919	−0.09073
Br−H	−0.14444	−0.14805	−0.13761	−0.14372
Br−N	−0.08229	−0.08979	−0.07266	−0.08401
Br−O	−0.07551	−0.08459	−0.06598	−0.07888
Br−P	−0.09768	−0.10613	−0.08883	−0.10189
Br−S	−0.07593	−0.08819	−0.06746	−0.08366
C−C	−0.15291	−0.15215	−0.14485	−0.15022
C=C	−0.27810	−0.28395	−0.27090	−0.28182
C≡C	−0.40270	−0.42049	−0.39622	−0.41714
C−Cl	−0.13052	−0.13490	−0.12052	−0.13227
C−F	−0.17733	−0.17778	−0.17000	−0.17572
C−H	−0.17404	−0.17734	−0.17145	−0.17646
C−N	−0.14340	−0.14225	−0.13471	−0.14014
C=N	−0.25116	−0.25506	−0.24229	−0.25265
C≡N	−0.36505	−0.37818	−0.35592	−0.37482
C−O	−0.15198	−0.15201	−0.14347	−0.14967
C=O	−0.27907	−0.28278	−0.26999	−0.28007
C−P	−0.11521	−0.11819	−0.10658	−0.11640

Continued on next page

Table 9.2 – *Continued from previous page*

	def2-SVPD	def2-TZVPPD	def2-SVPD‡	def2-TZVPPD‡
C=P	−0.18584	−0.19396	−0.17688	−0.19184
C≡P	−0.26547	−0.28179	−0.25695	−0.27955
C−S	−0.11820	−0.12217	−0.10833	−0.11951
C=S	−0.19788	−0.20642	−0.18802	−0.20381
Cl−Cl	−0.06780	−0.08521	−0.06038	−0.08290
Cl−F	−0.08043	−0.09154	−0.07323	−0.08916
Cl−H	−0.16247	−0.16769	−0.15659	−0.16569
Cl−N	−0.08910	−0.09728	−0.07982	−0.09453
Cl−O	−0.07507	−0.08631	−0.06654	−0.08340
Cl−P	−0.11142	−0.12090	−0.10277	−0.11867
Cl−S	−0.08294	−0.09746	−0.07483	−0.09491
F−F	−0.04591	−0.05453	−0.04068	−0.05322
F−H	−0.21700	−0.22120	−0.21349	−0.21939
F−N	−0.10916	−0.11131	−0.10145	−0.10913
F−O	−0.06877	−0.07488	−0.06220	−0.07280
F−P	−0.16600	−0.17418	−0.15731	−0.17200
F−S	−0.11864	−0.12760	−0.10923	−0.12469
H−H	−0.16613	−0.17292	−0.16612	−0.17291
H−N	−0.17752	−0.18088	−0.17393	−0.17960
H−O	−0.19299	−0.19670	−0.18879	−0.19492
H−P	−0.13348	−0.13722	−0.12958	−0.13608
H−S	−0.14536	−0.15018	−0.14091	−0.14783
N−N	−0.11113	−0.11148	−0.10223	−0.10899
N=N	−0.19335	−0.19762	−0.18375	−0.19511
N≡N	−0.33520	−0.34742	−0.32536	−0.34502
N−O	−0.10397	−0.10592	−0.09553	−0.10376
N=O	−0.17890	−0.18440	−0.16797	−0.18116
N−P	−0.11862	−0.12405	−0.10894	−0.12181
N=P	−0.15815	−0.16929	−0.14858	−0.16692
N≡P	−0.20025	−0.21513	−0.19063	−0.21302
N−S	−0.10161	−0.10817	−0.09117	−0.10518
N=S	−0.13377	−0.14599	−0.12430	−0.14236

Continued on next page

Table 9.2 – *Continued from previous page*

	def2-SVPD	def2-TZVPPD	def2-SVPD‡	def2-TZVPPD‡
O−O	−0.07599	−0.08106	−0.06878	−0.07877
O=O	−0.17301	−0.18188	−0.16589	−0.17967
O−P	−0.13497	−0.14314	−0.12550	−0.14082
O=P	−0.19210	−0.20883	−0.18123	−0.20583
O−S	−0.10182	−0.11114	−0.09185	−0.10816
O=S	−0.17044	−0.18892	−0.15843	−0.18580
P−P	−0.08651	−0.09212	−0.07861	−0.09084
P=P	−0.11807	−0.12907	−0.10971	−0.12738
P≡P	−0.15283	−0.16920	−0.14410	−0.16750
P−S	−0.09656	−0.10515	−0.08769	−0.10308
P=S	−0.13096	−0.14502	−0.12180	−0.14290
S−S	−0.08676	−0.09934	−0.07792	−0.09677
S=S	−0.13372	−0.15312	−0.12616	−0.15083

9.3.3. Charged Bonds

Bond dissociation energies within charged molecules have different values to those within neutral molecules. When applied to the problem at hand, when there is a formal charge on atoms of a bond, the energy associated with that bond should be different than if the same bond had zero formal charge on both atoms. A series of single positively and negatively charged molecules were constructed and negative bond dissociation energies calculated. Construction involved the addition or removal of a single proton (i.e. a hydrogen atom without any bound electrons) from the charged atom relative to the neutral molecule. These molecules give the negative bond dissociation energies for bonds where one atom has formal charge and the other is neutral. The set of charged bonds calculated here has been limited to those bonds between hydrogen, carbon, nitrogen and oxygen where there is either a positive formal charge on a nitrogen or a negative formal charge on an oxygen. Energies determined for the various charged bonds are given in table 9.3.

9.3.4. Lookup Tables

Calculated energies are stored in lookup tables. For atoms, the key to the lookup table LUT_{fc} is given by the pair (z,q) where z is the atomic number of the atom and q is the formal charge. For bonds (LUT_{bo}), the key is given by the quintuple (z_u,q_u,z_v,q_v,b) where z_u and z_v are the

Table 9.3: Calculated bond dissociation energies for the homolytic cleavage of bonds in mono-charged molecules. Energies are calculated with two different basis sets. The ‡ mark indicates energies account for the BSSE. All values are in E_h.

	def2-SVPD	def2-TZVPPD	def2-SVPD‡	def2-TZVPPD‡
$C-N^+$	−0.19017	−0.18982	−0.18183	−0.18821
$C=N^+$	−0.34727	−0.35210	−0.33935	−0.35003
$C\equiv N^+$	−0.51069	−0.52567	−0.50241	−0.52141
$C-O^-$	−0.14427	−0.14582	−0.13611	−0.14299
$H-N^+$	−0.20642	−0.21030	−0.20341	−0.20950
$H-O^-$	−0.17702	−0.18133	−0.17358	−0.17960
$N-N^+$	−0.14386	−0.14518	−0.13539	−0.14308
$N=N^+$	−0.25216	−0.25838	−0.23878	−0.25436
$N\equiv N^+$	−0.39447	−0.41002	−0.38508	−0.40761
$N-O^-$	−0.06923	−0.07366	−0.06108	−0.07082
N^+-O	−0.11734	−0.12095	−0.10892	−0.11868
$N^+=O$	−0.20950	−0.21802	−0.20096	−0.21572
$O-O^-$	−0.19368	−0.15766	−0.18796	−0.15539

atomic numbers of the atoms involved in the bond, b is the bond order and q_x is -1 if the formal charge on x is negative, 0 if there is no formal charge, and $+1$ if the formal charge is positive. In case there is not an explicit energy for a given quintuple, the energy defaults to that given by the quintuple where q_u and q_v are both 0.

9.4. Optimisation Methods

A number of different optimisation techniques were developed and implemented to solve the bond order and formal charge assignment problems. These included greedy, stochastic and exact optimisation methods. Each method is described below.

9.4.1. Local Optimisation

Local optimisation acts similarly to a steepest decent gradient optimisation method. It is a greedy method that searches for an optimal set of bond order and formal charge assignments by taking the lowest energy neighbour of a given electron distribution and iteratively applying the neighbour search until there are no lower energy neighbours. Computationally, it is a relatively cheap optimisation method, and will always converge. Being greedy, convergence is not guaranteed to be to the global minimum.

Set-up As per equation 9.3 and definition 9.2, the number of electrons required to be added to a molecule and the electron positions that they could be placed are determined. These positions are sorted by probability of being filled (see section 9.4.3.2), in descending probability order. An initial electron distribution is generated by filling the first e_T possible electron locations in the sorted ordering. The molecular energy with this initial electron distribution is calculated and used as the initial energy value.

Neighbours Local optimisation determines the energy change that would result when going from one electron distribution to each of its neighbours. The neighbours of an electron distribution are determined as follows. The multiset of possible positions for electrons to be placed, $\mathcal{P}$ is converted to a set P, i.e. duplicate members are removed. Every member $p \in P$ is checked to determine if it contains electrons in the electron distribution, i.e. $c_p(p) \neq 0$, meaning that it can be used as a source position. If it can be used as a source position, all other members $q \in P$ are checked to determine if they can hold another electron, i.e. $\text{mult}(\mathcal{P}, q) - c_p(q) > 0$, meaning that q can act as a target position. With a valid source position and target position, a neighbour of an electron distribution is the electron distribution produced when moving an electron or electron pair from the source position to the target position. Thus, all the neighbours of an electron distribution is the set of electron distributions produced from all possible source–target pairs.

Energy Minimisation Determining an optimal bond order and formal charge assignment using the local optimisation method is a straightforward process. Energy changes on going from the initial electron distribution to each of its neighbours are determined. If the initial electron distribution has at least one neighbouring electron distribution with lower energy, the neighbour search is repeated using the lowest energy neighbour as the initial electron distribution. This iterative process updating the initial electron distribution proceeds until there are no neighbours with lower energy, in which case the optimisation process has converged to a local minimum. The ability to enumerate multiple solutions is present if desired. It is provided by keeping a list of electron distributions with the same energy as the lowest energy distribution, and yielding all of the lowest energy distributions found upon completion of the optimisation. However, only the first member of the list is used as the initial electron distribution for each iteration of the optimisation procedure.

9.4.2. Genetic Algorithm

A genetic algorithm is a stochastic optimisation method emulating the process of natural selection. A population of candidate solutions, i.e. electron distributions, is evolved towards better

solutions through a combination of point mutations and recombination. The fitness of each candidate solution is evaluated using some objective function, the energy of the electron distribution (definition 9.1), and influences the make-up of future generations. To allow for easier programmatic manipulation, a candidate electron distribution is represented as an array of bits, with each bit representing a possible position for an electron or electron pair to be placed, giving of total of $|\mathcal{P}|$ bits. A `1` means that position is filled and a `0` that it is empty. As such, the total number of bits set to `1` is restricted to the number of electrons or electron pairs that need to be placed, e_T.

9.4.2.1. Initialisation

An initial population size consisting of N_p candidate solutions is generated. Candidate solutions can be generated either randomly or through a seeding mechanism. A random candidate solution, $\mathcal{S}$, is produced by randomly choosing elements from $\mathcal{P}$ such that $\mathcal{S} \subset \mathcal{P}$ and $|\mathcal{S}| = e_T$. The seeding mechanism employed uses a greedy position filling process to create an initial candidate solution before performing mutations to create further candidates. Greedy position filling is performed as follows. A base energy is calculated using the partial electron distribution where no electrons or electron pairs have been positioned. Positions are filled by iteratively choosing the position which causes the most negative change in the energy of the partial electron distribution. This initial candidate solution is then used as the source candidate solution and point mutated (section 9.4.2.3) to produce a further candidate solution. If either the mutated candidate solution has a lower electron distribution energy than the source solution, or the source solution has already been mutated five times, the source solution is replaced with the mutated candidate. Mutations proceed iteratively, updating the source candidate as described, until N_s candidate solutions have been generated this way, with $N_s \leq N_p$.

9.4.2.2. Breeding Selection

Breeding selection selects $N_p/2$ unique pairs of candidate solutions which are bred together to form N_p children. Each parent pair is obtained by randomly choosing two members of the initial population. The selection process can be biased towards selecting fitter parents by the ξ_b parameter. ξ_b is a scale parameter to affect the probability of selecting lower energy candidates. When $\xi_b = 0$ each candidate has an equal probability of being selected, while at $\xi_b = 1$, a candidate's probability of being selected depends entirely on its energy in relation to all other candidate energies. For a given value of ξ_b, the probability of selecting a state i is:

$$P_i = \frac{\xi_b \exp(-E_i)}{\sum_{j=1}^{N_p} \exp(-E_j)} + \frac{1-\xi_b}{N_p} \tag{9.6}$$

where E_i is the energy of candidate i.

9.4.2.3. Genetic Operators

Genetic operators are the driving force behind the ability of genetic algorithms to optimise solutions. There are two main operators, mutation and crossover, though others are possible. Only the mutation and crossover operators are used here, and are detailed below.

Mutation The mutation operator as employed here reverses the state of two random positions, one of which is filled, the other which is empty. A filled source position is randomly chosen and paired with a randomly selected non-degenerate unfilled target position. The fill state of these two positions is then flipped.

Crossover The crossover operator uses a two parent, a and b, single point crossover technique to create two children. Each parent independently undergoes mutation with a probability ρ_m, prior to the crossover process. Potential crossover points are then determined by tracking the total number of 1's present in each of the parents. A potential crossover point is one where the total number of 1's up until that point is the same between the two parents, ensuring that both children have the correct number of electrons or electron pairs. The crossover point, x, used is randomly chosen from the set of all potential crossover points, and two children produced. The first child matches parent a from 0 to x and parent b from x to $|\mathcal{P}|$. The second child is the opposite of this, matching parent b from 0 to x and parent a from x to $|\mathcal{P}|$.

9.4.2.4. Generational Progression

The selection process chooses which candidate solutions to pass onto the next generation from the combination of the produced child candidate solutions and the candidate solutions of the current generation, i.e. there are $2 \cdot N_p$ candidate solutions to choose from. The method used to select the next generation is the same as that used to select parents for breeding, though ξ_b in equation 9.6 is replaced by an independent parameter, ξ_g. Once a candidate solution has been selected for progression to the next generation, it is removed from the current generation.

9.4.2.5. Termination

Termination of the genetic algorithm occurs in one of two situations. The first occurs when some maximum number of generations, M, have passed. The second occurs when at least m generations have passed, and the lowest energy candidate solution has not changed for t generations. Further, a situationally more efficient brute force method, which calculates the energy

of all possible combinations of electron positions, is used instead of the genetic algorithm when $\binom{|\mathcal{P}|}{e_T} \leq mN$.

9.4.3. A*

A* is a path-finding algorithm for determining a minimum cost path between a start, s, and end, t, location.[68] It employs a search heuristic as a means to guide the path finding process towards more promising paths. A function $f(v) = g(v) + h(v)$, where $g(v)$ is the real cost of the path $s \ldots v$ and $h(v)$ is an heuristic estimate of the cost of the path $v \ldots t$, is assigned to each visited vertex v. The list of vertices to expand is stored in a priority queue, meaning the most promising vertices (lowest $f(v)$ value) are expanded first. Obviously, the nature of the heuristic function is going to influence the efficiency of the search algorithm. To be guaranteed to obtain a minimum cost path, the heuristic must be *admissible*, meaning 'optimistic'. That is, the true cost of the path $v \ldots t$ cannot be lower than $h(v)$. Dehof *et al.* utilised an A* approach as one their three optimisation methods for bond order assignment.[55] A similar approach is taken here.

Let $P \subset \mathcal{P}$ be the set of unique possible locations to place electrons. The electron distribution optimisation problem of the molecular graph $M = (A, B)$ previously outlined can be formulated into a $|P|$-level tree T, i.e. the path from the root vertex to a leaf will be of length $|P|$. Each level of the tree represents a possible location for electrons or electron pairs to be positioned. A vertex at level k has $m+1$ neighbours, where $m = \text{mult}(\mathcal{P}, u)$ where $u \in P$ is the position associated with level $k+1$, to allow for all possible electron or electron pair counts placed in u, from 0 to m.

To enable formulation of the scoring functions $g(v)$ and $h(v)$, some additional values must be defined. A partial electron distribution, $R(v)$, is denoted as the set of pairs (p, n) where p is a member of the path $s \ldots v$ and $n \in \{0, \ldots, \text{mult}(\mathcal{P}, p)\}$ is the number of electron or electron pairs placed there. $R(v)$ also contains pairs $(x, 0)$ for all elements $x \in A \cup B \setminus P$. $Q(v)$ is the set of *calculable* members $x \in A \cup B$ at vertex v, as described in definition 9.3.

Definition 9.3. Let $M = (A, B)$ be a molecular graph. A member $x \in A \cup B$ is deemed *calculable* at vertex $v \in T$ if the following conditions are met

1) $x \notin P \setminus R_i$ where R_i is the set of first members of $R(v)$;

2) If $x \in A$, for all neighbours $n \in N(x)$ condition 1 holds;

3) If $x \in B$ the pair $x = y, z \subseteq Q(v)$.

As condition 3 is a requirement for determining the calculability of bonds, all atoms have their calculability determined before any bonds do.

9.4.3.1. Scoring Functions

Each vertex that is visited through the A* search is assigned a score, $f(v)$ combining two components, $g(v)$ and $h(v)$. The exact cost of the path $s \ldots v$ is given by $g(v)$ and defined as

$$g(v) = \sum_{u \in Q(v)} E(u, R(v)) \tag{9.7}$$

where $E(u,R(v))$ is the energy of element $u \in A \cup B$ with the partial electron distribution as given by $R(v)$.

Promiscuous Heuristic The most obvious search heuristic to use is

$$h(v) = \sum_{u \in Q(v)^{\complement}} \min\{E[u]\} \tag{9.8}$$

where $E[u]$ is the set of all tabulated energies associated with atom/bond u. Obviously, this $h(v)$ scoring function is too optimistic and can be refined.

Abstemious Heuristic Combined with $\mathcal{P}$, if the partial electron distribution given by $R(v)$ is taken into account the search heuristic can be made less optimistic. A potential bond order n for a bond $b = \{x,y\}$ becomes invalid if the number of electrons or electron pairs required to obtain it is greater than $m = \text{mult}(\mathcal{P}, b)$ or is greater than $e_T - e_R$ where $e_R = \sum_{(i,j) \in R(v)} j$. Additionally, for both atoms $a \in \{x,y\}$ of the bond, if $n + \text{val}(a,R(v)) > \text{oct}(a)$ where $\text{val}(a,R(v))$ is the valence state of atom a within the partial electron distribution $R(v)$ and $\text{oct}(a)$ is the target octet or hypervalent valence state, the bond order also becomes invalid. A potential formal charge for an atom a becomes invalid if it cannot be attained, given the partial electron distribution $R(v)$, by placing electrons or electron pairs on the incident bonds not in $R(v)$. Thus $E[u]$ in equation 9.8 becomes the set of all *valid* tabulated energies associated with atom/bond u.

9.4.3.2. Ordering of P

The order in which the members of P are arranged into levels of T plays an important role in the efficiency of the A* algorithm. Having a poorly ordered set of possible locations can result in up to a fifty fold increase in the number of vertices of T that need to be expanded over a good ordering. Through experimentation, it was discovered that a good order tends to have those locations which are more likely to have electrons placed on them higher up the tree. To that end, positions of electron pairs were determined for all molecules in the MMFF94 validation suite and the KEGG Drug Database using the reference bond orders and formal charges. Atoms

were grouped based on their element and coordination number. Bonds were grouped based on the element and coordination number of both atoms that make up the bond. For each group, the total number of members with between zero and three electron pairs placed on them were determined and the probability of each electron pair count calculated. These results are tabulated in tables H.1 and H.2. For a given molecule, the set of possible positions to place electrons can be sorted in a descending order based on these probabilities, giving a simple means for the A* algorithm to be noticeably more performant.

9.4.4. Fixed Parameter Tractable (FPT)

In a similar vein to Dehof *et al.* we also implement a fixed parameter tractable (FPT) based approach.[55] Given a molecular graph $G = (V, E)$ which is a tree, the electron distribution problem can be easily solved using dynamic programming, i.e. recursively splitting the problem into smaller sub problems and solving the sub problems. Of course, not all molecular graphs are trees, but their generally sparse nature means that they are 'tree-like'.

Definition 9.4. Let G be a graph, T a tree, and let $\mathcal{V} = (V_t)_{t \in T}$ be a family of vertex sets $V_t \subseteq V(G)$ indexed by the nodes t of T. The pair $(T, \mathcal{V})$ is called a *tree-decomposition* of G if it satisfies the following three conditions:

T1) $V(G) = \bigcup_{t \in T} V_t$;

T2) for every edge $e \in G$ there exists a $t \in T$ such that both ends of e lie in V_t;

T3) $V_{t_1} \cap V_{t_3} \subseteq V_{t_2}$ whenever $t_1, t_2, t_3 \in T$ satisfy $t_2 \in t_1 T t_3$.

Conditions T1 and T2 together say that G is the union of the subgraphs $G[V_t]$; we call these subgraphs and the sets V_t themselves the *parts* of $(T, \mathcal{V})$ and say that $(T, \mathcal{V})$ is a tree-decomposition of G into these parts. Condition T3 implies that the parts of $(T, \mathcal{V})$ are organised roughly like a tree. Figure 9.1b shows a tree decomposition of the graph in figure 9.1a. The *width* of $(T, \mathcal{V})$ is the number

$$\max \big\{ |V_t| - 1 : t \in T \big\},$$

and the *tree-width* $\mathrm{tw}(G)$ of G is the least width of any tree-decomposition of G.

Following from this definition, the vertices t of a tree-decomposition will be referred to as the *nodes* of the tree-decomposition. Any mention of vertices will refer to the underlying graph G. To simplify the algorithm we utilise nice tree-decompositions.

Definition 9.5. A tree decomposition $(T, \mathcal{V})$ of G is called *nice* if it satisfies the following conditions

N1) T is rooted at a leaf node r and $V_r = \emptyset$;

N2) For every leaf $l \in T$, $V_l = \emptyset$;

N3) Every node $t \in T$ has at most two children;

N4) If $t \in T$ has two children, p and q then $V_t = V_p = V_q$ and is known as a join node;

N5) If $t \in T$ has one child, p then one of the following conditions is true:

a) $V_t \subset V_p$ and $|V_t| = |V_p| - 1$ and is known as a forget node with forgotten vertex $v_{t_f} \coloneqq V_p \setminus V_t$.

b) $V_t \supset V_p$ and $|V_t| = |V_p| + 1$ and is known as an introduce node with introduced vertex $v_{t_i} \coloneqq V_t \setminus V_p$.

Forget and introduce nodes are defined in relation to the path $t_l \ldots t_r$. Figure 9.1c shows a nice tree decomposition as produced from the tree decomposition in figure 9.1b.

The process for obtaining a nice tree decomposition of a molecular graph is as follows. A minimum width tree decomposition of a molecular graph is obtained using the QuickBB method[69] as implemented in the LibTW library.[70] In accordance with condition N2 for nice tree decompositions, additional nodes i with $V_i = \emptyset$ are added to each leaf of the tree decomposition, and a root node is carefully chosen from amongst them to minimise the total number of partial solutions (condition N1). The tree decomposition is then converted from an undirected tree to a directed tree with edges pointing away from the root node.

Condition N3 is met as follows. As long as T has a node t with more than two children, let $t_1, t_2 \ldots t_p$ be the children of t. A new vertex t' is added to $(T, \mathcal{V})$ with $V_t = V_{t'}$ in the following manner. The parent of t' is set to t and for each child t_i of t with $i \geq 2$ t' is made the parent of t_i. This shifts the over-degree node problem from t to t', but with degree reduced by one, thus the process is repeated until condition N3 is met.

Moving on to condition N4, as long as T has a node t with children t_1 and t_2 with $V_t \neq V_{t_1} \neq V_{t_2}$, additional nodes t'_1 and t'_2 are added to $(T, \mathcal{V})$ with $V_t = V_{t'_i}$ for $i \in \{1, 2\}$. The parent of t'_1 and t'_2 is set to t and the parent of t_1 and t_2 changed to t'_1 and t'_2 respectively. The two children are treated independently such that if $V_t \neq V_{t_1}$ but $V_t = V_{t_2}$, only $(t_1, V_{t'_1})$ is added to $(T, \mathcal{V})$.

The final step is to make T satisfy conditions N5a and N5b. For every node t with child t_1 if $V_t \nsubseteq V_{t_1}$ and $V_t \not\supseteq V_{t_1}$, an additional node t'_1 with $V_{t'_1} = V_t \cap V_{t_1}$ as added to $(T, \mathcal{V})$. The parent of t'_1 is set to t and the parent of t_1 is set to t'_1. This means that forget nodes can be added as

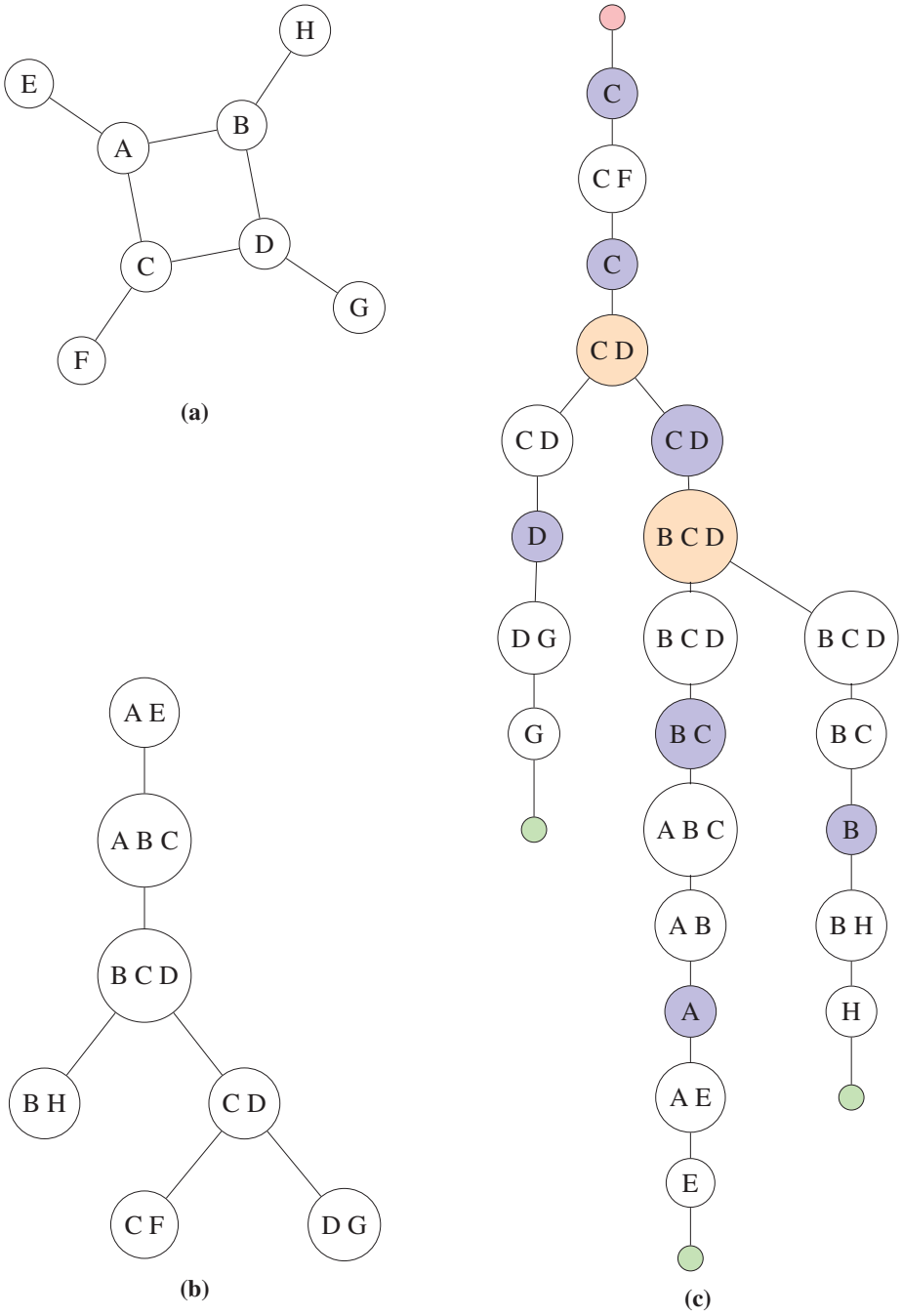


Figure 9.1: **(a)** A graph $G = (V,E)$ with $V = \{\text{A,B,C,D,E,F,G,H}\}$ and $E = \{\text{(A,B),(A,C),(A,E),(B,D),(B,H),(C,D),(C,F),(D,G)}\}$; **(b)** a tree-decomposition $(T,\mathcal{V})$ of G as per definition 9.4; **(c)** a nice tree-decomposition of that in b as per definition 9.5. The nodes of the nice tree-decomposition are coloured red for the root node, green for leaf nodes, white for introduce nodes, blue for forget nodes and orange for join nodes.

descendants of t_1' and introduce nodes as ancestors. For every vertex t with child t_1, if $V_t \supset V_{t_1}$ and $|V_t| = |V_{t_1}| + k$ for some $k > 1$, a new introduce node t_1' is introduced to $(T, \mathcal{V})$ as the child of t and parent of t_1. Let $u \in V_t \setminus V_{t_1}$ be a vertex of G, then $V_{t_1'} = V_{t_1} \cup \{u\}$. As a simple means to increase efficiency of the optimisation algorithm, u is chosen such that it minimises the number of partial solutions introduced when going from t_1 to t_1'. This process is repeated until all nodes along the path $t_1 \ldots t$ satisfy condition N5b. Similarly, for every node t with child t_1, if $V_t \subset V_{t_1}$ and $|V_{t_1}| = |V_t| + k$ for some $k > 1$, a new forget node t_1' is introduced to $(T, \mathcal{V})$ as the child of t and parent of t_1 with $V_{t_1'} = V_{t_1} \setminus \{u\}$ where $u \in V_{t_1} \setminus V_t$ is a vertex of G, repeating until condition N5a is met. In the same manner as the introduce node case, u is chosen to maximise the number of partial solutions removed when going from t_1 to t_1'. At this point, all conditions for a nice tree decomposition are met.

The electron distribution optimisation process requires determining energies for both atoms and bonds. Explicitly adding edges into the tree decomposition bags enables this, and simplifies the algorithm at the expense of increasing the tree width. With explicit edges, the notion that $\mathcal{V}$ is the family of vertex sets $V_t \subseteq V(G)$ no longer holds. Rather $V_t \subseteq V(G) \cup E(G)$. To determine the formal charge of an atom, electrons on the atom and all incident bond orders need to be provided. To determine the bond order of a bond electrons in the bond need to be provided. However, if charged bonds are in use, the energy of a bond requires the electrons in the bond and the formal charges of the atoms of the bond in order to be calculated. This requires that before a bond can be scored, all other bonds incident to either of the atoms of the bond must also be present. This is achieved by adding an introduce vertex for every not yet introduced edge immediately proceeding the introduce vertex of the first atom component of the edge. A forget vertex for each edge is added immediately proceeding the forget vertex of the second atom component of the edge. Of course, the edge is additionally added to all bags along the path from introduce vertex to forget vertex. Edges are introduced to the tree decomposition immediately after conversion to a directed tree.

9.4.4.1. Algorithm

Let $t \in T$ be node of the nice tree-decomposition of a graph. Then $\mathfrak{X}_t$ is the set of forgotten vertices $v_{s_f} \in V_s$ associated with the forget nodes of the subtree T_t induced on T below (and including) the node t. The total number of electrons to place e_T and the multiset $\mathcal{P}$ of positions at which the electrons can be placed are determined as per section 9.2. Each node t is assigned a score table S_t indexed by the ordered pair $(l,k) \in L_t \times K_t$ where

$$L_t = \{n_{\min}, \ldots, n_{\max}\} \tag{9.9}$$

$$n_{\min} = \max\{0, e_T - |\mathcal{P}| + |\mathcal{P} \cap \mathcal{X}_t|\} \tag{9.10}$$

$$n_{\max} = \min\{e_T, |\mathcal{P} \cap \mathcal{X}_t|\} \tag{9.11}$$

$$K_t = X_1 \times \cdots \times X_j \tag{9.12}$$

$$X_j = \{(j,k) : j \in V_t, 0 \leq k \leq \text{mult}(\mathcal{P}, j)\}. \tag{9.13}$$

with $S_t[l,k]$ being the minimum energy of forgotten vertices $\mathcal{X}_t$ with $l \in L_t$ forgotten electrons and the additional constraint of further partial electron distribution $k \in K_t$. Beginning from the leaves of the nice tree decomposition, and scoring only when all children of a node have been scored, the algorithm distinguishes the kind of each node and determines the score matrix as follows:

Leaf Nodes Leaf nodes are empty sets, so the score table of a leaf node is also empty.

Introduce Nodes Let $t \in T$ be the introduce node with child c, and $x_t = \mathcal{X}_t \setminus \mathcal{X}_c$. Then

$$S_t[l,k] = \begin{cases} S_p[l, k \setminus x_t] & \text{if } k \setminus x_t \neq \emptyset, \\ \infty & \text{otherwise.} \end{cases} \tag{9.14}$$

Forget Nodes Let $t \in T$ be the forget node with child c, and $x_t = \mathcal{X}_c \setminus \mathcal{X}_t$. Then

$$S_t[l,k] = \min_{\substack{n \in \{0,\ldots,\text{mult}(\mathcal{P},x_t)\} \\ p \in L_c : p+n=l}} \left\{ E(x_t, k \cup \mathcal{X}_t, n) + S_c[p, k \cup x_t] \right\} \tag{9.15}$$

where $E(x_t, k \cup \mathcal{X}_t, n)$ is the energy of x_t with n electrons positioned and the partial electron distribution $k \cup \mathcal{X}_t$.

Join Nodes Let $t \in T$ be the parent of c_1 and c_2 with $V_t = V_i$ for $i \in 1,2$. Then

$$S_t[l,k] = \min_{(p,q) \in L_{c_1} \times L_{c_2} : p+q=l} \left\{ S_{c_1}[p,k] + S_{c_2}[q,k] \right\} \tag{9.16}$$

Root node Each nice tree decomposition has only one root node $r \in T$ which is formally a forget node. However the score table of the root node is unpopulated as $K_r = \emptyset$. Rather than fill a score table, the minimised electron distribution energy is determined. Let c be the child of r with $x_r = \mathcal{X}_c$. Then the minimum energy is given by

$$E_{\min} = \min_{\substack{n \in \{0,\ldots,\text{mult}(\mathcal{P},x_r)\} \\ p \in L_c : p+n=e_T}} \left\{ E(x_r, \mathcal{X}_r, n) + S_r[p, x_r] \right\} \tag{9.17}$$

9.4.5. Evaluation

The performance of each algorithm was assessed in terms of both speed, and the ability to correctly assign bond orders and formal charges. The latter requires reference data, i.e. a set of molecules for which both properties are already known. The reference data must also represented aromatic bonds in kekulised form, i.e. alternating single and double bonds, and explict hydrogens must be present. To match these requirements, three molecular databases were chosen to provide validation sets, the MMFF94 validation suite,[2] the KEGG Drug Database,[56] and ZINC15.[57–59] Both the MMFF94 validation suite and KEGG Drug Database were used as validation sets for the BALL method.[55]

The MMFF94 validation suite contains 761 structures for small molecules and ions, with bond orders, formal charges and missing hydrogen atoms manually added by the authors.[2] Formal charges and bond orders are available in either hypervalent or dative representation. The hypervalent representation has been chosen here. All molecules in the suite were parsed by OpenBabel[41] and removed if any errors or warnings were issued. Additionally, if they contained three or fewer atoms, elements not contained in the LUT_{fc} lookup table (section 9.3.4), or an odd number of valence electrons they were also removed. This reduced the suite to 700 unique molecules for testing purposes. The set of molecules can be found in table G.1.

The KEGG Drug Database contains a large number of drug like molecules.[56] Structure files contain only two dimensional coordinate information, meaning that they are a perfect test set for topology only bond order and formal charge assignment. Hydrogen atoms are not explicitly present in the coordinate files, but atom typing means that they can be easily added using standard rules by OpenBabel.[41] Furthermore, some of the structure files contain multiple molecules, and each molecule may appear in more than one file, or also appear in the MMFF94 validation suite. To prevent skewing the analysis through inclusion of identical[‡] molecules, each file was split into connected components, and each component checked against all others, including those of the MMFF94 set, for uniqueness. Again, molecules with three or fewer atoms, containing elements not in LUT_{fc}, or with an odd number of valence electrons were also ignored. This preparation gave 5955 unique molecules, which are listed in table G.2.

ZINC15 is a free online database containing over 100 million commercially available compounds.[57–59] Molecules are available as SMILES strings, meaning that they do not have explicit hydrogen atoms which must again be added in using the standard rules employed in OpenBabel. With such a large database, it is infeasible to check the algorithms against all valid and unique molecules. Rather, 11,000 molecules were randomly chosen and passed through the same filtering processes as the MMFF94 validation suite and KEGG Drug Database. This resulted in

[‡]stereo isomerisation is ignored when comparing molecules

10844 unique molecules for validation which are listed in table G.3.

In section 9.4.5.1 the four algorithms developed here are compared with one another. Using the FPT algorithm, section 9.4.5.2 compares the results obtained using an exact algorithm with the reference bond order and formal charge assignments of the three molecular databases. This section also compares the results of different computation levels to obtain the bond and atom energies, and the accuracy of the exact algorithm compared with that of the BALL method.

9.4.5.1. Comparison of Algorithms

The four algorithms developed here are compared amongst themselves. The first comparison is to determine how well and often the algorithms produce the minimum energy electron distribution. As both A* and FPT are exact, given enough time they will give identical results so there is no point in determining the accuracy of these two methods. Because they are exact methods, their results can be used as a basis to determine the accuracy of the other methods. Another point of interest is the relative efficiencies of the four algorithms. To determine this, each method is run with a two and a half second time-out period, that is if the algorithm has not completed within 2.5 seconds it is halted and treated as a fail. This time-out period was chosen as a pragmatic limit for the computational expense to determine an electron distribution for a given molecule. It does not imply that the algorithm has failed to optimise, rather that the time period it would take to optimise is much larger than 2.5 seconds. The more efficient methods will have less fails due to time-outs. The results of this method of efficiency determination are not entirely reliable, as they are skewed towards smaller systems and do not consider the different scalability of the algorithms, but they do provide an easy to obtain estimate.

For these tests, the atom and bond energies used were calculated using the def2-SVPD basis set with the BSSE accounted for through the counterpoise correction method. Bond energies for bonds with separated charge were not included. Reference energies were determined for each molecule in the test sets using the FPT algorithm. Energies, as opposed to actual bond order and formal charge assignments, are used for reference as the energy of two resonance structures will be identical whereas the bond order and formal charge states will not, even though each resonance state is a valid, and minimum, bond order and formal charge assignment. Accuracies of the non exact algorithms were determined using only the subset of molecules which did not time-out. The genetic algorithm was run using the following parameters: a population size $N_p = 50$ with $N_s = 25$ of the initial population generated using the seeding method, a breeding candidate selection scale parameter value $\xi_b = 0.25$, a mutation rate $\rho_m = 0.1$, a generational progression scale parameter value $\xi_g = 0.25$, a minimum of $m = 25$ generations, maximum of $M = 100$ generations, and a termination criterion of $t = 20$ generations. The A* algorithm

Table 9.4: Efficiency and accuracy results for the four different algorithms. The efficiency measure (⌛) is a count of the total number of molecules which timed-out the algorithm with a 2.5 second time-out period. The accuracy (◎) is the percentage of molecules for which the correct minimum energy electron distribution was determined. Failures due to time-outs are removed from the accuracy calculation. Due to the stochastic nature of the genetic algorithm, it was run five times and the results averaged across the runs. Standard deviations in the results are thus provided.

	MMFF94 (700 total)			
	⌛	σ	◎ %	σ
Local Optimisation	21		76.29	
Genetic Algorithm	120.2	6.40	49.54	0.69
A*	257		100.00	
FPT	188		100.00	
	KEGG (5955 total)			
Local Optimisation	1703		71.19	
Genetic Algorithm	3367.4	10.48	40.95	0.15
A*	3828		100.00	
FPT	2996		100.00	
	ZINC15 (10844)			
Local Optimisation	591		70.77	
Genetic Algorithm	4771.2	22.35	47.28	0.13
A*	5458		100.00	
FPT	2659		100.00	

utilised the abstemious heuristic. Results are given in table 9.4.

Clearly, the local optimisation method is the most efficient optimisation method. It also has fairly good accuracy, which is primarily due to the ordering of P, and is thus well suited to obtaining an heuristically good electron distribution. By contrast, with the parameters used here, the genetic algorithm performs poorly, with only approximately 50% accuracy. The accuracy can be improved through optimisation of the parameters to use, but this will lead to a noticeable decrease in the efficiency of the algorithm as the population size or number of generations would likely have to be increased to give noticeable accuracy improvement. Alternatively, or in addition to parameter optimisation, the genetic algorithm could be combined with the local optimisation algorithm to give a memetic algorithm. A memetic algorithm uses a local search, like the local optimisation method, to reduce the likelihood of premature convergence of the overarching genetic algorithm. Of the two exact algorithms, the FPT algorithm is the most efficient. This becomes even more noticeable if the time-out period is extended as it becomes apparent that the FPT algorithm scales far better than the A* algorithm. Due to the overhead associated

with constructing a nice tree-decomposition, the A* algorithm is more efficient than the FPT algorithm when the molecule has few positions for electrons to be placed.

9.4.5.2. Comparison to Reference Assignments

To compare with the reference assignments of the three molecular databases, the FPT algorithm is used. An electron distribution energy for the reference assignment is calculated using the reference formal charges and bond orders. Results from the FPT algorithm are compared to this energy value. As an energy value is effectively a representation of the counts of various formal charges and bond orders, comparing the energies accounts for different resonance structures produced from the optimisation process which would not match up exactly with the reference assignments but are fundamentally the same assignment. A small number of reference assignments contained bond orders which were not present in the LUT_{bo} lookup table, leading to the reference energy being calculated as ∞. These molecules remain in the data set as such a result will influence all methods equally. Electron distributions are optimised using energies calculated with both the def2-SVPD and def2-TZVPPD basis sets, with and without the BSSE accounted for through counterpoise correction and with and without different energies for bonds with separated charge. Additionally, the energy of the first fifty minimum penalty score bond order assignments returned by the BALL method are calculated and compared with the reference energies. If any of the BALL calculated assignments had an energy matching the reference energy, that molecule was deemed to be correctly calculated by the BALL method. Three forms of the BALL method were used: when the formal charge of the reference assignments were provided to the algorithm, when no formal charges were provided, and when no formal charges were provided but the energy calculated only included contributions from bonds. These energies are only calculated with the def2-SVPD basis set without counterpoise correction and no different energies for bonds with separated charges as the results for them should be independent of the energies used. The results of comparing the energies with the reference assignment energies provide a benchmark test for the FPT algorithm developed and used here. Results obtained are given in table 9.5.

These results show that the FPT algorithm developed here can reproduce reference electron distribution energies with a high degree of reliability, regardless of the level of theory used to calculate atom and bond energies. This is particularly appealing as it means the bond energy and atom energy tables need only be extended at the computationally cheapest level of QM theory. Accounting for the BSSE through counterpoise correction also has no noticeable affect on the accuracy of the results. In only two cases were the results different when BSSE was accounted for, which is not a significant enough difference to justify the vastly increased computational cost

Table 9.5: Percent of optimised electron distribution energies which match the energy calculated for the reference assignment. Energies were calculated with two basis sets, with and with bonds with separated charge having different energies (Q-bonds) and with and without counterpoise correction to the BSSE. For the results utilising the BALL method, energies were calculated with the def2-SVPD basis set.

	Q-bonds	BSSE	MMFF94	KEGG	ZINC15
def2-SVPD	N	N	83.57	97.25	94.32
def2-SVPD	N	Y	83.57	97.25	94.32
def2-SVPD	Y	N	91.00	98.09	96.36
def2-SVPD	Y	Y	91.14	98.09	96.36
def2-TZVPPD	N	N	83.57	97.25	94.32
def2-TZVPPD	N	Y	83.57	97.25	94.32
def2-TZVPPD	Y	N	91.00	98.09	96.36
def2-TZVPPD	Y	Y	91.00	98.09	96.35
BALL w/ FC	N	N	97.43	98.86	98.51
BALL w/o FC	N	N	63.86	84.85	55.36
BALL w/o FC (BO)	N	N	86.14	94.49	96.67

associated with determining the counterpoise correction. The other outcome of note is that the use of different energies for bonds with charge separation leads to a noticeable improvement in the accuracy of the optimised energies, particularly with the MMFF94 validation suite. Primarily this is due to many nitrogen containing molecules being better represented with charged bond energies than without.

Comparison with the BALL method provides a measure of how well the FPT method developed here compares with a current state of the art method. When formal charges are provided to the BALL method, it out performs the FPT method. This is not an entirely fair comparison though as the BALL method is working with additional information which is not only unavailable to our FPT method, but is also being determined in parallel with the bond order assignment. Having said that, the level of accuracy attained is not too dissimilar to that of the BALL method. It is in the case where formal charges have not been supplied to the BALL method were our FPT implementation really shines. This is the situation when bond orders and formal charges need to be determined within the CherryPicker algorithm. Here, our FPT algorithm vastly outperforms the BALL method, though, again, this is not a fair comparison as molecules with any formal charges will not match with the reference assignment. Accounting for only the energy contribution due to bonds is more representative of the true performance of the BALL method. If formal charges were back calculated from the bond order assignment, the true performance of the BALL method would fall somewhere between these two cases. With this in mind, we can conclude that the accuracy of our FPT algorithm matches or exceeds that of the current state of

the art BALL method when formal charges are unknown.

The BALL method does have a distinct advantage over our FPT algorithm, and that is to do with the implementation. The BALL method is implemented in C++ which is a more performant programming language than Python, in which our algorithms are implemented. This performance gap could be closed somewhat by some tweaks to our FPT algorithm focused around calculation of the score table, and optimisation of the nice tree-decomposition. Such tweaks are left for future work, along with possible reimplementation in a lower level programming language, such as C++.

9.5. Conclusion

Four different algorithms for the calculation of optimal bond order and formal charge assignment were developed and implemented, utilising energies calculated with high level QM methods. Of the four, the FPT algorithm showed the best performance in both computational efficiency and accuracy. Results obtained show that there is no difference in accuracy with different levels of QM computation, indicating that extension of the algorithms to be able to handle additional element and bond types need only consider performing QM calculations at the lowest level of theory. In comparison with the state of the art BALL method, the FPT algorithm developed here performs remarkably well, attaining accuracies close to those attained when the BALL method is provided with formal charges from the reference molecules, and exceeding the accuracy attained when no formal charges were provided.

Bibliography

(1) A. K. Rappe, C. J. Casewit, K. S. Colwell, W. A. Goddard and W. M. Skiff, "UFF, a full periodic table force field for molecular mechanics and molecular dynamics simulations", *Journal of the American Chemical Society*, 1992, **114**, 10024–10035.

(2) T. A. Halgren, "Merck molecular force field. I. Basis, form, scope, parameterization, and performance of MMFF94", *Journal of Computational Chemistry*, 1996, **17**, 490–519.

(3) A. A. S. T. Ribeiro, B. A. C. Horta and R. B. de Alencastro, "MKTOP: a Program for Automatic Construction of Molecular Topologies", *Journal of the Brazilian Chemical Society*, 2008, **19**, 1433–1435.

(4) G. A. Kaminski, R. A. Friesner, J. Tirado-Rives and W. L. Jorgensen, "Evaluation and Reparametrization of the OPLS-AA Force Field for Proteins via Comparison with Accurate Quantum Chemical Calculations on Peptides", *The Journal of Physical Chemistry B*, 2001, **105**, 6474–6487.

(5) J. Wang, W. Wang, P. A. Kollman and D. A. Case, "Automatic atom type and bond type perception in molecular mechanical calculations", *Journal of Molecular Graphics and Modelling*, 2006, **25**, 247–260.

(6) J. Wang, R. M. Wolf, J. W. Caldwell, P. A. Kollman and D. A. Case, "Development and testing of a general amber force field", *Journal of Computational Chemistry*, 2004, **25**, 1157–1174.

(7) A. W. Schüttelkopf and D. M. F. van Aalten, "PRODRG: a tool for high-throughput crystallography of protein–ligand complexes", *Acta Crystallographica Section D*, 2004, **60**, 1355–1363.

(8) M. Lundborg and E. Lindahl, "Automatic GROMACS Topology Generation and Comparisons of Force Fields for Solvation Free Energy Calculations", *The Journal of Physical Chemistry B*, 2015, **119**, 810–823.

(9) A. K. Malde, L. Zuo, M. Breeze, M. Stroet, D. Poger, P. C. Nair, C. Oostenbrink and A. E. Mark, "An Automated Force Field Topology Builder (ATB) and Repository: Version 1.0", *Journal of Chemical Theory and Computation*, 2011, **7**, 4026–4037.

(10) S. Canzar, M. El-Kebir, R. Pool, K. Elbassioni, A. K. Malde, A. E. Mark, D. P. Geerke, L. Stougie and G. W. Klau, "Charge Group Partitioning in Biomolecular Simulation", *Journal of Computational Biology*, 2013, **20**, 188–198.

(11) K. Koziara, M. Stroet, A. Malde and A. Mark, "Testing and validation of the Automated Topology Builder (ATB) version 2.0: prediction of hydration free enthalpies", *Journal of Computer-Aided Molecular Design*, 2014, **28**, 221–233.

(12) J. A. Lemkul, W. J. Allen and D. R. Bevan, "Practical Considerations for Building GROMOS-Compatible Small-Molecule Topologies", *Journal of Chemical Information and Modeling*, 2010, **50**, 2221–2235.

(13) R. M. Betz and R. C. Walker, "Paramfit: Automated optimization of force field parameters for molecular dynamics simulations", *Journal of Computational Chemistry*, 2015, **36**, 79–87.

(14) B. T. Miller, R. P. Singh, J. B. Klauda, M. Hodošček, B. R. Brooks and H. L. Woodcock, "CHARMMing: A New, Flexible Web Portal for CHARMM", *Journal of Chemical Information and Modeling*, 2008, **48**, 1920–1929.

(15) E. Vanquelef, S. Simon, G. Marquant, E. Garcia, G. Klimerak, J. C. Delepine, P. Cieplak and F.-Y. Dupradeau, "R.E.D. Server: a web service for deriving RESP and ESP charges and building force field libraries for new molecules and molecular fragments", *Nucleic Acids Research*, 2011, **39**, W511–W517.

(16) L.-P. Wang, T. J. Martinez and V. S. Pande, "Building Force Fields: An Automatic, Systematic, and Reproducible Approach", *The Journal of Physical Chemistry Letters*, 2014, **5**, 1885–1891.

(17) W. L. Jorgensen, J. Chandrasekhar, J. D. Madura, R. W. Impey and M. L. Klein, "Comparison of simple potential functions for simulating liquid water", *The Journal of Chemical Physics*, 1983, **79**, 926–935.

(18) S. Grimme, "A General Quantum Mechanically Derived Force Field (QMDFF) for Molecules and Condensed Phase Simulations", *Journal of Chemical Theory and Computation*, 2014, **10**, 4497–4514.

(19) M. S. Gordon, D. G. Fedorov, S. R. Pruitt and L. V. Slipchenko, "Fragmentation Methods: A Route to Accurate Calculations on Large Systems", *Chemical Reviews*, 2012, **112**, 632–672.

(20) W. D. Cornell, P. Cieplak, C. I. Bayly, I. R. Gould, K. M. Merz, D. M. Ferguson, D. C. Spellmeyer, T. Fox, J. W. Caldwell and P. A. Kollman, "A Second Generation Force Field for the Simulation of Proteins, Nucleic Acids, and Organic Molecules", *Journal of the American Chemical Society*, 1995, **117**, 5179–5197.

(21) N. Foloppe and A. D. MacKerell, Jr., "All-atom empirical force field for nucleic acids: I. Parameter optimization based on small molecule and condensed phase macromolecular target data", *Journal of Computational Chemistry*, 2000, **21**, 86–104.

(22) A. D. MacKerell, D. Bashford, M. Bellott, R. L. Dunbrack, J. D. Evanseck, M. J. Field, S. Fischer, J. Gao, H. Guo, S. Ha, D. Joseph-McCarthy, L. Kuchnir, K. Kuczera, F. T. K. Lau, C. Mattos, S. Michnick, T. Ngo, D. T. Nguyen, B. Prodhom, W. E. Reiher, B. Roux, M. Schlenkrich, J. C. Smith, R. Stote, J. Straub, M. Watanabe, J. Wirkiewicz-Kuczera, D. Yin and M. Karplus, "All-Atom Empirical Potential for Molecular Modeling and Dynamics Studies of Proteins", *The Journal of Physical Chemistry B*, 1998, **102**, 3586–3616.

(23) N. Schmid, A. Eichenberger, A. Choutko, S. Riniker, M. Winger, A. Mark and W. van Gunsteren, "Definition and testing of the GROMOS force-field versions 54A7 and 54B7", *European Biophysics Journal*, 2011, **40**, 843–856.

(24) J. P. M. Jambeck and A. P. Lyubartsev, "Derivation and Systematic Validation of a Refined All-Atom Force Field for Phosphatidylcholine Lipids", *The Journal of Physical Chemistry B*, 2012, **116**, 3164–3179.

(25) J. P. M. Jambeck and A. P. Lyubartsev, "An Extension and Further Validation of an All-Atomistic Force Field for Biological Membranes", *Journal of Chemical Theory and Computation*, 2012, **8**, 2938–2948.

(26) D. Poger, W. F. Van Gunsteren and A. E. Mark, "A new force field for simulating phosphatidylcholine bilayers", *Journal of Computational Chemistry*, 2010, **31**, 1117–1125.

(27) J. P. M. Jambeck and A. P. Lyubartsev, "Another Piece of the Membrane Puzzle: Extending Slipids Further", *Journal of Chemical Theory and Computation*, 2013, **9**, 774–784.

(28) J. Taylor, N. E. Whiteford, G. Bradley and G. W. Watson, "Validation of all-atom phosphatidylcholine lipid force fields in the tensionless {NPT} ensemble", *Biochimica et Biophysica Acta (BBA) - Biomembranes*, 2009, **1788**, 638–649.

(29) J. B. Klauda, R. M. Venable, J. A. Freites, J. W. O'Connor, D. J. Tobias, C. Mondragon-Ramirez, I. Vorobyov, A. D. MacKerell and R. W. Pastor, "Update of the CHARMM All-Atom Additive Force Field for Lipids: Validation on Six Lipid Types", *The Journal of Physical Chemistry B*, 2010, **114**, 7830–7843.

(30) S.-W. Chiu, S. A. Pandit, H. L. Scott and E. Jakobsson, "An Improved United Atom Force Field for Simulation of Mixed Lipid Bilayers", *The Journal of Physical Chemistry B*, 2009, **113**, 2748–2763.

(31) S. E. Feller and A. D. MacKerell, "An Improved Empirical Potential Energy Function for Molecular Simulations of Phospholipids", *The Journal of Physical Chemistry B*, 2000, **104**, 7510–7515.

(32) S. E. Feller, D. Yin, R. W. Pastor and J. MacKerell, A. D., "Molecular dynamics simulation of unsaturated lipid bilayers at low hydration: parameterization and comparison with diffraction studies", *Biophysical Journal*, 1997, **73**, 2269–2279.

(33) O. Berger, O. Edholm and F. Jähnig, "Molecular dynamics simulations of a fluid bilayer of dipalmitoylphosphatidylcholine at full hydration, constant pressure, and constant temperature", *Biophysical Journal*, 1997, **72**, 2002–2013.

(34) Ö. A. Skjevik, B. D. Madej, R. C. Walker and K. Teigen, "LIPID11: A Modular Framework for Lipid Simulations Using Amber", *The Journal of Physical Chemistry B*, 2012, **116**, 11124–11136.

(35) C. J. Dickson, B. D. Madej, Ö. A. Skjevik, R. M. Betz, K. Teigen, I. R. Gould and R. C. Walker, "Lipid14: The Amber Lipid Force Field", *Journal of Chemical Theory and Computation*, 2014, **10**, 865–879.

(36) J. Gallier, *Discrete Mathematics, Second Edition In Progress*, Springer, 2nd edn., 2017.

(37) R. Diestel, *Graph Theory*, Springer-Verlang Berline Heidelberg, 5th edn., 2016.

(38) M. Nic, J. Jirat, D. of Chemical Nomenclature, S. R. I. U. of Pure, A. Chemistry and B. Kosata, *IUPAC goldbook*, IUPAC, 2006.

(39) R. S. Cahn, C. Ingold and V. Prelog, "Specification of Molecular Chirality", *Angewandte Chemie International Edition in English*, 1966, **5**, 385–415.

(40) V. Prelog and G. Helmchen, "Basic Principles of the CIP-System and Proposals for a Revision", *Angewandte Chemie International Edition in English*, 1982, **21**, 567–583.

(41) N. M. O'Boyle, M. Banck, C. A. James, C. Morley, T. Vandermeersch and G. R. Hutchison, "Open Babel: An open chemical toolbox", *Journal of Cheminformatics*, 2011, **3**, 33–33.

(42) K. Paton, “An Algorithm for Finding a Fundamental Set of Cycles of a Graph”, *Communications of the ACM*, 1969, **12**, 514–518.

(43) A. A. Hagberg, D. A. Schult and P. J. Swart, Proceedings of the 7th Python in Science Conference (SciPy2008), Pasadena, CA USA, 2008, pp. 11–15.

(44) L. P. Cordella, P. Foggia, C. Sansone and M. Vento, “A (sub)graph isomorphism algorithm for matching large graphs”, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2004, **26**, 1367–1372.

(45) C. Oostenbrink, A. Villa, A. E. Mark and W. F. Van Gunsteren, “A biomolecular force field based on the free enthalpy of hydration and solvation: The GROMOS force-field parameter sets 53A5 and 53A6”, *Journal of Computational Chemistry*, 2004, **25**, 1656–1676.

(46) *SDBSWeb*, accessed June 2017, `http://sdbs.db.aist.go.jp` (visited on).

(47) R. J. Abraham and M. Mobli, *Modelling 1H NMR Spectra of Organic Compounds: Theory, Applications and NMR Prediction Software*, Wiley, 2008.

(48) A. R. Leach, D. P. Dolata and K. Prout, “Automated conformational analysis and structure generation: algorithms for molecular perception”, *Journal of Chemical Information and Computer Sciences*, 1990, **30**, 316–324.

(49) E. C. Meng and R. A. Lewis, “Determination of molecular topology and atomic hybridization states from heavy atom coordinates”, *Journal of Computational Chemistry*, 1991, **12**, 891–898.

(50) J. C. Baber and E. E. Hodgkin, “Automatic assignment of chemical connectivity to organic molecules in the Cambridge Structural Database”, *Journal of Chemical Information and Computer Sciences*, 1992, **32**, 401–406.

(51) E. Lang, C.-W. von der Lieth and T. Förster, “Automatic assignment of bond orders based on the analysis of the internal coordinates of molecular structures”, *Analytica Chimica Acta*, 1992, **265**, 283–289.

(52) M. Hendlich, F. Rippmann and G. Barnickel, “BALI: Automatic Assignment of Bond and Atom Types for Protein Ligands in the Brookhaven Protein Databank”, *Journal of Chemical Information and Computer Sciences*, 1997, **37**, 774–778.

(53) M. Froeyen and P. Herdewijn, “Correct Bond Order Assignment in a Molecular Framework Using Integer Linear Programming with Application to Molecules Where Only Non-Hydrogen Atom Coordinates Are Available”, *Journal of Chemical Information and Modeling*, 2005, **45**, 1267–1274.

(54) P. Labute, "On the Perception of Molecules from 3D Atomic Coordinates", *Journal of Chemical Information and Modeling*, 2005, **45**, 215–221.

(55) A. K. Dehof, A. Rurainski, Q. B. A. Bui, S. Böcker, H.-P. Lenhof and A. Hildebrandt, "Automated bond order assignment as an optimization problem", *Bioinformatics*, 2011, **27**, 619–625.

(56) S. Goto, Y. Okuno, M. Hattori, T. Nishioka and M. Kanehisa, "LIGAND: database of chemical compounds and reactions in biological pathways", *Nucleic Acids Research*, 2002, **30**, 402–404.

(57) J. J. Irwin and B. K. Shoichet, "ZINC - A Free Database of Commercially Available Compounds for Virtual Screening", *Journal of Chemical Information and Modeling*, 2005, **45**, 177–182.

(58) J. J. Irwin, T. Sterling, M. M. Mysinger, E. S. Bolstad and R. G. Coleman, "ZINC: A Free Tool to Discover Chemistry for Biology", *Journal of Chemical Information and Modeling*, 2012, **52**, 1757–1768.

(59) T. Sterling and J. J. Irwin, "ZINC 15 – Ligand Discovery for Everyone", *Journal of Chemical Information and Modeling*, 2015, **55**, 2324–2337.

(60) A. E. Reed, L. A. Curtiss and F. Weinhold, "Intermolecular interactions from a natural bond orbital, donor-acceptor viewpoint", *Chemical Reviews*, 1988, **88**, 899–926.

(61) P. Piecuch, S. A. Kucharski, K. Kowalski and M. Musiał, "Efficient computer implementation of the renormalized coupled-cluster methods: The R-CCSD[T], R-CCSD(T), CR-CCSD[T], and CR-CCSD(T) approaches", *Computer Physics Communications*, 2002, **149**, 71–96.

(62) P. Piecuch and M. Włoch, "Renormalized coupled-cluster methods exploiting left eigenstates of the similarity-transformed Hamiltonian", *The Journal of Chemical Physics*, 2005, **123**, 224105:1–10.

(63) S. Boys and F. Bernardi, "The calculation of small molecular interactions by the differences of separate total energies. Some procedures with reduced errors", *Molecular Physics*, 1970, **19**, 553–566.

(64) F. Weigend and R. Ahlrichs, "Balanced basis sets of split valence, triple zeta valence and quadruple zeta valence quality for H to Rn: Design and assessment of accuracy", *Physical Chemistry Chemical Physics*, 2005, **7**, 3297–3305.

(65) M. W. Schmidt, K. K. Baldridge, J. A. Boatz, S. T. Elbert, M. S. Gordon, J. H. Jensen, S. Koseki, N. Matsunaga, K. A. Nguyen, S. Su, T. L. Windus, M. Dupuis and J. A. Montgomery, "General atomic and molecular electronic structure system", *Journal of Computational Chemistry*, 1993, **14**, 1347–1363.

(66) M. S. Gordon and M. W. Schmidt, in, ed. C. E. Dykstra, G. Frenking, K. S. Kim and G. E. Scuseria, Elsevier, 2005, ch. Advances in electronic structure theory: GAMESS a decade later, pp. 1167–1189.

(67) C. Møller and M. Plesset, "Note on an approximation treatment for many-electron systems", *Phys. Rev.*, 1934, **46**, 618–622.

(68) P. E. Hart, N. J. Nilsson and B. Raphael, "A Formal Basis for the Heuristic Determination of Minimum Cost Paths", *IEEE Transactions on Systems Science and Cybernetics*, 1968, **4**, 100–107.

(69) V. Gogate and R. Dechter, "A Complete Anytime Algorithm for Treewidth", *ArXiv e-prints*, 2012.

(70) T. van Dijk, J. P. van den Heuvel and W. Slob, *Computing treewidth with libtw*, tech. rep., University of Utrecht, 2006.

Part III.

Monolayers

10. Introduction

10.1. Naphthalimides

In sensing applications, fluorescence is seen as a valuable property due to its rapid response times and high sensitivity. Naphthalimide structures, such as the 1,8-naphthalimide in figure 10.1, have absorption and fluorescence emission spectra lying within the UV and visible regions.[1] Various photo-physical properties can be fine tuned through careful structural design as synthetic modifications are readily accommodated on either the aromatic naphthalene moiety (generally at the 4 position), or at the N-imide site. Such functionalised naphthalimides have been utilised in a wide range of applications such as hydrogen sulfide (H_2S) detection,[2,3] detection of mercury and methyl mercury ions[4] and DNA binding.[5]

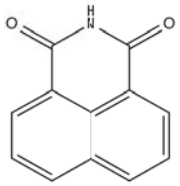


Figure 10.1: 1,8-naphthalimide structure

Substitution at the N-imide site with a long chain alkane, such as octadecane ($C_{18}H_{38}$) results in an amphiphilic naphthalimide. When placed on the surface of a sub-phase, such as water, amphiphilic molecules arrange themselves such that their hydrophilic ends or head group, the naphthalimide moiety, are solvated whilst their hydrophobic ends or tails, the alkyl chain, remain oriented away from the sub-phase. This orientation process leads to a single molecule thick layer of the amphiphile forming on the surface; a monolayer.

Experimentally, monolayers of these alkyl functionalised naphthalimide molecules can be immobilised on a solid substrate using the Langmuir-Blodgett technique. To do so, a Langmuir trough is prepared with a water sub-phase and a sample of the naphthalimide molecules added to disperse across the the air–sub-phase interface surface. Barriers slowly move in along the surface of the sub-phase, compressing the naphthalimide molecules into an ordered monolayer. Transfer to a solid substrate occurs when the substrate is passed through the compressed monolayer.

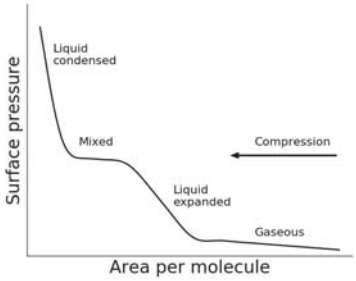


Figure 10.2: An idealised depiction of a pressure-area isotherm for a long chain organic compound monolayer. Upon compression of the monolayer, the monolayer phase transitions from a gaseous state to a condensed state, through a liquid expanded state.

10.2. Monolayers

Molecules on the surface of liquids experience an imbalance of forces compared to the bulk as they have a greater attraction to other molecules on the surface than to molecules in the air above. This imbalance is known as surface tension (γ). Rather than quote the surface tension, more often the surface pressure (Π) is reported. Surface pressure is expressed as the reduction of the surface tension of pure water (γ_0) due to the monolayer (γ_m):

$$\Pi = \gamma_0 - \gamma_m. \tag{10.1}$$

Monolayers exhibit phase transitions on compression, which are generally characterised by pressure–area isotherms. An idealised pressure-area isotherm is shown in figure 10.2. In the gaseous phase, molecules are spread out over a large area with minimal interaction with one another. As the monolayer is compressed, molecules move into a liquid expanded phase where the tails adopt random orientations, with no long range translational or orientational order. Further compression leads to the liquid condensed phase with greater orientational order as the molecule tails line up away from the water in order to achieve close packing, leading to areas close to the cross sectional area of the molecule. Compression past this point leads to monolayer collapse as the monolayer buckles and folds in on itself forming multi layers. The horizontal plateau corresponds to the first order phase transition from liquid expanded to liquid condensed phases. Such plateaus are not always seen in isotherms due to impurities or higher order phase transitions.[6]

10.2.1. Monolayer Simulation

From a molecular dynamics simulation, the surface tension can be calculated as the ensemble average

$$\gamma = \left\langle \frac{L_z}{n} \left(P_{zz} - \frac{P_{xx} + P_{yy}}{2} \right) \right\rangle \tag{10.2}$$

where L_z is the length of the simulation box in the z direction, n is the number of surfaces, and P_{xx}, P_{yy} and P_{zz} are the x, y and z components of the pressure tensor respectively. Simulation of a monolayer at the air–water interface generally proceeds as follows. A monolayer of the molecule or molecules is created, laying across the xy plane, and solvated such that the head group interacts with the solvent. Sufficient solvent needs to be added to ensure that the monolayer molecules only see one solvent surface, i.e. the surface they are interacting with. The length of the simulation box in the z direction is extended to give a vacuum above the monolayer (as opposed to air) and ensure that the tails do not interact with the periodic image of the bulk solvent. Simulation is then carried out under a relevant thermodynamic ensemble.

Three common thermodynamic ensembles (section 1.1.2.3) used for monolayer simulations are the NPT, NγT, and NVT ensembles. In all ensembles, the number of particles (N) is maintained and the temperature (T) is maintained by coupling to an external temperature bath at a given target temperature. In the NPT ensemble the pressure is maintained through coupling to an external pressure bath at a target pressure. Coupling can be applied in either an anisotropic manner, where each box vector is coupled independently, or a semi-isotropic manner, where x and y box vectors and coupled together and the z box vector is coupled independently. The NγT ensemble is similar to the NPT ensemble in that it involves coupling to an external pressure bath. However, as opposed to coupling the box vectors to the pressure, the surface tension is coupled to the xy plane of the box and the z box vector is coupled as in the NPT ensemble.

Previous computational work on pressure-area isotherms has focused on lipid monolayers, in particular dipalmitoyl phosphatidylcholine (DPPC) monolayers.[7–13] Although not directly comparable to naphthalimide isotherms, as their head groups are more polar than naphthalimide head groups and they have two hydrophobic tails as opposed to one, such simulations do provide insight into the successes and limitations of molecular dynamics simulations. Baoukina *et al.* ran coarse grained simulations in the NPT ensemble for a large monolayer (4096 lipid molecules) at 300 K. They were able to distinguish four distinct phases in the simulated isotherm, indicating the ability for molecular dynamics simulations to provide insight into the full range of system states expected for a monolayer isotherm.[10] Duncan and Larson investigated the effect of ensemble in coarse grained simulations. They found that the NVT ensemble does not allow for

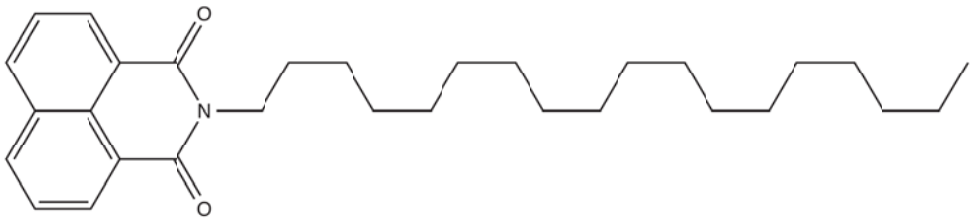


Figure 10.3: Structure of the naphthalimide molecule investigated here.

sufficient pressure relaxation[7], which has also been reported by others,[11] leading to predictions of larger surface pressures at a given area than those predicted by simulations run with the NγT ensemble.

Another common theme of molecular dynamics simulations of monolayers is determining the best way to accurately determine surface pressure for simulated isotherms. The surface tension of SPC water has been computationally predicted to be $54.7\,\mathrm{mN\,m^{-1}}$ from simulations at 300 K,[14] whereas experimentally it has been measured at $71.20\,\mathrm{mN\,m^{-1}}$ at 303 K.[15] This discrepancy can be primarily attributed to the difference between surface tension of water at an air interface, which is the case for experimental measurements, and surface tension of water at a vacuum interface, which is the case for simulations. If the computational value is used in determining the surface pressure, significantly lower surface pressure values would be determined. The general consensus is that use of the experimental value would not take away from the accuracy of simulation results as simulations are primarily concerned with measuring the headgroup–water and tail–vacuum interactions, which are not affected by water–vacuum interactions.[7,8]

Previous work carried out by an MSc student from the University of Southampton, Dasha Draper,[16] who visiting the Allison group in 2016 as part of a collaboration with Dr Jon Kitchen, involved investigation of the pressure–area isotherms for a number of functionalised *N*-steryl naphthalimides in the NVT ensemble. However, the results obtained were disappointing due to deficiencies in the simulation conditions. As a consequence it was decided to investigate a more robust simulation methods, beginning with the choice of ensemble. This section of work investigates the effect of ensemble on the properties of a naphthalimide monolayer through compression. The naphthalimide investigated is *N*-steryl naphthalimide shown in figure 10.3. Monolayers are produced and simulated using the NVT, NPT and NγT ensembles in duplicate, generating pressure–area isotherms. Further, the properties of the monolayers at different surface pressures and areas are compared.

11. Methods

11.1. Parameterisation

Parameters for the naphthalimide molecule shown in figure 10.3 were produced through manual application of the CherryPicker algorithm described in part II. Bonded and non-bonded parameters for the alkyl chain were taken from Poger *et al.* lipid tail parameters.[17] Parameters for the naphthalimide moiety were determined based on existing phenylalanine and tryptophan parameters in the GROMOS 54A7 force field.[18] As there were no existing reference parameters for the imide bridge portion of the molecule, parameters were produced by averaging across a number of different parameter sets obtained from the ATB[19–21] containing the correct molecule fragment, and slightly refined to obtain integer charge groups. The parameters produced are given in appendix E.

11.2. Computational Details

11.2.1. System Construction

The naphthalimide structure file was oriented such that the tail of the molecule was aligned with the z-axis. It was then tessellated onto an evenly spaced twelve-by-twelve grid to give an area per molecule of $0.40\,\mathrm{nm}^2$. Solvation was performed by adding a 4.2 nm slab of SPC water below the naphthalimide monolayer beginning from the plane formed by the nitrogen atoms of the imide bridge. This resulted in 5680 water molecules being added. To keep the naphthalimide tails from interacting with the periodic image of the water slab, the size of the simulation box in the z direction was extended by 80 nm in order to introduce a tail–vacuum interface. The gaseous phase of a monolayer is computationally expensive to calculate as it requires a large amount of separation between neighbouring naphthalimide molecules and consequently a large volume of water molecules, and not of major interest. As such, the system construction is designed to mostly ignore this phase.

11.2.2. Simulation Conditions

All simulations were performed using GROMACS version 2016.1[22–28] with periodic boundary conditions applied in all directions. A simulation time step of 2 fs was utilised. In all cases, bonds were constrained to their equilibrium values using the LINCS algorithm.[29,30] Neighbour searching was performed using the Verlet buffer scheme, with a single range cut-off of 1.4 nm. Beyond the cut-off distance, long range electrostatic interactions were approximated with a reaction field scheme.[31,32] Temperature was maintained by coupling the solvent and monolayer to separate external baths using a velocity rescaling thermostat[33] with a coupling constant τ of 0.1 ps. After performing an energy minimisation, initial velocities were randomly generated by selecting from a Maxwell distribution at 50 K. After 10 ps of simulation at 50 K, the temperature was gradually increased to 298 K over a period of 200 ps. The first 1 ns of simulation, including the time used to increase the temperature, was performed with position restraints applied to the monolayer atoms. These restraints were then removed and a further 80 ns of simulation undertaken. All simulations were performed in duplicate with different random initial velocities. The final 50 ns of simulation were used for analysis.

11.2.3. NPT Simulations

NPT simulations were performed with semi-isotropic pressure coupling, that is the pressure in the *xy* plane was coupled independently of the *z* axis. Coupling was to an external bath using a Berendsen barostat[34] with a coupling constant of 1.0 ps. Compressibility was set to 4.5 Pa^{-1} in the *xy* plane and 0 Pa^{-1} along the *z* axis in order to maintain a constant box height. Simulations were undertaken with reference pressures of 750, 500, 250, 0, -100, -200, -300, -400, -500, -600, -750 and -1000 kPa in the *xy* plane and 100 kPa along the *z* axis. These pressure values were chosen based on pressures obtained in previous NVT based simulations.[16]

11.2.4. NγT Simulations

NγT simulations were performed with surface tension coupling for surfaces parallel to the *xy*-plane. Like the NPT simulations, compressibility was set to 4.5 Pa^{-1} in the *xy* plane and 0 Pa^{-1} along the *z* axis in order to maintain a constant box height. Reference surface tensions used were 65, 80, 95, 100, 105, 107.5, 110, 112.5, 115, 117.5, 120, 122.5, 125, 127.5, and 130 $\mathrm{mN\,m}^{-1}$.

11.2.5. NVT Simulations

NVT simulations were performed at constant volume, thus there was no pressure coupling. However, initial conformations were obtained by selecting simulation frames from the initial 1 ns of

simulation in the NγT ensemble with areas per molecule close to the desired values. Based on experimental pressure-area isotherms, target areas per molecule were selected at 16 evenly spaced points between 0.266 nm^2 and 0.476 nm^2 inclusive.

11.3. Surface Pressure Calculation

Surface pressures were calculated from each simulation as per equation 10.1. A reference value of 71.90 $mN\,m^{-1}$ was used for γ_0, which was obtained by interpolation of available experimental surface tension values at temperatures from 0 °C to 100 °C.[15] As the simulation method involved only a single naphthalimide monolayer, and thus also a single water–vacuum interface, the surface tension of the monolayer was extracted from each simulation as

$$\gamma_m = \gamma - \gamma_0 \tag{11.1}$$

where γ is the surface tension as calculated by equation 10.2. This approach has been previously utilised by Mohammad-Aghaie *et al.*[8] The values were averaged across all simulations performed using the same conditions. Estimates of the error in both the surface pressure and area values were determined as follows. Data from each simulation was divided into five evenly sized blocks, and the block mean calculated. For each block, the absolute deviation from the overall mean across all simulations was determined. The error was then set to the mean of these deviations.

12. Results and Discussion

The pressure–area isotherms calculated for the naphthalimide monolayer simulated under three different ensembles, with varying values of the volume (NVT), pressure in the *xy*-plane (NPT), or surface tension (NγT) are shown in figure 12.1. Each of the ensembles produce pressure–area isotherms around different portions of the compression space. By design, the NVT ensemble evenly covers areas between 0.26 nm^2 and 0.48 nm^2. The reference pressures to which the pressure in the *xy*-plane was coupled in the NPT ensemble simulations caused the areas to be much lower than the initial 0.40 nm^2, from 0.10 nm^2 to 0.27 nm^2. Although these reference pressure values enabled sampling of the highly compressed states, this indicates that the pressure values

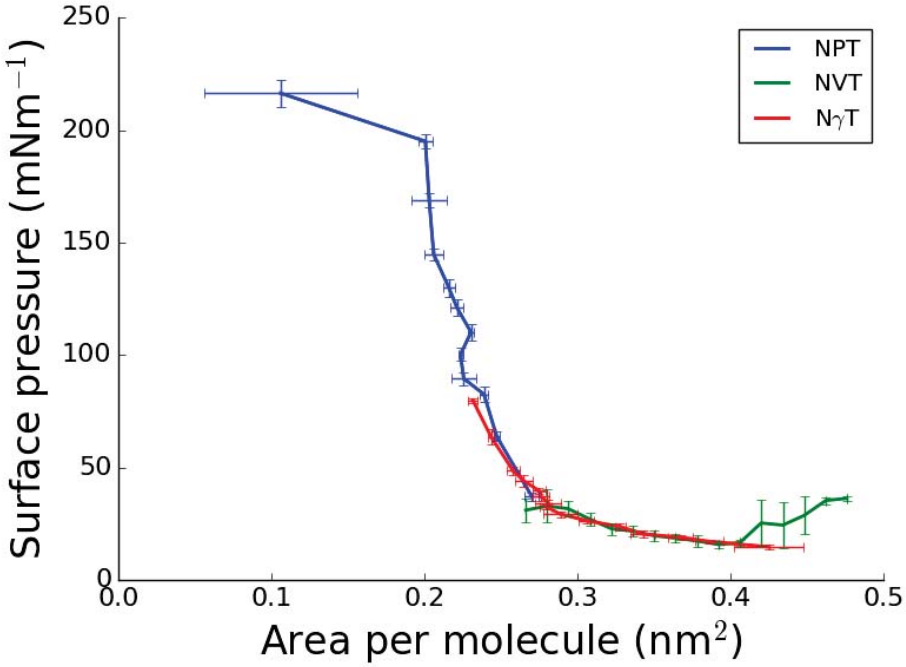


Figure 12.1: Pressure–area isotherms plots the naphthalimide monolayer simulated in three different ensembles. Error bars show the average deviation of the block-averaged mean fromt he overall mean for each datum.

used for coupling were poorly chosen, as they did not allow for sampling at the more desirable lower compression levels. From the pressure measured in the NγT simulations, a more appropriate range of pressure values would be between $-600\,\mathrm{kPa}$ and $-1200\,\mathrm{kPa}$, which should give areas of between $0.20\,\mathrm{nm}^2$ and $0.45\,\mathrm{nm}^2$. Based on the slope of the isotherm, the NPT ensemble simulations gave monolayers primarily in the liquid condensed phase. The NγT ensemble shows an isotherm closer to the ideal case. There is a relatively even compression of the monolayer from $0.42\,\mathrm{nm}^2$ to $0.23\,\mathrm{nm}^2$, covering transition from the gaseous phase through to the liquid condensed phase.

In the regions where the various ensembles have produced overlapping areas, they converge to the same result, indicating that any of the ensembles should be capable of producing reasonable pressure–area isotherms. Outside of overlapping regions, the NPT ensemble continues to converge nicely, as indicated by the small error estimates, except at the smaller areas, where area convergence degrades. At the smaller areas, the area per molecule approaches or exceeds the cross sectional area of the naphthalimide molecules, causing monolayer buckling or collapse. On the other hand, the larger areas found with the NVT ensemble show very poor surface pressure convergence. As the simulation box has expanded in the *xy*-plane relative to the initial conformational set-up, the water layer is not as thick in these simulations, potentially leading to the monolayer 'seeing' the water–vacuum interface.

From these results, it can be concluded that the NγT ensemble should be used going forward. Even though all ensembles should be able to produce reasonable results, the NγT ensemble allows for fine control of surface pressure and area results, as well as exceptional convergence. As such, all further analysis focuses exclusively on the NγT results.

12.1. Monolayer Structural Properties

Visual inspection of the monolayers, as shown in figure 12.2, shows that initially, the monolayer is in a liquid expanded phase, with random tail orientations, and consequently is thin. As the monolayer is compressed, the tails orient themselves away from the water sub-phase, increasing the thickness of the monolayer. Though the simulations boxes used here are relatively small, the transition from liquid expanded to liquid condensed phase can be seen through the mixed phase present, for example with a coupling surface tension of $112.5\,\mathrm{mN\,m^{-1}}$, where there are regions of the monolayer with distinctly more ordered tails, as well as the randomly ordered tails.

One means of numerically showing this progression to a more ordered phase is through the deuterium order parameter S_{CD}. The deuterium order parameter provides a measure of the relative orientation of individual C$-$D bonds with respect to the bilayer normal. The order parameter

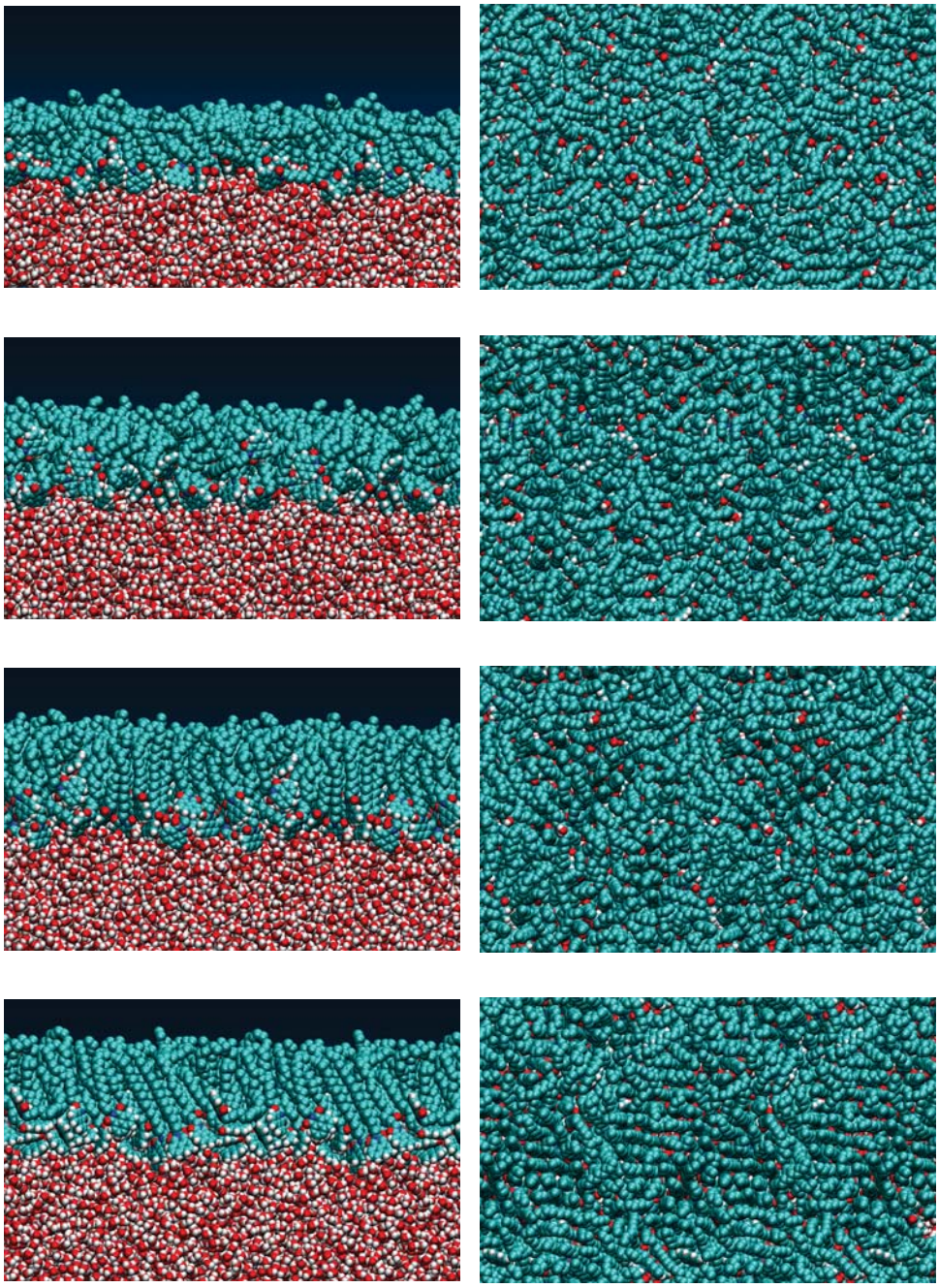


Figure 12.2: Snapshots from simulations showing cross sectional (left) and top down (right) views of the monolayers produced with surface tension coupling values of (from bottom to top) $95.0\,\mathrm{mN\,m^{-1}}$, $112.5\,\mathrm{mN\,m^{-1}}$, $122.5\,\mathrm{mN\,m^{-1}}$ and $130.0\,\mathrm{mN\,m^{-1}}$, with areas per molecule of $0.2579\,\mathrm{nm^2}$, $0.2792\,\mathrm{nm^2}$, $0.3426\,\mathrm{nm^2}$ and $0.4248\,\mathrm{nm^2}$ respectively.

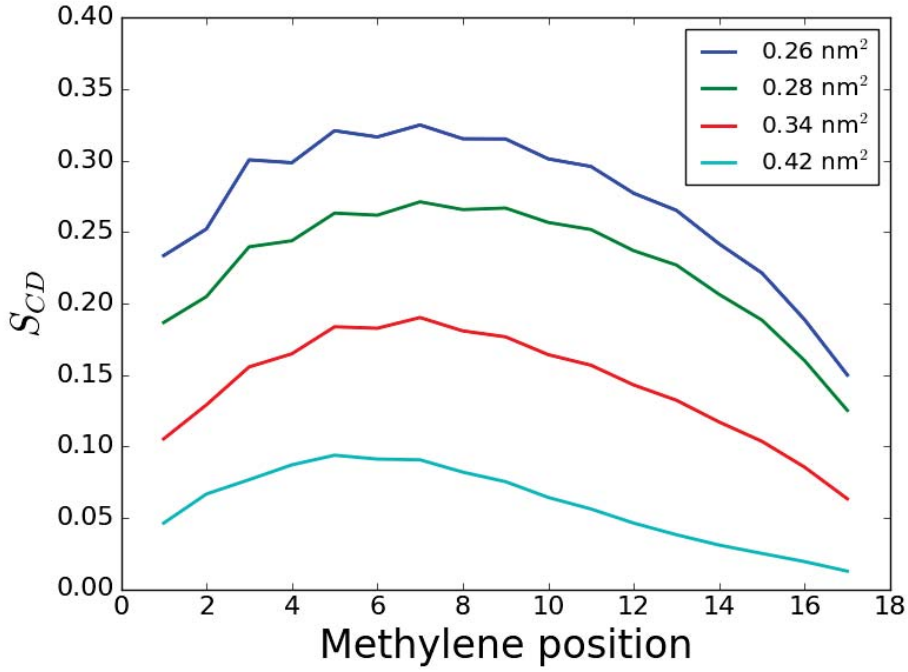


Figure 12.3: Deuterium order parameter S_{CD} profile of the alkyl tail of the naphthalimide monolayer. The S_{CD} values are averaged over all naphthalimide molecules. Methylene positions are numbered beginning from the methylene bound to the imide bridge.

S^i_{CD} of a methylene at position i is defined as:

$$S^i_{CD} = \frac{1}{2}\left\langle 3\cos^2\theta_i - 1\right\rangle \tag{12.1}$$

where θ_i is the angle between a C−D vector of the ith methylene in an alkyl chain and the normal to the monolayer or z axis. As the GROMOS 54A7 force field uses a UA representation in which aliphatic hydrogen atoms are incorporated into the carbons to which they are bound, the positions of the hydrogen atoms were constructed based on the positions of the neighbouring atoms, assuming ideal tetrahedral geometry. Values of S^i_{CD} range between 1 and -0.5, where a value of 1 corresponds to total order along the z axis, a value of 0 is isotropic orientation, i.e. no overall orientation, and a value of -0.5 is total order perpendicular to the z axis.

Figure 12.3 shows the progression of the deuterium order parameter as the monolayer is compressed from an area per molecule of 0.42 nm^2 to 0.26 nm^2. The monolayer starts out with a large degree of isotropic order, particularly with the methylene carbons close to the terminus. As the area per molecule decreases, the alkyl chains orient themselves with respect to the mono-

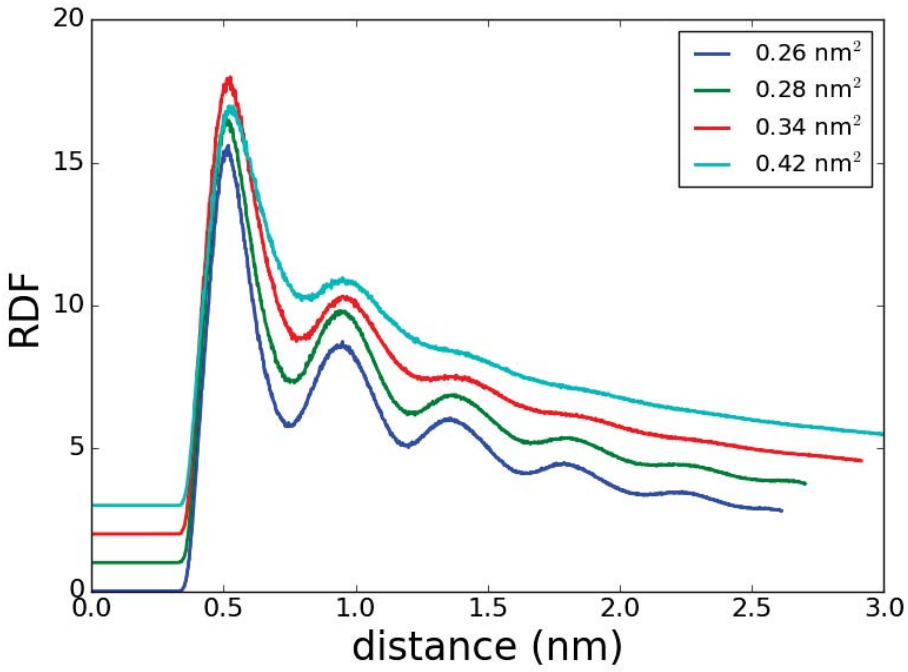


Figure 12.4: Radial distribution function of the ninth carbon from the imide bridge. Different areas per molecule show the effect of compression on the RDF. Larger areas per molecule are progressively shifted up by 1 to allow for easier comparison of peaks.

layer normal, leading to a steadily increasing deuterium order parameter.

Complementary to the deuterium order parameter analysis is the radial distribution function (RDF). Whereas deuterium order parameters describe the orientational order of alkyl chains, the RDF is commonly used to describe the translational ordering of liquids. This is important as though a deuterium order parameter of zero commonly indicates isotropic ordering, it can also be attained with a perfectly ordered system oriented at the magic angle of 54.7°.[35] The RDF effectively gives the probability of finding a specified particle at a radius r from the original particle. At small values of r, the RDF is zero, due to the size of molecules and that molecules cannot occupy the same space. As r increases, the RDF varies in a periodic fashion. The periodicity implies that surrounding molecules can occupy shells around the central molecule. Observing distinct peaks at large values of r indicates a high level of translational ordering within the system.

Figure 12.4 shows the RDFs of the ninth carbon from the imide bridge as the monolayer is compressed from an area per molecule of 0.42 nm^2 to 0.26 nm^2. This carbon is chosen as it is near the peak of the deuterium order parameter profile at small areas per molecule. At large

areas per molecule, the RDF shows a distinct initial peak and a slight secondary peak before decaying away. As the area per molecule decreases, additional distinct peaks become apparent, at regular intervals, and the secondary peak becomes more prominent. At an area per molecule of $0.26\,\mathrm{nm}^2$, there are four easily discernible peaks at regular intervals, with a slight fifth peak visible. This indicates that not only do the alkyl chains have increased orientational order on compression, but that they also have increased translational order.

12.2. Conclusion

The results presented here show that the three ensembles investigated produce similar results for the pressure–area isotherms, excluding large areas per molecule in the case of the NVT ensemble and small areas per molecule for the NPT ensemble. As such, it is recommended that further work on these naphthalimide monolayer systems proceed utilising the NγT ensemble. Using this ensemble, simulations were able to reproduce the compression behaviour of a monolayer. That is, transitioning from a liquid expanded phase to a liquid condensed phase via a mixed phase. This ability was confirmed through visual analysis, and the progression of the deuterium order parameters and the RDF of a central carbon atom during compression.

Bibliography

(1) R. M. Duke, E. B. Veale, F. M. Pfeffer, P. E. Kruger and T. Gunnlaugsson, "Colorimetric and fluorescent anion sensors: an overview of recent developments in the use of 1,8-naphthalimide-based chemosensors", *Chemical Society Reviews*, 2010, **39**, 3936–3953.

(2) S.-A. Choi, C. S. Park, O. S. Kwon, H.-K. Giong, J.-S. Lee, T. H. Ha and C.-S. Lee, "Structural effects of naphthalimide-based fluorescent sensor for hydrogen sulfide and imaging in live zebrafish", *Scientific Reports*, 2016, **6**, 26203:1–10.

(3) Z. Luo, K. Yin, Z. Yu, M. Chen, Y. Li and J. Ren, "A fluorescence turn-on chemosensor for hydrogen sulfate anion based on quinoline and naphthalimide", *Spectrochimica Acta Part A: Molecular and Biomolecular Spectroscopy*, 2016, **169**, 38–44.

(4) Y. Yang, Z. Yao, W. Li, K. Chen, L. Liu and H.-C. Wu, "Selective detection of mercury(II) and methylmercury(II) via coordination-induced emission of a small-molecule probe", *Science China Chemistry*, 2016, **59**, 1651–1657.

(5) M. Brana, M. Cacho, M. A. Garcia, B. de Pascual-Teresa, A. Ramos, M. T. Dominguez, J. M. Pozuelo, C. Abradelo, M. F. Rey-Stolle, M. Yuste, M. Banez-Coronel and J. C. Lacal, "New Analogues of Amonafide and Elinafide, Containing Aromatic Heterocycles: Synthesis, Antitumor Activity, Molecular Modeling, and DNA Binding Properties", *Journal of Medicinal Chemistry*, 2004, **47**, 1391–1399.

(6) M. Phillips and D. Chapman, "Monolayer characteristics of saturated 1,2-diacyl phosphatidylcholines (lecithins) and phosphatidylethanolamines at the air-water interface", *Biochimica et Biophysica Acta (BBA) - Biomembranes*, 1968, **163**, 301–313.

(7) S. L. Duncan and R. G. Larson, "Comparing Experimental and Simulated Pressure-Area Isotherms for DPPC", *Biophysical Journal*, 2008, **94**, 2965–2986.

(8) D. Mohammad-Aghaie, E. Mace, C. A. Sennoga, J. M. Seddon and F. Bresme, "Molecular Dynamics Simulations of Liquid Condensed to Liquid Expanded Transitions in DPPC Monolayers", *The Journal of Physical Chemistry B*, 2010, **114**, 1325–1335.

(9) H. Dominguez, A. M. Smondyrev and M. L. Berkowitz, "Computer Simulations of Phosphatidylcholine Monolayers at Air/Water and CCl4/Water Interfaces", *The Journal of Physical Chemistry B*, 1999, **103**, 9582–9588.

(10) S. Baoukina, L. Monticelli, S. J. Marrink and D. P. Tieleman, "Pressure-Area Isotherm of a Lipid Monolayer from Molecular Dynamics Simulations", *Langmuir*, 2007, **23**, 12617–12623.

(11) A. W. Mauk, E. L. Chaikof and P. J. Ludovice, "Structural Characterization of Self-Assembled Lipid Monolayers by NπT Simulation", *Langmuir*, 1998, **14**, 5255–5266.

(12) A. Olzynska, M. Zubek, M. Roeselova, J. Korchowiec and L. Cwiklik, "Mixed DPPC/POPC Monolayers: All-atom Molecular Dynamics Simulations and Langmuir Monolayer Experiments", *Biochimica et Biophysica Acta (BBA) - Biomembranes*, 2016, **1858**, 3120–3130.

(13) A. J. Kox, J. P. J. Michels and F. W. Wiegel, "Simulation of a lipid monolayer using molecular dynamics", *Nature*, 1980, **287**, 317–319.

(14) C. Vega and E. de Miguel, "Surface tension of the most popular models of water by using the test-area simulation method", *The Journal of Chemical Physics*, 2007, **126**, 154707:1–10.

(15) *CRC Handbook of Chemistry and Physics*, ed. D. R. Lide, CRC Press, 2001.

(16) D. Draper, MSc in Chemsitry, "Producing Surface Pressure-Area Isotherms for a set of Functionalised Naphthalimides Using Molecular Dynamics Simulations", *University of Southampton*, 2017.

(17) D. Poger, W. F. Van Gunsteren and A. E. Mark, "A new force field for simulating phosphatidylcholine bilayers", *Journal of Computational Chemistry*, 2010, **31**, 1117–1125.

(18) N. Schmid, A. Eichenberger, A. Choutko, S. Riniker, M. Winger, A. Mark and W. van Gunsteren, "Definition and testing of the GROMOS force-field versions 54A7 and 54B7", *European Biophysics Journal*, 2011, **40**, 843–856.

(19) S. Canzar, M. El-Kebir, R. Pool, K. Elbassioni, A. K. Malde, A. E. Mark, D. P. Geerke, L. Stougie and G. W. Klau, "Charge Group Partitioning in Biomolecular Simulation", *Journal of Computational Biology*, 2013, **20**, 188–198.

(20) K. Koziara, M. Stroet, A. Malde and A. Mark, "Testing and validation of the Automated Topology Builder (ATB) version 2.0: prediction of hydration free enthalpies", *Journal of Computer-Aided Molecular Design*, 2014, **28**, 221–233.

(21) A. K. Malde, L. Zuo, M. Breeze, M. Stroet, D. Poger, P. C. Nair, C. Oostenbrink and A. E. Mark, "An Automated Force Field Topology Builder (ATB) and Repository: Version 1.0", *Journal of Chemical Theory and Computation*, 2011, **7**, 4026–4037.

(22) M. J. Abraham, T. Murtola, R. Schulz, S. Páll, J. C. Smith, B. Hess and E. Lindahl, "GROMACS: High performance molecular simulations through multi-level parallelism from laptops to supercomputers", *SoftwareX*, 2015, **1–2**, 19–25.

(23) H. Berendsen, D. van der Spoel and R. van Drunen, "GROMACS: A message-passing parallel molecular dynamics implementation", *Computer Physics Communications*, 1995, **91**, 43–56.

(24) B. Hess, C. Kutzner, D. van der Spoel and E. Lindahl, "GROMACS 4: Algorithms for Highly Efficient, Load-Balanced, and Scalable Molecular Simulation", *Journal of Chemical Theory and Computation*, 2008, **4**, 435–447.

(25) E. Lindahl, B. Hess and D. van der Spoel, "GROMACS 3.0: a package for molecular simulation and trajectory analysis", *Molecular modeling annual*, 2001, **7**, 306–317.

(26) S. Páll, M. J. Abraham, C. Kutzner, B. Hess and E. Lindahl, in *Solving Software Challenges for Exascale: International Conference on Exascale Applications and Software, EASC 2014, Stockholm, Sweden, April 2-3, 2014, Revised Selected Papers*, ed. S. Markidis and E. Laure, Springer International Publishing, Cham, 2015, pp. 3–27.

(27) S. Pronk, S. Páll, R. Schulz, P. Larsson, P. Bjelkmar, R. Apostolov, M. R. Shirts, J. C. Smith, P. M. Kasson, D. van der Spoel, B. Hess and E. Lindahl, "GROMACS 4.5: a high-throughput and highly parallel open source molecular simulation toolkit", *Bioinformatics*, 2013, **29**, 845–854.

(28) D. Van Der Spoel, E. Lindahl, B. Hess, G. Groenhof, A. E. Mark and H. J. C. Berendsen, "GROMACS: Fast, flexible, and free", *Journal of Computational Chemistry*, 2005, **26**, 1701–1718.

(29) B. Hess, H. Bekker, H. J. C. Berendsen and J. G. E. M. Fraaije, "LINCS: A linear constraint solver for molecular simulations", *Journal of Computational Chemistry*, 1997, **18**, 1463–1472.

(30) B. Hess, "P-LINCS: A Parallel Linear Constraint Solver for Molecular Simulation", *Journal of Chemical Theory and Computation*, 2008, **4**, 116–122.

(31) J. Barker and R. Watts, "Monte Carlo studies of the dielectric properties of water-like models", *Molecular Physics*, 1973, **26**, 789–792.

(32) R. Watts, "Monte Carlo studies of liquid water", *Molecular Physics*, 1974, **28**, 1069–1083.

(33) G. Bussi, D. Donadio and M. Parrinello, "Canonical sampling through velocity rescaling", *The Journal of Chemical Physics*, 2007, **126**, 014101:1–7.

(34) H. J. C. Berendsen, J. P. M. Postma, W. F. van Gunsteren, A. DiNola and J. R. Haak, "Molecular dynamics with coupling to an external bath", *The Journal of Chemical Physics*, 1984, **81**, 3684–3690.

(35) L. Vermeer, B. L de Groot, V. Reat, A. Milon and J. Czaplicki, "Acyl Chain Order Parameter Profiles in Phospholipid Bilayers: Computation from Molecular Dynamics Simulations and Comparison with H-2 NMR Experiments", 2007, **36**, 919–31.

13. Summary and Future Endeavours

13.1. SpinningTop

In part I, a method for fitting a Fourier series to noisy, multidimensional data was developed. The method utilises robust regression to account for noise in the data, and phasors to linearise the non-linear cosine function. Derivation of the mathematical underpinnings of the method were outlined in detail and applied to the specific case of fitting dihedral terms within a molecular dynamics force field. Though the force field chosen was the GROMOS 54A7 force field, the generality of the method means that it can be applied to any force field.

A test of the method was performed with the goal of improving the amino acid backbone dihedral parameter values within the GROMOS 54A7 force field. To this end, a subset of the twenty naturally occurring amino acids was selected for improvement. The selection process involved a pairwise comparison of Ramachandran plots generated from data available in the RCSB PDB. This process identified which amino acids were similar to one another, with only the most similar of each group undergoing the improvement process.

For each amino acid selected, a reference quantum chemical potential energy surface of the dipeptide was generated. The difference between this energy surface and a corresponding free energy surface generated through local elevation enhanced molecular dynamics simulations was used as the data to fit to. In this way, when the parameters determined through the fitting process were applied to the free energy surface, it would match the quantum chemical potential energy surface. These results were then compared with corresponding pseudo free energy surfaces generated from data available in the RCSB PDB. Smaller amino acids, such as glycine and alanine, showed much improved free energy surfaces in comparison to the existing GROMOS 54A7 force field. With the larger amino acids, improvements were less noticeable, as sampling of the degrees of freedom in the side chains was diminished, resulting in poorer convergence of the energy surfaces.

There are a number of areas related to this part which could be investigated in the future. The simplest is to improve the energy surfaces through improved sampling. For the molecular dynamics free energy surface, this can be obtained through extending the time length of the simulations, whereas the quantum chemical potential energy surface would require further sampling

of conformational space. Additionally, comparing surfaces formed by dipeptides with those obtained from x-ray crystal structures is problematic as the backbone dihedral angles are restrained within a protein due to the secondary structure and so will have different energetics to a dipeptide which has no secondary structure. This could be improved by attaining high resolution, solution phase NMR studies of the dipeptides in order to compare against.

13.2. CherryPicker

Part II details the CherryPicker algorithm which is designed to automatically parameterise large biomolecules. The algorithm uses subgraph isomorphism matching to identify fragments of small molecules which match portions of a larger molecule. As these small molecules are pre-parameterised, the parameters associated with all the matching fragments are collated and used to determine parameters for the matching portions of the larger molecule. The algorithm was tested in a proof of concept manner by parameterising a collection of medium sized organic molecules and comparing ^{1}H NMR chemical shifts calculated from simulated ensembles of structures to experimentally obtained spectra. These results indicated that NMR shifts were mostly able to be reproduced.

As part of the CherryPicker algorithm, a number of optimisation techniques were developed to automatically and accurately determine bond order and formal charge values. The best of these optimisation techniques, an FPT based method, was found to perform equal to or better than current state of the art methods which are only capable of determining bond orders.

Two primary aims of future work on this topic exist. The first is to improve performance. The algorithms were implemented in Python, which is a dynamically typed language and thus has somewhat poor performance. Preliminary work indicates that implementing in a lower level, statically typed language, such as C++, should result in at least a fifty-fold increase in performance. The second aim is to test parameters obtained throughly. Here, a minor test was undertaken using NMR data. However, the reference data was low resolution and the ability of NMR to distinguish well parameterised from poorly parameterised systems is currently unknown. As such, alternative testing methods should be investigated. These could include parameterisation of complex lipids and comparison of bilayer simulations with simulations performed using well tested manual parameters, as well as binding energies of large ligands in protein systems. Additionally, determining the applicability of NMR to testing parameterisation would be another option.

13.3. Monolayers

Finally, part III details a small investigation into the effect of thermodynamic ensemble on the properties of a napthalimide monolayer. Parameters for the napthalimide molecule were produced by manually applying the methods involved in the CherryPicker algorithm. A monolayer was formed and simulated under the NVT, NPT and NγT ensembles. Results showed that within the same isothermal range, the three ensembles performed similarly. This is an expected outcome, but it is good to confirm that it is the case. As this work was only a minor portion of this thesis, it has vast opportunity to be expanded upon. Functionalised napthalimides could be tested, conditions between the three ensembles could be selected to more clearly cover the same isothermal range, and simulations could be run for extended time scales to ensure convergence of relevant properties.

Part IV.

Appendices

A. Dihedral Energy Profiles

Dihedral energy profiles for each mono substituted 3-ethyl hexane compound described in section 3.3.

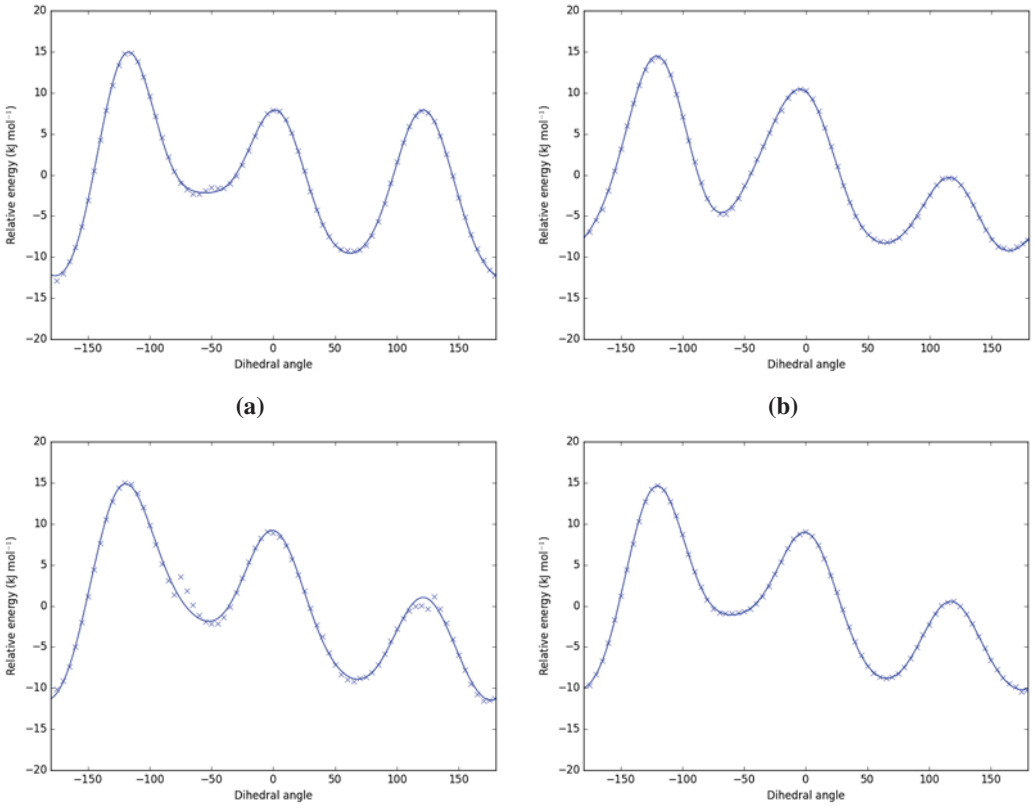


Figure A.1: Mono methyl substituted 3-ethyl hexane dihedral energy profiles, with methyl at **(a)** R_0 (RMSD = 0.179 kJ mol^{-1}), **(b)** R_1 (RMSD = 0.143 kJ mol^{-1}), **(c)** R_2 (RMSD = 0.561 kJ mol^{-1}) and **(d)** R_3 (RMSD = 0.105 kJ mol^{-1}).

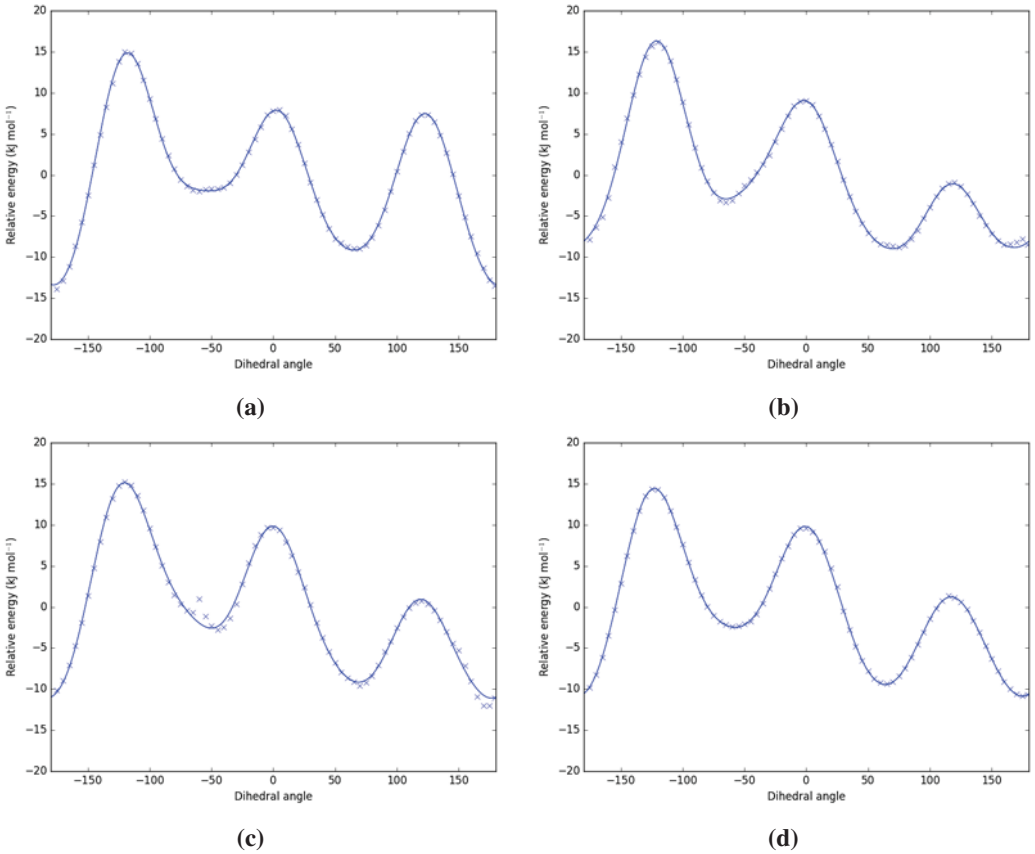


Figure A.2: Mono amine substituted 3-ethyl hexane dihedral energy profiles, with amine at **(a)** R_0 (RMSD = 0.177 kJ mol^{-1}), **(b)** R_1 (RMSD = 0.288 kJ mol^{-1}), **(c)** R_2 (RMSD = 0.531 kJ mol^{-1}) and **(d)** R_3 (RMSD = 0.114 kJ mol^{-1}).

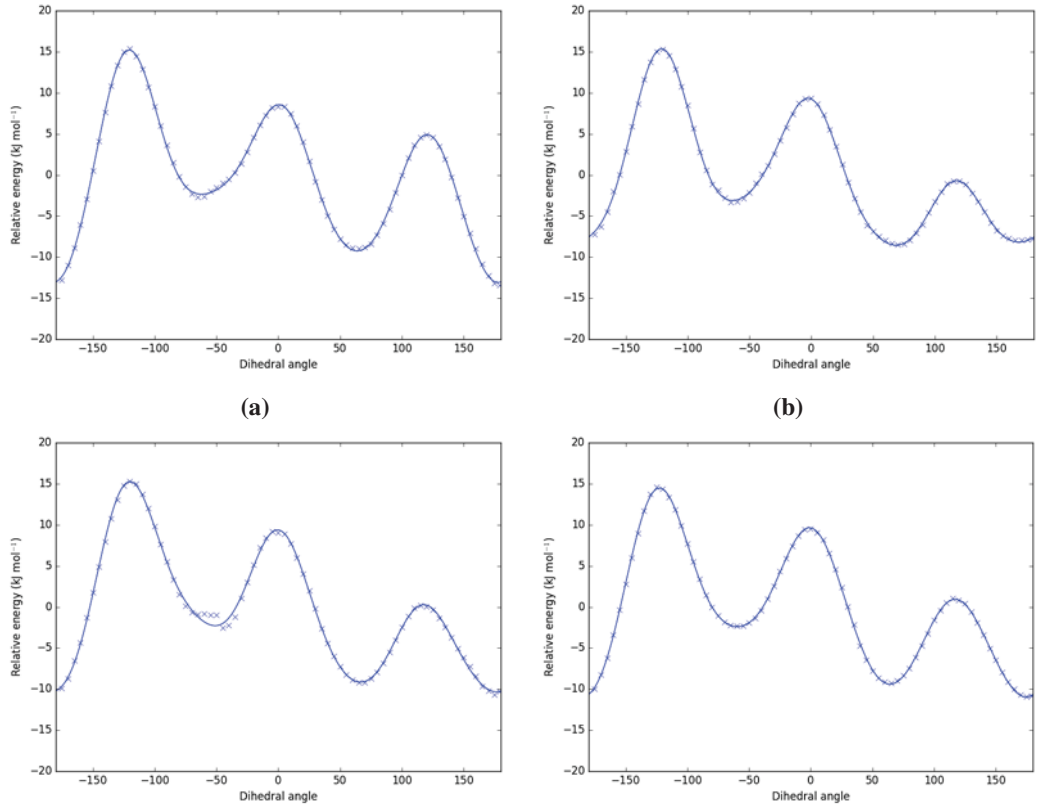


Figure A.3: Mono hydroxy substituted 3-ethyl hexane dihedral energy profiles, with hydroxide at **(a)** R_0 (RMSD = 0.190 kJ mol^{-1}), **(b)** R_1 (RMSD = 0.169 kJ mol^{-1}), **(c)** R_2 (RMSD = 0.358 kJ mol^{-1}) and **(d)** R_3 (RMSD = 0.116 kJ mol^{-1}).

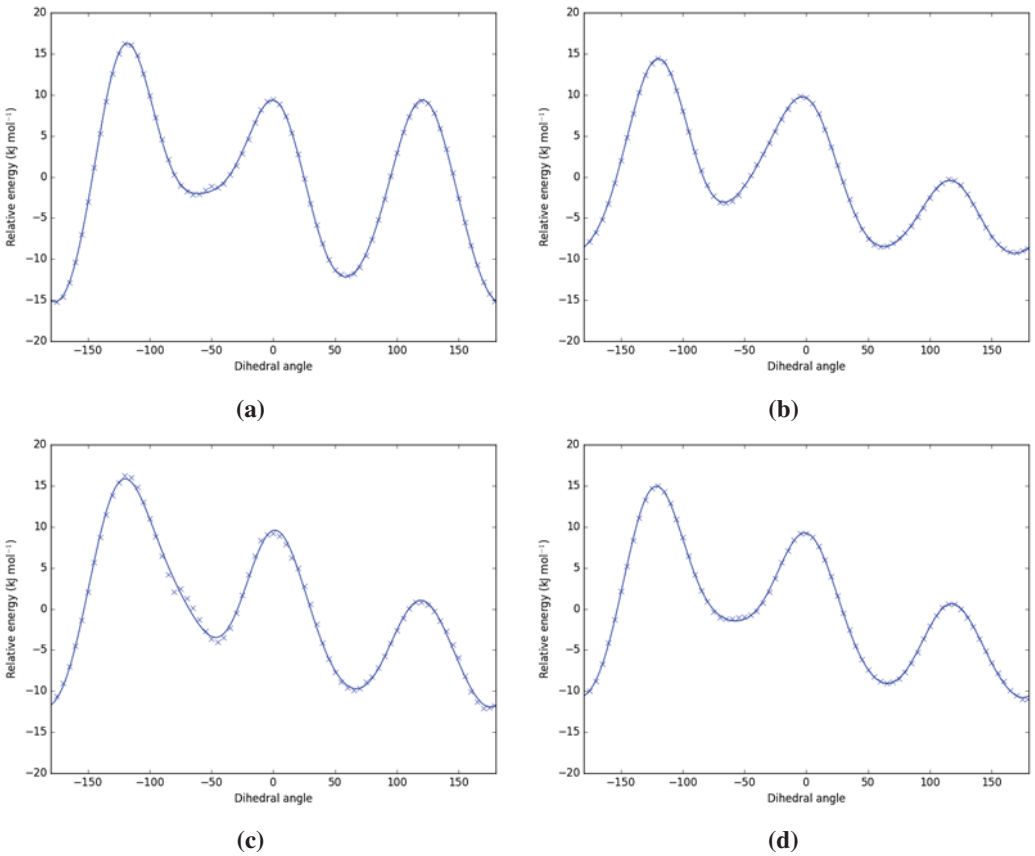


Figure A.4: Mono thio substituted 3-ethyl hexane dihedral energy profiles, with thiol at **(a)** R_0 (RMSD = 0.138 kJ mol^{-1}), **(b)** R_1 (RMSD = 0.124 kJ mol^{-1}), **(c)** R_2 (RMSD = 0.402 kJ mol^{-1}) and **(d)** R_3 (RMSD = 0.149 kJ mol^{-1}).

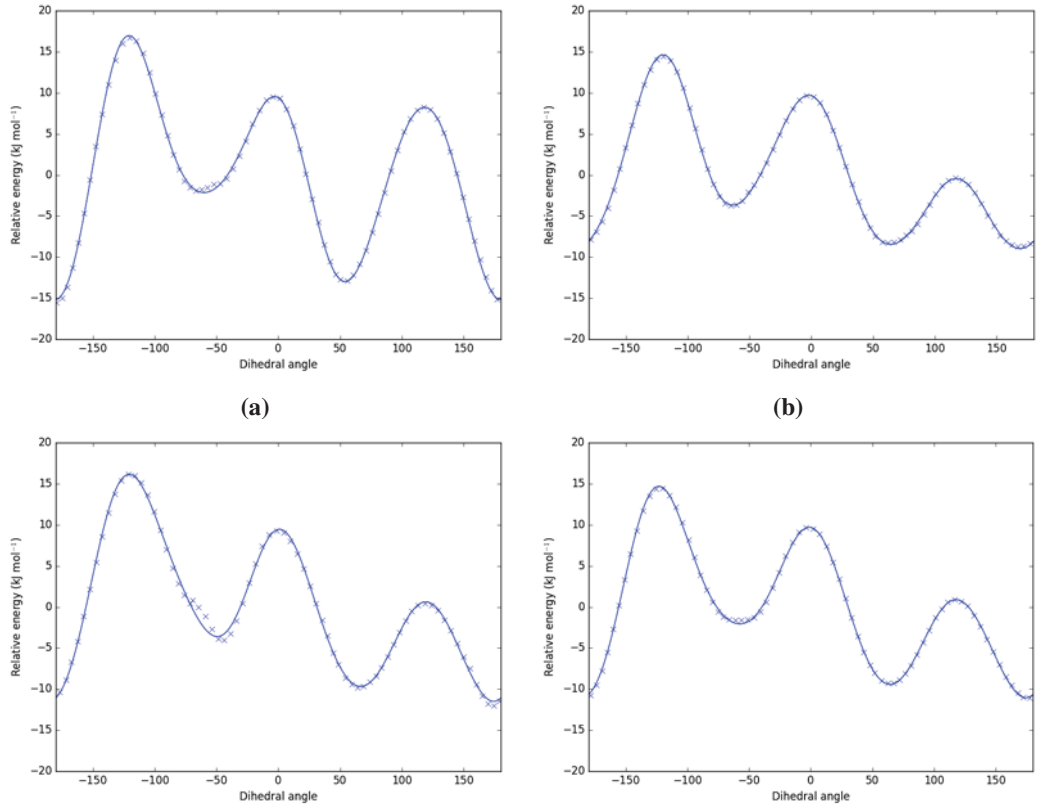


Figure A.5: Mono chloro substituted 3-ethyl hexane dihedral energy profiles, with chlorine at **(a)** R_0 (RMSD = 0.234 kJ mol^{-1}), **(b)** R_1 (RMSD = 0.145 kJ mol^{-1}), **(c)** R_2 (RMSD = 0.435 kJ mol^{-1}) and **(d)** R_3 (RMSD = 0.172 kJ mol^{-1}).

B. Sampling Density RMSD Values with Different Sample Sets

Effect of fitting to different subsets of the data obtained in section 3.4, at regular intervals.

Table B.1: Results of sampling density changes with RMSD for substituted 3-ethyl hexane. RMSD values are in kJ mol^{-1}.

	5°	10°	15°	20°	30°
Hydrogen	0.166	0.166	0.168	0.174	0.313
		0.166	0.168	0.172	0.393
			0.168	0.172	0.514
				0.173	0.530
					0.448
					0.289

Continued on next page

Table B.1 – *Continued from previous page*

		5°	10°	15°	20°	30°
Carbon	R_0	0.179	0.179	0.183	0.188	0.248
			0.179	0.182	0.194	0.411
				0.179	0.193	0.682
					0.191	0.827
						0.779
						0.552
	R_1	0.143	0.145	0.150	0.149	0.212
			0.145	0.145	0.145	0.281
				0.152	0.153	0.459
					0.147	0.543
						0.512
						0.363
	R_2	0.561	0.612	0.584	0.620	0.708
			0.541	0.564	0.752	0.830
				0.651	0.587	0.939
					0.572	0.964
						0.767
						0.598
	R_3	0.105	0.109	0.111	0.115	0.256
			0.108	0.113	0.111	0.357
				0.116	0.118	0.503
					0.121	0.533
						0.427
						0.239

Continued on next page

Table B.1 – *Continued from previous page*

		5°	10°	15°	20°	30°
Nitrogen	R_0	0.177	0.179	0.182	0.188	0.245
			0.179	0.182	0.188	0.368
				0.181	0.197	0.582
					0.188	0.720
						0.695
						0.494
	R_1	0.288	0.291	0.296	0.311	0.439
			0.290	0.304	0.304	0.511
				0.334	0.308	0.630
					0.304	0.687
						0.629
						0.516
	R_2	0.531	0.545	0.586	0.544	0.966
			0.521	0.540	0.546	0.779
				0.525	0.545	0.879
					0.531	0.885
						0.769
						0.611
	R_3	0.114	0.119	0.119	0.119	0.339
			0.117	0.117	0.131	0.356
				0.120	0.141	0.413
					0.126	0.373
						0.264
						0.170

Continued on next page

Table B.1 – *Continued from previous page*

		5°	10°	15°	20°	30°
Oxygen	R_0	0.190	0.192	0.192	0.202	0.303
			0.192	0.194	0.199	0.353
				0.193	0.203	0.492
					0.196	0.551
						0.497
						0.382
	R_1	0.169	0.172	0.178	0.213	0.289
			0.171	0.177	0.199	0.440
				0.180	0.199	0.612
					0.184	0.644
						0.522
						0.297
	R_2	0.358	0.366	0.396	0.360	0.551
			0.374	0.372	0.371	0.645
				0.384	0.390	0.757
					0.378	0.722
						0.591
						0.494
	R_3	0.116	0.117	0.121	0.128	0.321
			0.118	0.126	0.123	0.333
				0.128	0.122	0.376
					0.130	0.337
						0.244
						0.157

Continued on next page

Table B.1 – *Continued from previous page*

		5°	10°	15°	20°	30°
Sulfur	R_0	0.138	0.139	0.145	0.156	0.238
			0.139	0.141	0.148	0.343
				0.145	0.157	0.599
					0.152	0.690
						0.629
						0.425
	R_1	0.124	0.125	0.126	0.131	0.176
			0.125	0.125	0.131	0.247
				0.127	0.131	0.386
					0.126	0.473
						0.448
						0.320
	R_2	0.402	0.411	0.424	0.484	0.634
			0.414	0.404	0.427	0.663
				0.471	0.426	0.781
					0.411	0.733
						0.624
						0.516
	R_3	0.149	0.151	0.154	0.156	0.359
			0.150	0.154	0.158	0.414
				0.157	0.153	0.516
					0.153	0.494
						0.388
						0.220

Continued on next page

Table B.1 – *Continued from previous page*

		5°	10°	15°	20°	30°
Chlorine	R_0	0.234	0.236	0.237	0.245	0.819
			0.236	0.236	0.242	0.400
				0.236	0.247	0.616
					0.244	1.135
						1.490
						7.614
	R_1	0.145	0.146	0.147	0.154	7.221
			0.146	0.146	0.146	0.385
				0.148	0.151	0.244
					0.149	0.603
						0.884
						0.927
	R_2	0.435	0.434	0.464	0.433	1.069
			0.438	0.427	0.447	0.789
				0.466	0.489	0.613
					0.451	1.973
						1.202
						1.337
	R_3	0.172	0.173	0.177	0.175	0.644
			0.172	0.175	0.173	0.370
				0.175	0.177	0.362
					0.175	0.726
						0.997
						0.981

C. PDB Ramachandran Plots

Ramachandran plots produced from PDB source data. In all cases, the left hand figure is the normalised plot used for point wise comparisons as per section 4.1, and the right hand figure is a log plot of the counts of each Φ/Ψ pair.

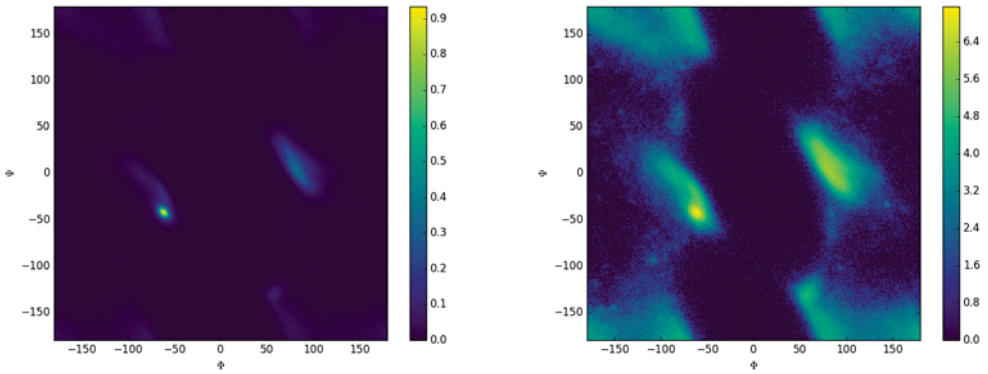


Figure C.1: Glycine Ramachandran plots.

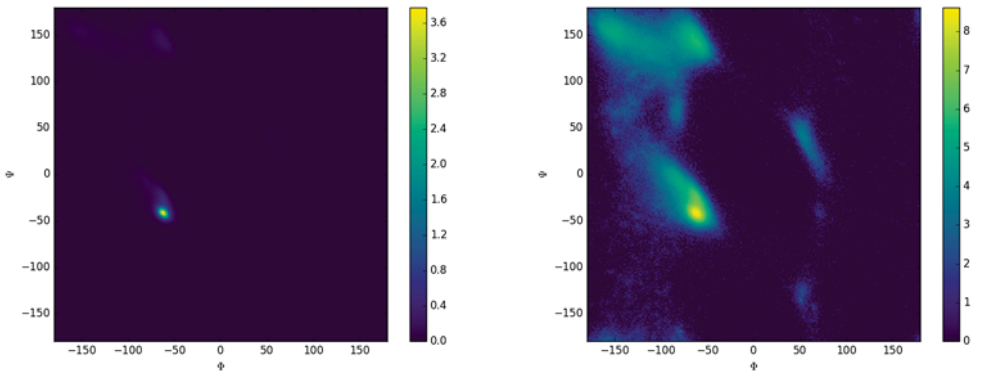


Figure C.2: Alanine Ramachandran plots.

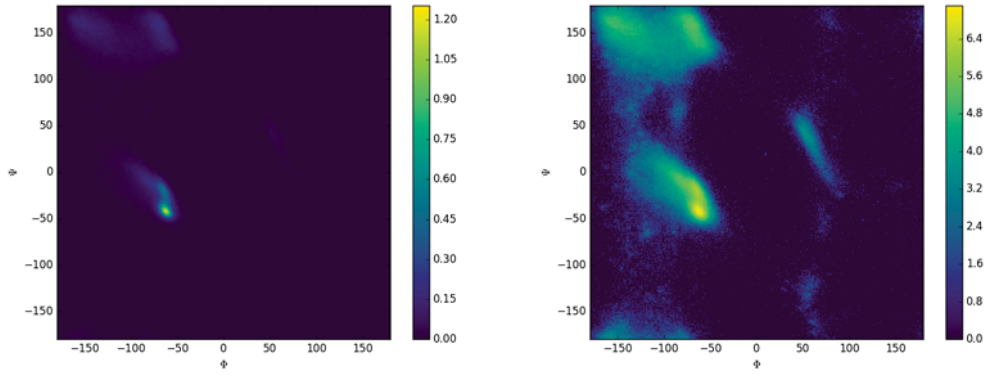


Figure C.3: Serine Samachandran plots.

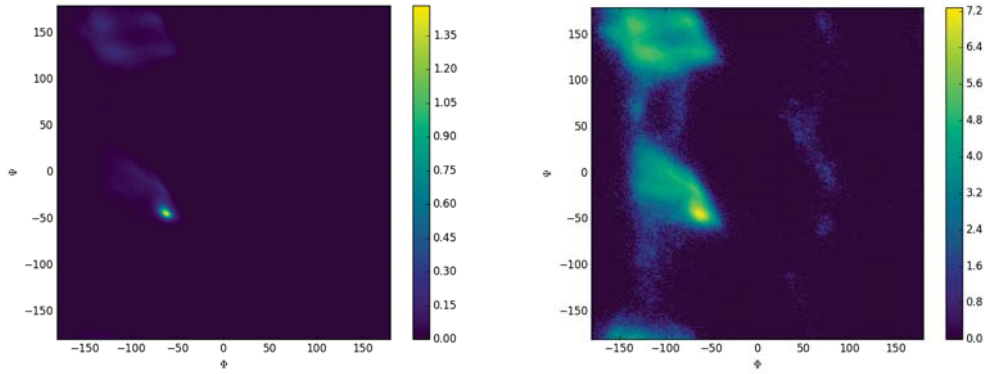


Figure C.4: Threonine Ramachandran plots.

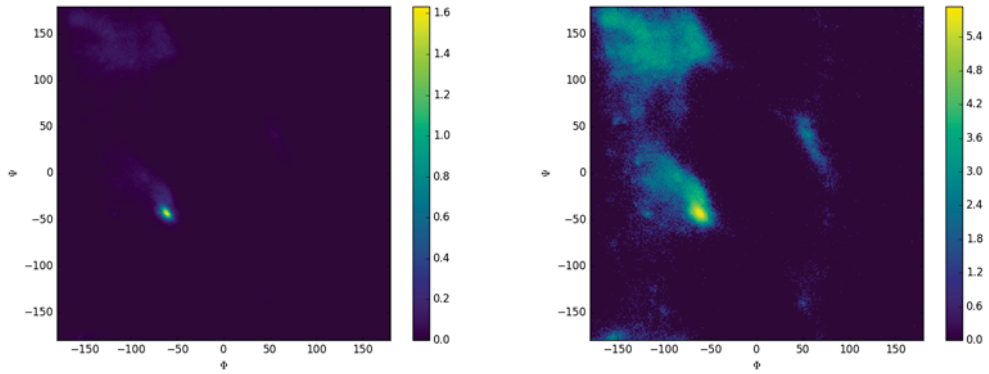


Figure C.5: Cysteine Ramachandran plots.

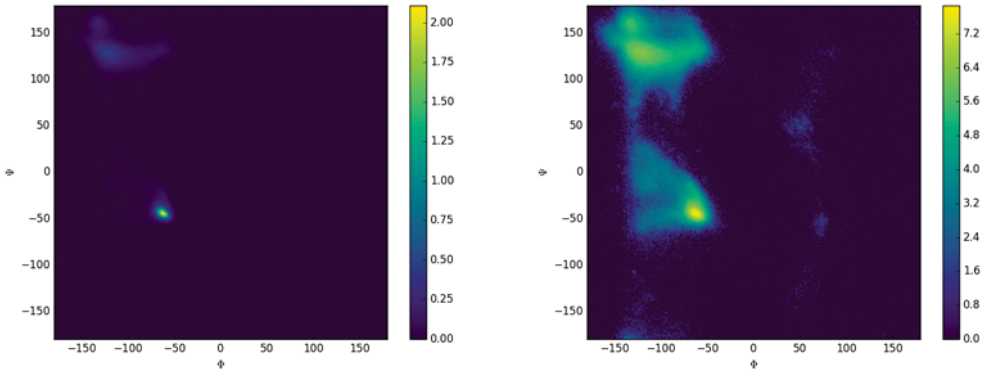


Figure C.6: Valine Ramachandran plots.

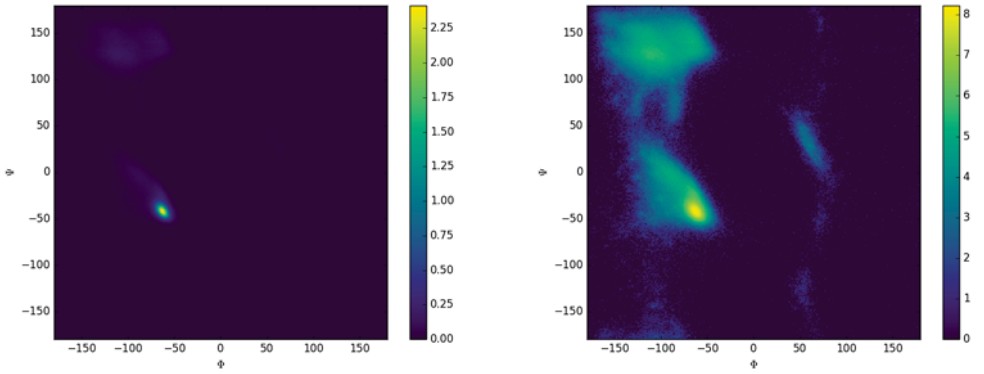


Figure C.7: Leucine Ramachandran plots.

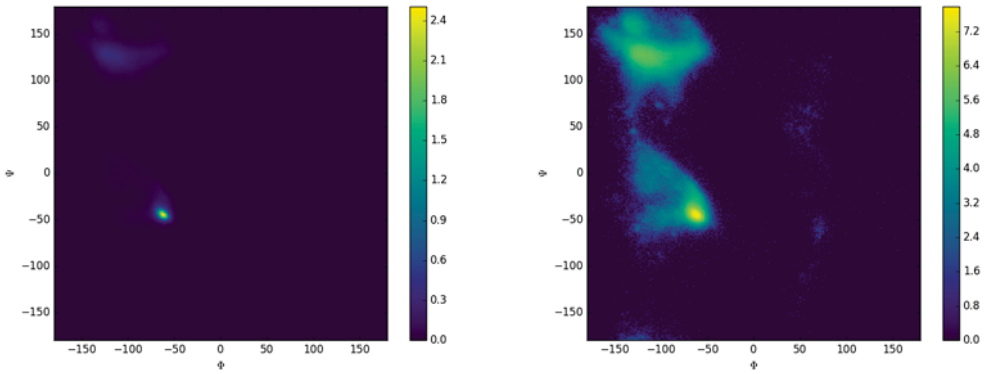


Figure C.8: Isoleucine Ramachandran plots.

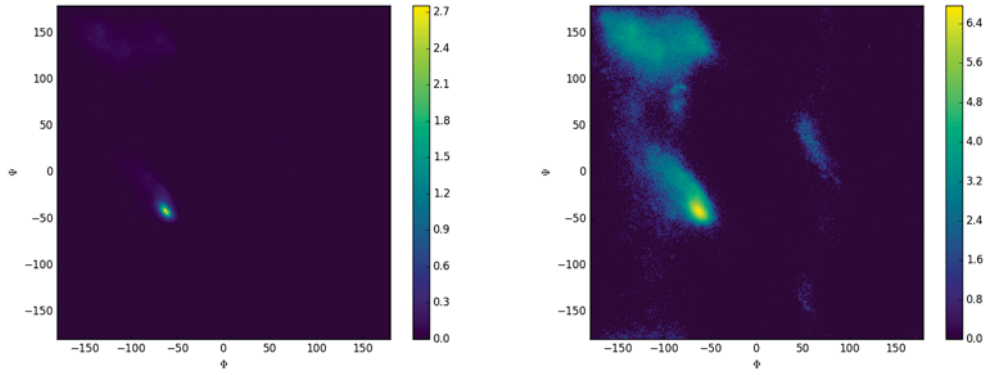


Figure C.9: Methionine Ramachandran plots.

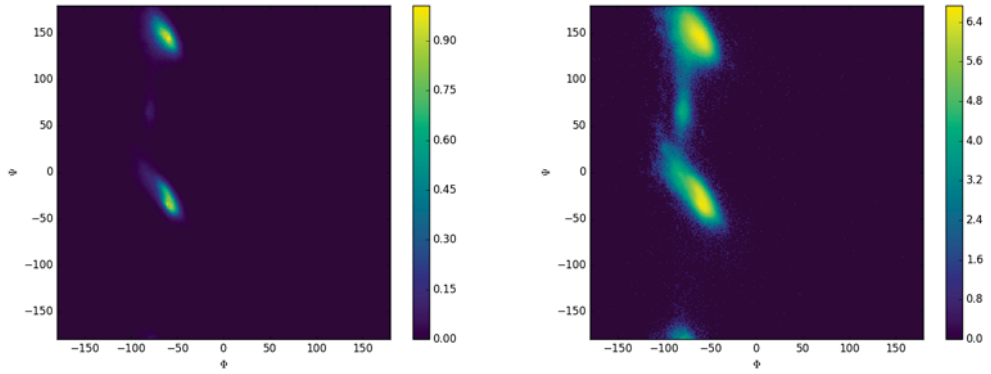


Figure C.10: Proline Ramachandran plots.

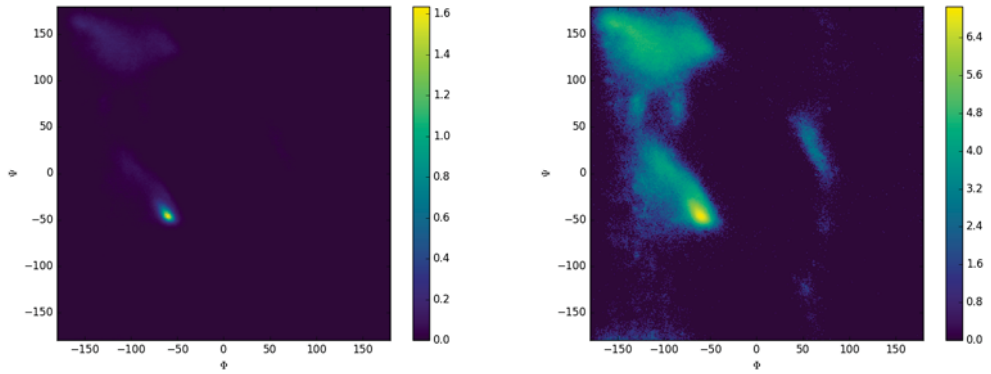


Figure C.11: Phenylalanine Ramachandran plots.

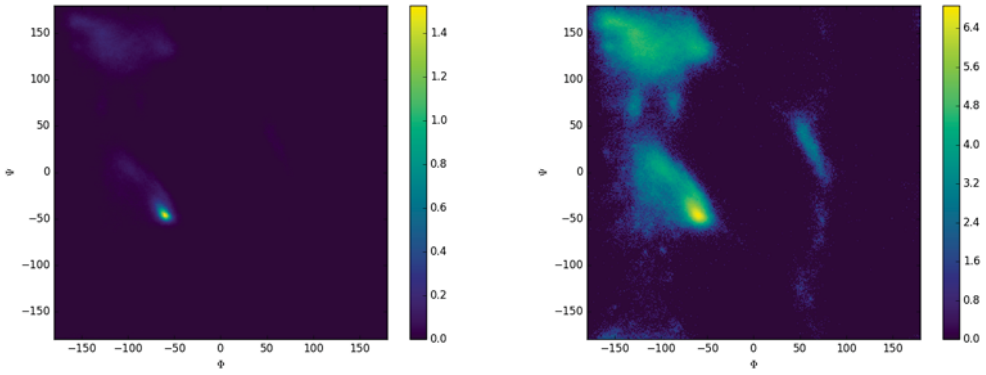


Figure C.12: Tyrosine Ramachandran plots.

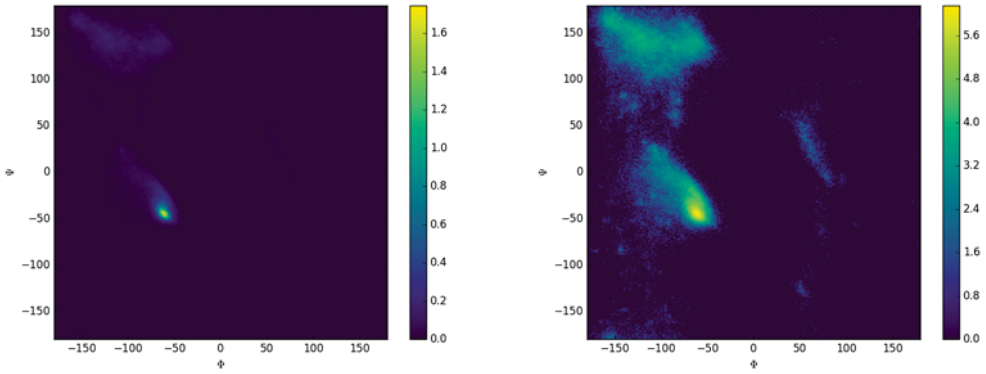


Figure C.13: Tryptophan Ramachandran plots.

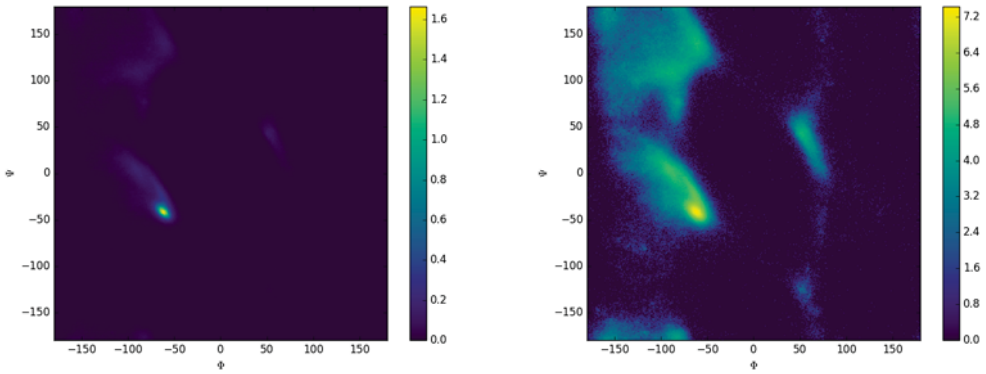


Figure C.14: Aspartic Acid Ramachandran plots.

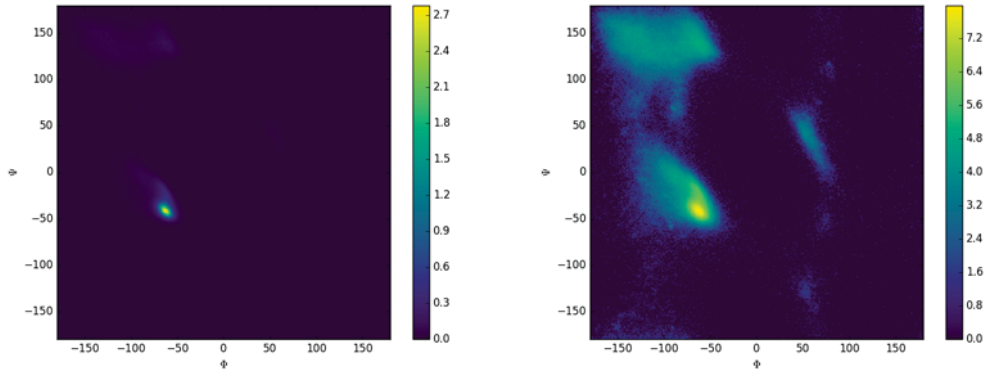


Figure C.15: Glutamic Acid Ramachandran plots.

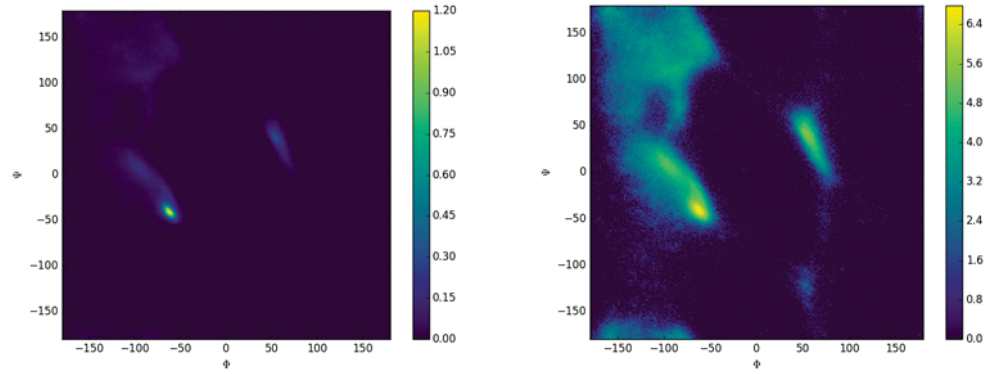


Figure C.16: Asparagine Ramachandran plots.

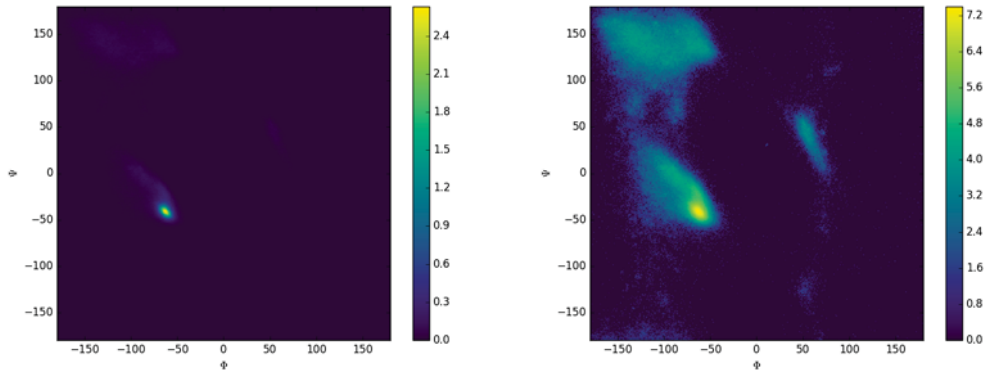


Figure C.17: Glutamine Ramachandran plots.

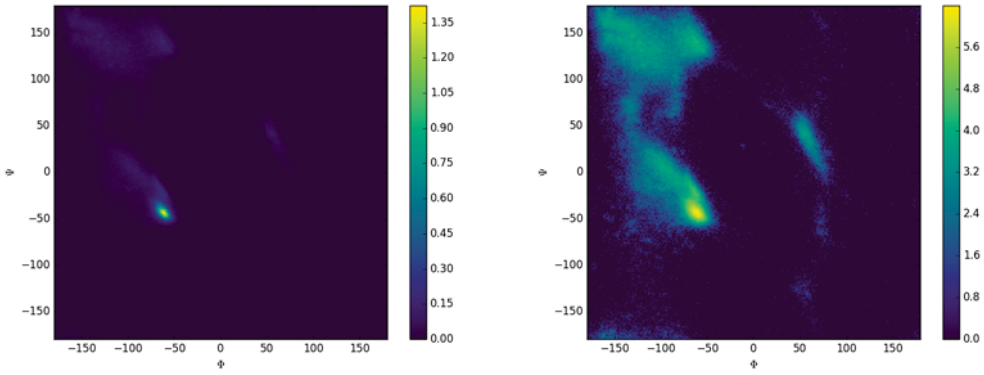


Figure C.18: Histidine Ramachandran plots.

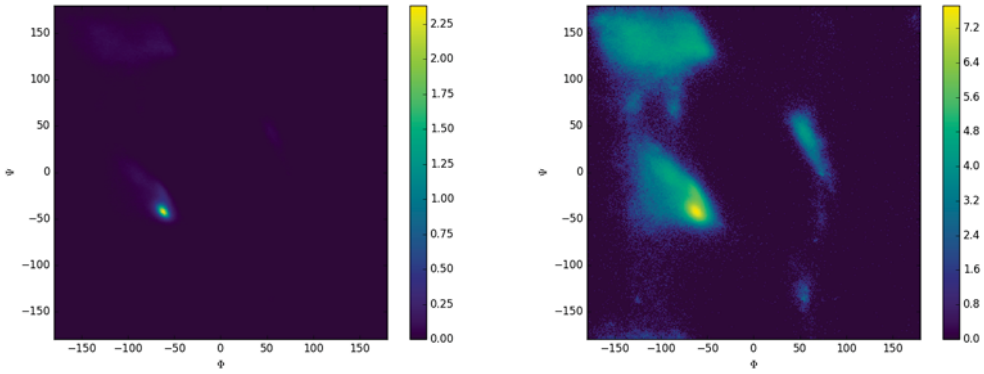


Figure C.19: Lysine Ramachandran plots.

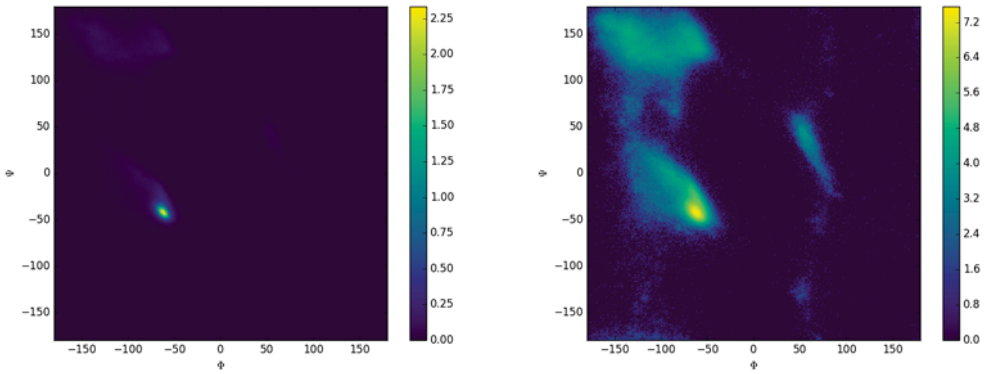


Figure C.20: Arginine Ramachandran plots.

D. Raw Energy Surfaces

Raw QM and free energy surfaces obtained as part of chapter 5. In all cases, diamonds mark the positions of the calculated data and contours are placed every 2 kJ mol^{-1}.

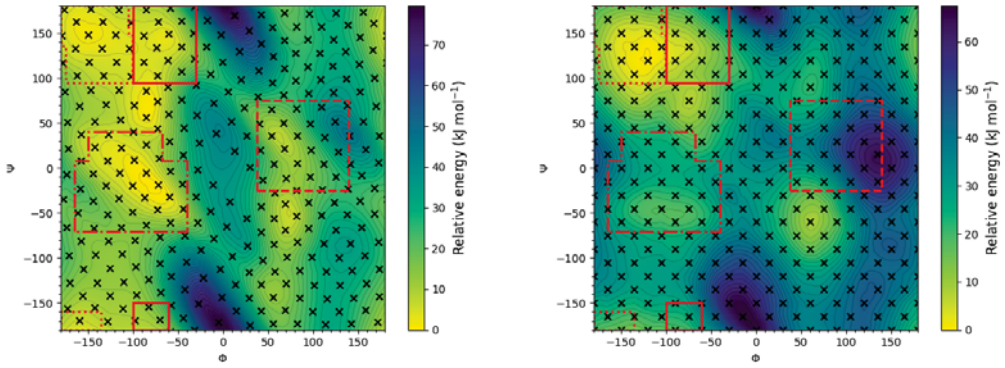


Figure D.1: Raw alanine dipeptide energy surfaces. The left hand figure shows the raw QM potential energy surface and the right hand figure shows the raw local elevation-molecular dynamics free energy surface.

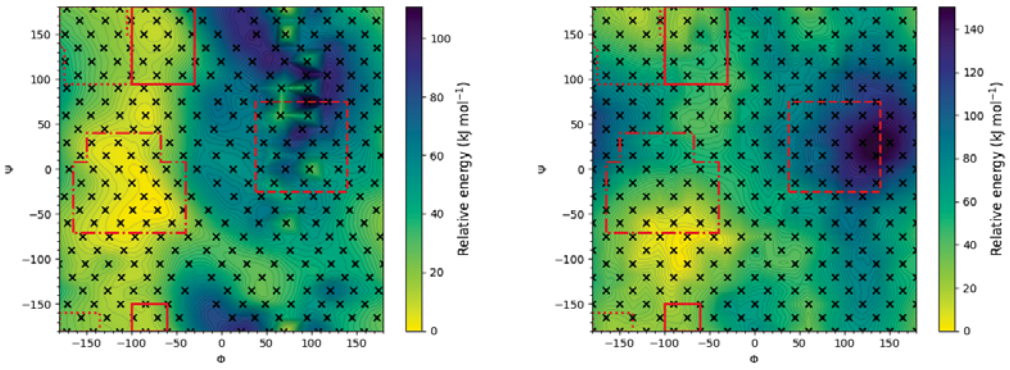


Figure D.2: Raw aspartic acid dipeptide energy surfaces. The left hand figure shows the raw QM potential energy surface and the right hand figure shows the raw local elevation-molecular dynamics free energy surface.

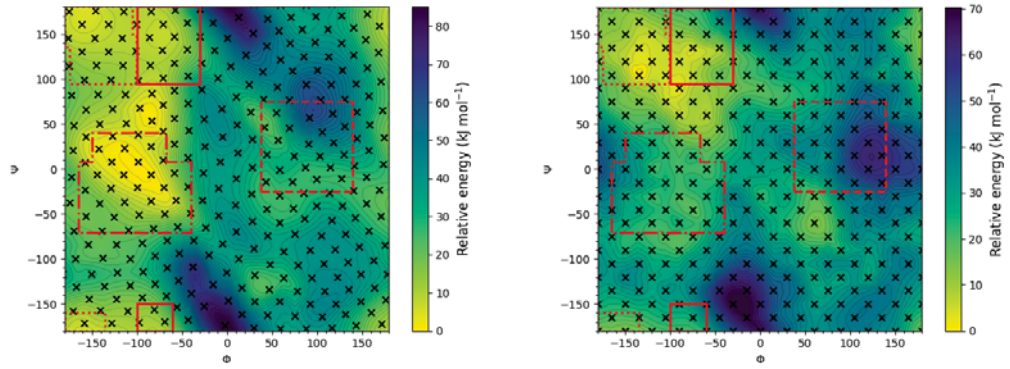


Figure D.3: Raw cysteine dipeptide energy surfaces. The left hand figure shows the raw QM potential energy surface and the right hand figure shows the raw local elevation-molecular dynamics free energy surface.

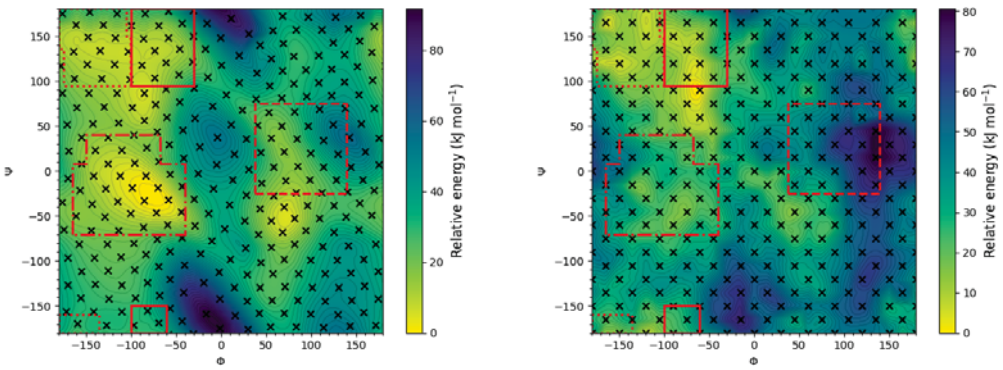


Figure D.4: Raw glutamine dipeptide energy surfaces. The left hand figure shows the raw QM potential energy surface and the right hand figure shows the raw local elevation-molecular dynamics free energy surface.

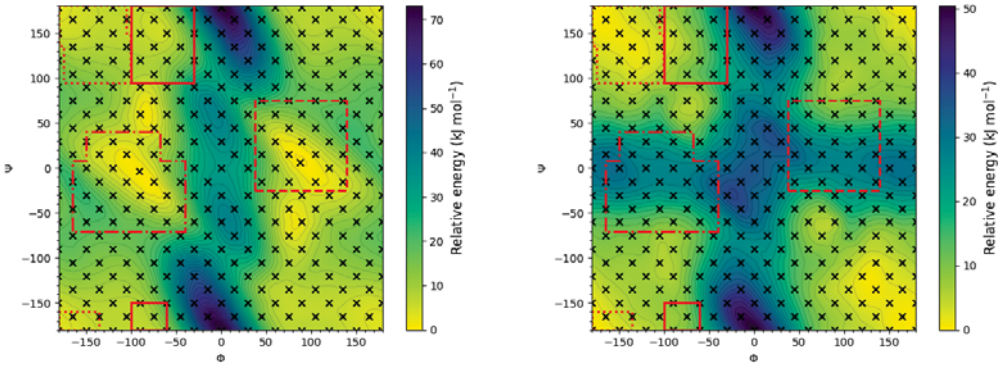


Figure D.5: Raw glycine dipeptide energy surfaces. The left hand figure shows the raw QM potential energy surface and the right hand figure shows the raw local elevation-molecular dynamics free energy surface.

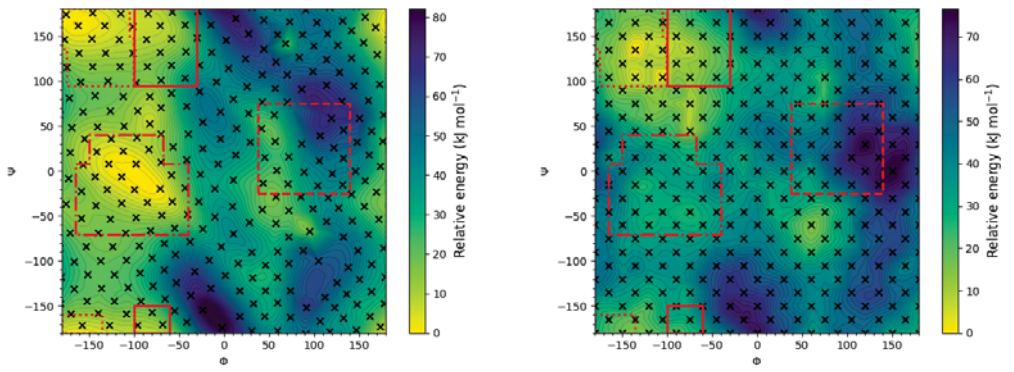


Figure D.6: Raw phenylalanine dipeptide energy surfaces. The left hand figure shows the raw QM potential energy surface and the right hand figure shows the raw local elevation-molecular dynamics free energy surface.

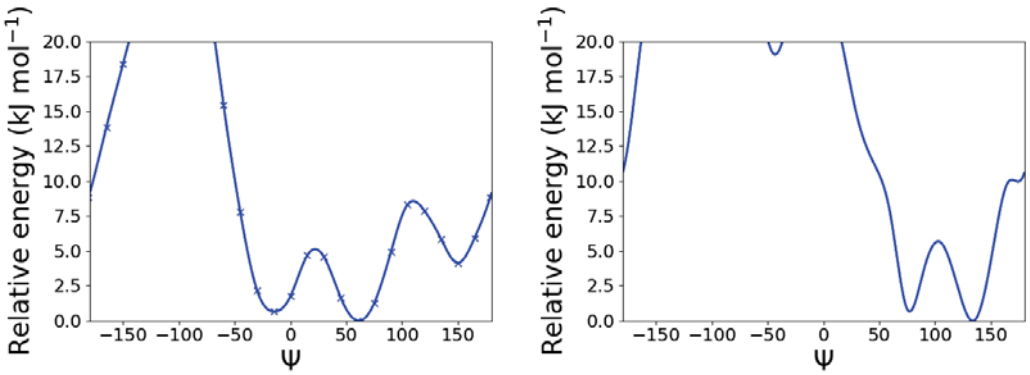


Figure D.7: Raw proline dipeptide energy surfaces. The left hand figure shows the raw QM potential energy surface and the right hand figure shows the raw local elevation-molecular dynamics free energy surface.

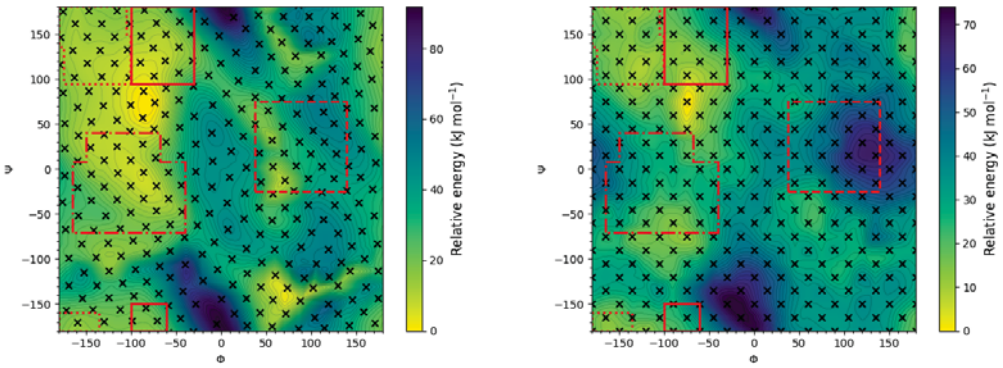


Figure D.8: Raw serine dipeptide energy surfaces. The left hand figure shows the raw QM potential energy surface and the right hand figure shows the raw local elevation-molecular dynamics free energy surface.

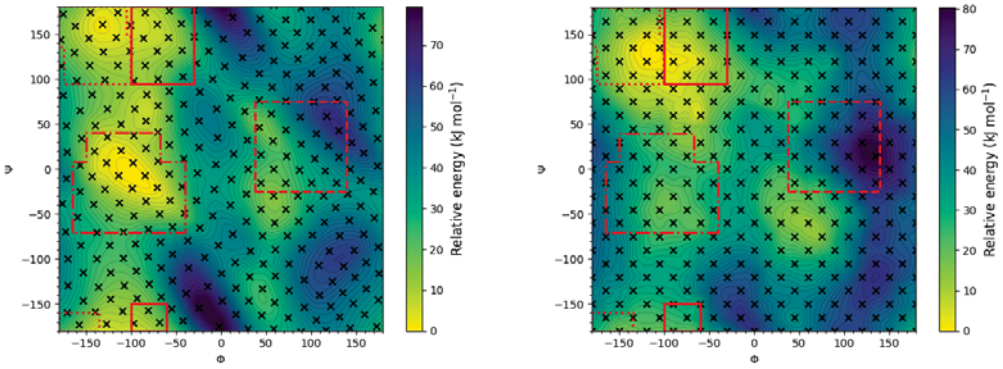


Figure D.9: Raw valine dipeptide energy surfaces. The left hand figure shows the raw QM potential energy surface and the right hand figure shows the raw local elevation-molecular dynamics free energy surface.

E. Naphthalimide Parameters

Parameters used for simulation of naphthalimide monolayers.

```
[ atoms ]
; nr type resnr  resid  atom  cgnr   charge   mass         total_charge
1    CH3    1    ISOQ    C30    1    0.00000  15.0350     ;  0.00000
2    CH2    1    ISOQ    C29    2    0.00000  14.0270     ;  0.00000
3    CH2    1    ISOQ    C28    3    0.00000  14.0270     ;  0.00000
4    CH2    1    ISOQ    C27    4    0.00000  14.0270     ;  0.00000
5    CH2    1    ISOQ    C26    5    0.00000  14.0270     ;  0.00000
6    CH2    1    ISOQ    C25    6    0.00000  14.0270     ;  0.00000
7    CH2    1    ISOQ    C24    7    0.00000  14.0270     ;  0.00000
8    CH2    1    ISOQ    C23    8    0.00000  14.0270     ;  0.00000
9    CH2    1    ISOQ    C22    9    0.00000  14.0270     ;  0.00000
10   CH2    1    ISOQ    C21   10    0.00000  14.0270     ;  0.00000
11   CH2    1    ISOQ    C20   11    0.00000  14.0270     ;  0.00000
12   CH2    1    ISOQ    C19   12    0.00000  14.0270     ;  0.00000
13   CH2    1    ISOQ    C18   13    0.00000  14.0270     ;  0.00000
14   CH2    1    ISOQ    C17   14    0.00000  14.0270     ;  0.00000
15   CH2    1    ISOQ    C16   15    0.00000  14.0270     ;  0.00000
16   CH2    1    ISOQ    C15   16    0.00000  14.0270     ;  0.00000
17   CH2    1    ISOQ    C14   17    0.00000  14.0270     ;  0.00000
18   CH2    1    ISOQ    C13   18    0.35300  14.0270
19    NT    1    ISOQ     N1   18   -0.35300  14.0067     ;  0.00000
20     C    1    ISOQ    C11   19    0.70000  12.0110
21     O    1    ISOQ     O1   19   -0.45000  15.9994
22     C    1    ISOQ     C2   19   -0.25000  12.0110     ; -0.00000
23     C    1    ISOQ     C1   20   -0.14000  12.0110
24    HC    1    ISOQ     H1   20    0.14000   1.0080     ;  0.00000
25     C    1    ISOQ     C3   21    0.00000  12.0110     ;  0.00000
26     C    1    ISOQ     C4   22    0.00000  12.0110     ;  0.00000
```

```
27     C    1    ISOQ     C5   23   -0.14000  12.0110
28    HC    1    ISOQ     H2   23    0.14000   1.0080     ;  0.00000
29     C    1    ISOQ     C6   24   -0.14000  12.0110
30    HC    1    ISOQ     H3   24    0.14000   1.0080     ;  0.00000
31     C    1    ISOQ     C8   25   -0.14000  12.0110
32    HC    1    ISOQ     H4   25    0.14000   1.0080     ;  0.00000
33     C    1    ISOQ     C9   26   -0.14000  12.0110
34    HC    1    ISOQ     H5   26    0.14000   1.0080     ;  0.00000
35     C    1    ISOQ    C10   27   -0.14000  12.0110
36    HC    1    ISOQ     H6   27    0.14000   1.0080     ;  0.00000
37     C    1    ISOQ     C7   28   -0.25000  12.0110
38     C    1    ISOQ    C12   28    0.70000  12.0110
39     O    1    ISOQ     O2   28   -0.45000  15.9994     ; -0.00000
; total charge of the molecule:   0.00000
[ bonds ]
;  ai   aj  funct   c0         c1
1     2    2   0.1530     7.1500e+06
2     3    2   0.1530     7.1500e+06
3     4    2   0.1530     7.1500e+06
4     5    2   0.1530     7.1500e+06
5     6    2   0.1530     7.1500e+06
6     7    2   0.1530     7.1500e+06
7     8    2   0.1530     7.1500e+06
8     9    2   0.1530     7.1500e+06
9    10    2   0.1530     7.1500e+06
10   11    2   0.1530     7.1500e+06
11   12    2   0.1530     7.1500e+06
12   13    2   0.1530     7.1500e+06
13   14    2   0.1530     7.1500e+06
14   15    2   0.1530     7.1500e+06
15   16    2   0.1530     7.1500e+06
16   17    2   0.1530     7.1500e+06
17   18    2   0.1530     7.1500e+06
18   19    2   0.1480     7.6400e+06
19   20    2   0.1400     8.5400e+06
19   38    2   0.1400     8.5400e+06
```

```
20   21    2   0.1230     1.6600e+07
20   22    2   0.1480     5.7300e+06
22   23    2   0.1390     1.0800e+07
22   25    2   0.1390     1.0800e+07
23   24    2   0.1090     1.2300e+07
23   29    2   0.1390     1.0800e+07
25   26    2   0.1390     1.0800e+07
25   37    2   0.1390     1.0800e+07
26   27    2   0.1390     1.0800e+07
26   31    2   0.1390     1.0800e+07
27   28    2   0.1090     1.2300e+07
27   29    2   0.1390     1.0800e+07
29   30    2   0.1090     1.2300e+07
31   32    2   0.1090     1.2300e+07
31   33    2   0.1390     1.0800e+07
33   34    2   0.1090     1.2300e+07
33   35    2   0.1390     1.0800e+07
35   36    2   0.1090     1.2300e+07
35   37    2   0.1390     1.0800e+07
37   38    2   0.1480     5.7300e+06
38   39    2   0.1230     1.6600e+07
[ pairs ]
1     4    1
2     5    1
3     6    1
4     7    1
5     8    1
6     9    1
7    10    1
8    11    1
9    12    1
10   13    1
11   14    1
12   15    1
13   16    1
14   17    1
```

```
15   18    1
16   19    1
17   20    1
17   38    1
18   21    1
18   22    1
18   37    1
18   39    1
20   24    1
20   39    1
21   23    1
21   25    1
21   38    1
22   30    1
23   28    1
24   25    1
24   27    1
24   30    1
25   28    1
25   32    1
25   36    1
25   39    1
26   30    1
26   34    1
27   32    1
28   30    1
28   31    1
31   36    1
32   34    1
32   35    1
34   36    1
34   37    1
35   39    1
36   38    1
[ angles ]
;  ai   aj   ak  funct   angle     fc
```

1	2	3	2	111.00	530.00
2	3	4	2	111.00	530.00
3	4	5	2	111.00	530.00
4	5	6	2	111.00	530.00
5	6	7	2	111.00	530.00
6	7	8	2	111.00	530.00
7	8	9	2	111.00	530.00
8	9	10	2	111.00	530.00
9	10	11	2	111.00	530.00
10	11	12	2	111.00	530.00
11	12	13	2	111.00	530.00
12	13	14	2	111.00	530.00
13	14	15	2	111.00	530.00
14	15	16	2	111.00	530.00
15	16	17	2	111.00	530.00
16	17	18	2	111.00	530.00
17	18	19	2	111.00	530.00
18	19	20	2	117.00	635.00
18	19	38	2	117.00	635.00
20	19	38	2	124.00	730.00
19	20	21	2	124.00	730.00
19	20	22	2	115.00	610.00
21	20	22	2	121.00	685.00
20	22	23	2	120.00	560.00
20	22	25	2	120.00	560.00
23	22	25	2	120.00	560.00
22	23	24	2	120.00	505.00
22	23	29	2	120.00	560.00
24	23	29	2	120.00	505.00
22	25	26	2	120.00	560.00
22	25	37	2	120.00	560.00
26	25	37	2	120.00	560.00
25	26	27	2	120.00	560.00
25	26	31	2	120.00	560.00
27	26	31	2	120.00	560.00
26	27	28	2	120.00	505.00

```
26   27   29    2    120.00    560.00
28   27   29    2    120.00    505.00
23   29   27    2    120.00    560.00
23   29   30    2    120.00    505.00
27   29   30    2    120.00    505.00
26   31   32    2    120.00    505.00
26   31   33    2    120.00    560.00
32   31   33    2    120.00    505.00
31   33   34    2    120.00    505.00
31   33   35    2    120.00    560.00
34   33   35    2    120.00    505.00
33   35   36    2    120.00    505.00
33   35   37    2    120.00    560.00
36   35   37    2    120.00    505.00
25   37   35    2    120.00    560.00
25   37   38    2    120.00    560.00
35   37   38    2    120.00    560.00
19   38   37    2    115.00    610.00
19   38   39    2    124.00    730.00
37   38   39    2    121.00    685.00
[ dihedrals ]
; GROMOS improper dihedrals
;  ai   aj   ak   al  funct   angle     fc
23   22   29   24    2      0.00    167.42
22   23   25   20    2      0.00    167.42
25   22   26   37    2      0.00    167.42
26   25   27   31    2      0.00    167.42
27   26   29   28    2      0.00    167.42
29   23   27   30    2      0.00    167.42
37   25   35   38    2      0.00    167.42
31   26   33   32    2      0.00    167.42
33   31   35   34    2      0.00    167.42
35   37   33   36    2      0.00    167.42
20   22   21   19    2      0.00    167.42
38   37   39   19    2      0.00    167.42
19   20   38   18    2      0.00    167.42
```

```
[ dihedrals ]
;  ai   aj   ak   al  funct    ph0      cp     mult
1     2    3    4    1      0.00      5.92    3
2     3    4    5    1      0.00      5.92    3
3     4    5    6    1      0.00      5.92    3
4     5    6    7    1      0.00      5.92    3
5     6    7    8    1      0.00      5.92    3
6     7    8    9    1      0.00      5.92    3
7     8    9   10    1      0.00      5.92    3
8     9   10   11    1      0.00      5.92    3
9    10   11   12    1      0.00      5.92    3
10   11   12   13    1      0.00      5.92    3
11   12   13   14    1      0.00      5.92    3
12   13   14   15    1      0.00      5.92    3
13   14   15   16    1      0.00      5.92    3
14   15   16   17    1      0.00      5.92    3
15   16   17   18    1      0.00      5.92    3
16   17   18   19    1      0.00      5.92    3
17   18   19   20    1      0.00      3.77    6
38   19   20   22    1    180.00     41.80    2
20   19   38   37    1    180.00     41.80    2
19   20   22   25    1    180.00     41.80    2
25   22   23   29    1    180.00     41.80    2
20   22   25   37    1    180.00     41.80    2
22   23   29   27    1    180.00     41.80    2
37   25   26   31    1    180.00     41.80    2
26   25   37   35    1    180.00     41.80    2
25   26   27   29    1    180.00     41.80    2
25   26   31   33    1    180.00     41.80    2
26   27   29   23    1    180.00     41.80    2
26   31   33   35    1    180.00     41.80    2
31   33   35   37    1    180.00     41.80    2
33   35   37   25    1    180.00     41.80    2
25   37   38   19    1    180.00     41.80    2
[ exclusions ]
;  ai   aj  funct  ;  GROMOS 1-4 exclusions
```

19 23
19 25
19 35
20 26
20 29
20 37
22 27
22 31
22 35
22 38
23 26
23 37
25 29
25 33
26 35
26 38
27 33
27 37
29 31
31 37
33 38

F. ATB Molecules in SRC9064

Table F.1: List of the mol IDs for molecules obtained from the ATB and used for Athenaeum generation.

8	10	12	13	14	15	16	17	18	19	20	21
22	27	33	34	37	38	43	49	63	64	68	87
91	92	98	100	102	103	104	113	118	121	122	123
125	126	127	128	139	140	144	146	152	153	158	161
179	181	182	183	186	188	193	194	195	197	199	209
217	218	219	220	221	222	224	225	226	227	228	230
231	232	233	234	235	236	238	240	241	242	243	244
245	249	250	251	252	253	255	260	261	262	263	264
266	267	271	272	274	275	279	280	281	282	283	284
285	286	287	288	290	292	293	294	295	296	298	300
305	308	310	311	316	317	318	320	321	322	323	326
328	329	330	331	338	339	340	342	346	347	349	351
352	353	354	356	358	359	360	361	362	363	365	366
367	368	369	370	373	374	375	378	380	407	408	421
429	430	431	433	435	436	437	438	439	440	441	442
445	446	447	448	450	451	453	457	458	459	461	463
464	465	466	467	468	469	470	471	473	474	475	476
477	479	480	481	482	483	484	485	486	487	488	489
490	491	492	493	495	496	498	503	504	505	506	507
508	512	517	518	519	520	521	522	524	531	541	547
549	561	586	588	589	590	595	600	601	602	604	605
606	607	608	611	612	614	618	623	624	625	626	627
628	629	631	632	633	635	636	637	639	640	641	642
643	644	645	646	647	648	649	651	652	653	654	655
656	657	658	659	660	661	662	663	664	665	666	667
670	671	672	674	675	676	677	680	681	682	683	684

Continued on next page

Continued from previous page

685	686	687	688	689	690	691	692	694	695	699	700
701	702	703	706	707	708	709	710	711	712	713	714
715	716	717	718	719	720	721	722	744	745	751	752
753	755	756	757	758	759	761	763	764	766	767	768
769	770	771	772	773	774	775	776	777	778	780	781
782	783	784	785	786	787	788	789	790	791	792	793
794	796	797	798	799	800	801	802	803	804	806	807
808	809	810	811	812	813	814	815	816	817	818	819
820	821	822	823	824	825	826	827	828	829	836	842
843	845	846	847	848	850	851	853	854	856	858	859
860	861	865	867	868	869	870	871	873	874	875	876
877	878	879	880	881	882	883	884	885	886	887	888
889	890	891	892	894	895	902	903	904	907	908	909
910	914	916	917	918	919	920	921	922	923	924	925
926	927	928	929	930	931	932	933	934	935	936	937
938	939	940	941	942	943	944	945	946	947	948	949
950	951	952	953	954	955	956	957	958	959	960	961
962	963	964	965	966	967	968	969	970	971	972	973
974	976	977	978	979	981	982	983	984	985	987	988
989	990	991	992	993	994	995	996	997	998	999	1000
1001	1002	1003	1004	1005	1006	1007	1008	1009	1010	1011	1012
1013	1014	1015	1016	1017	1018	1019	1020	1021	1022	1023	1024
1025	1026	1027	1028	1029	1030	1031	1032	1033	1034	1035	1036
1037	1038	1039	1040	1042	1043	1044	1045	1046	1053	1056	1057
1058	1061	1065	1070	1075	1076	1078	1081	1088	1089	1090	1091
1094	1096	1118	1121	1147	1149	1151	1160	1161	1167	1171	1173
1174	1175	1176	1177	1178	1179	1180	1181	1183	1186	1187	1189
1192	1193	1194	1196	1198	1199	1201	1202	1204	1206	1208	1210
1211	1212	1214	1215	1254	1255	1256	1257	1258	1259	1260	1262
1263	1266	1269	1283	1284	1285	1287	1291	1303	1307	1313	1327
1332	1335	1346	1347	1357	1359	1360	1361	1395	1400	1401	1402
1403	1407	1408	1414	1430	1476	1496	1509	1527	1609	1620	1629
1630	1631	1633	1634	1637	1638	1661	1672	1673	1674	1683	1685
1688	1691	1693	1699	1700	1701	1703	1707	1715	1716	1717	1718

Continued on next page

Continued from previous page

1721	1742	1745	1747	1748	1754	1757	1762	1774	1775	1776	1777
1781	1782	1783	1789	1797	1799	1807	1808	1809	1811	1826	1827
1830	1848	1850	1851	1853	1854	1855	1856	1859	1860	1861	1862
1863	1864	1865	1866	1867	1868	1873	1874	1875	1876	1877	1878
1879	1880	1881	1882	1883	1884	1885	1886	1887	1892	1907	1909
1910	1911	1912	1913	1914	1915	1917	1918	1919	1920	1921	1922
1923	1925	1926	1927	1928	1929	1930	1931	1932	1933	1934	1935
1936	1939	1940	1941	1942	1943	1944	1945	1947	1948	1950	1951
1952	1953	1954	1956	1959	1962	1964	1967	1969	1972	1975	1976
1980	1981	1983	1984	1991	1992	1993	1994	2012	2015	2027	2029
2042	2043	2046	2066	2081	2089	2098	2103	2112	2120	2121	2123
2125	2126	2127	2136	2137	2152	2154	2159	2174	2183	2187	2192
2202	2204	2205	2212	2220	2221	2222	2223	2225	2227	2228	2233
2234	2236	2237	2242	2258	2268	2270	2271	2272	2273	2274	2275
2289	2290	2292	2293	2299	2300	2314	2319	2330	2337	2350	2351
2356	2357	2367	2368	2369	2370	2371	2373	2374	2377	2378	2379
2381	2403	2404	2408	2409	2411	2412	2413	2415	2417	2418	2421
2422	2424	2425	2426	2427	2428	2429	2430	2433	2434	2435	2437
2438	2439	2441	2446	2447	2448	2449	2451	2461	2462	2476	2477
2481	2484	2486	2487	2488	2489	2490	2491	2492	2493	2494	2495
2496	2498	2499	2500	2501	2502	2503	2504	2505	2506	2509	2511
2514	2518	2527	2528	2529	2530	2531	2532	2533	2534	2535	2541
2542	2543	2544	2545	2546	2547	2548	2549	2555	2562	2563	2567
2581	2582	2583	2589	2590	2591	2592	2596	2610	2616	2617	2618
2619	2620	2622	2624	2625	2626	2627	2629	2638	2639	2640	2641
2645	2649	2653	2654	2656	2658	2660	2661	2662	2663	2664	2665
2666	2667	2679	2680	2681	2693	2700	2701	2702	2704	2708	2711
2717	2735	2776	2778	2796	2797	2801	2802	2803	2804	2805	2806
2807	2808	2809	2810	2811	2812	2813	2814	2815	2816	2817	2818
2819	2820	2821	2822	2825	2827	2828	2829	2830	2831	2832	2833
2868	2869	2870	2878	2879	2883	2886	2890	2892	2894	2896	2897
2898	2899	2900	2901	2903	2904	2905	2906	2914	2915	2916	2925
2928	2935	2941	2942	2944	2945	2946	2947	2948	2949	2950	2951
2952	2953	2954	2955	2956	2957	2958	2961	2962	2966	2968	2969

Continued on next page

Continued from previous page

2970	2976	2977	2978	2979	2980	2981	2982	2983	2984	2985	2986
2987	2988	2989	2990	2991	2993	2994	2995	2996	2997	2999	3004
3005	3007	3008	3009	3013	3014	3023	3033	3034	3035	3056	3060
3080	3083	3085	3109	3111	3112	3113	3116	3119	3125	3126	3130
3131	3144	3148	3162	3163	3164	3165	3166	3167	3168	3169	3170
3171	3172	3174	3175	3176	3177	3178	3179	3180	3181	3182	3186
3187	3188	3189	3196	3197	3199	3200	3201	3208	3214	3218	3221
3232	3240	3243	3244	3245	3246	3249	3264	3265	3272	3273	3274
3275	3280	3288	3289	3291	3292	3300	3301	3302	3313	3318	3321
3323	3324	3326	3327	3328	3329	3335	3336	3337	3338	3339	3340
3341	3345	3346	3348	3349	3351	3353	3359	3360	3361	3362	3363
3364	3369	3371	3372	3373	3374	3375	3376	3377	3378	3379	3381
3382	3383	3384	3385	3386	3387	3388	3389	3391	3392	3393	3394
3396	3397	3398	3399	3400	3402	3413	3415	3416	3417	3418	3419
3420	3421	3422	3423	3424	3425	3427	3428	3429	3430	3431	3432
3433	3434	3436	3438	3439	3441	3442	3443	3444	3445	3446	3447
3448	3449	3450	3451	3452	3458	3461	3462	3465	3466	3470	3472
3473	3474	3475	3476	3477	3478	3480	3481	3482	3489	3490	3491
3493	3494	3495	3497	3498	3511	3512	3513	3516	3517	3518	3519
3520	3521	3522	3523	3524	3525	3526	3527	3528	3529	3533	3538
3539	3540	3541	3542	3544	3545	3546	3547	3548	3549	3550	3551
3553	3554	3555	3588	3591	3596	3613	3617	3619	3631	3632	3633
3635	3636	3637	3639	3640	3642	3643	3647	3648	3649	3650	3651
3652	3653	3654	3655	3656	3657	3658	3659	3660	3661	3662	3663
3666	3667	3669	3670	3671	3672	3673	3674	3675	3677	3678	3679
3680	3681	3684	3685	3686	3688	3689	3690	3692	3693	3696	3697
3699	3700	3701	3702	3703	3705	3706	3708	3709	3710	3712	3714
3715	3716	3718	3719	3720	3725	3726	3727	3729	3730	3731	3732
3733	3734	3735	3736	3737	3738	3739	3740	3741	3742	3743	3744
3745	3750	3758	3759	3761	3763	3764	3765	3767	3772	3773	3774
3775	3776	3777	3778	3779	3780	3781	3782	3783	3784	3786	3787
3789	3791	3793	3794	3795	3797	3798	3799	3800	3801	3802	3803
3804	3805	3806	3808	3809	3810	3811	3813	3814	3815	3816	3817
3822	3823	3824	3825	3826	3827	3834	3839	3840	3841	3843	3855

Continued on next page

Continued from previous page

3856	3857	3859	3861	3865	3866	3867	3868	3869	3870	3871	3872
3873	3875	3877	3880	3881	3882	3883	3885	3886	3887	3888	3889
3890	3891	3892	3894	3895	3896	3897	3898	3901	3902	3904	3906
3909	3911	3912	3913	3914	3915	3919	3921	3938	3943	3945	3946
3949	3951	3953	3955	3956	3957	3958	3962	3963	3964	3966	3967
3968	3970	3971	3972	3973	3974	3977	3978	3979	3982	3983	3985
3986	3987	3989	3990	3991	3992	3994	3995	3996	3997	3998	4003
4005	4007	4008	4009	4010	4012	4016	4017	4019	4021	4022	4023
4024	4025	4026	4027	4028	4043	4045	4049	4050	4052	4053	4054
4055	4056	4057	4059	4060	4068	4076	4083	4085	4097	4098	4103
4106	4107	4108	4109	4112	4114	4117	4120	4122	4126	4128	4132
4133	4136	4155	4156	4162	4168	4169	4185	4191	4210	4214	4219
4250	4254	4260	4264	4269	4270	4274	4275	4276	4281	4282	4283
4287	4288	4289	4292	4296	4300	4305	4306	4323	4331	4336	4341
4347	4348	4349	4355	4356	4358	4368	4369	4370	4386	4388	4389
4390	4391	4392	4393	4403	4404	4405	4413	4415	4432	4435	4437
4438	4439	4440	4442	4445	4447	4448	4449	4450	4451	4452	4458
4459	4460	4468	4471	4472	4473	4475	4478	4479	4480	4490	4491
4505	4506	4518	4524	4525	4550	4558	4560	4563	4564	4568	4595
4596	4606	4610	4621	4622	4631	4632	4633	4634	4635	4636	4638
4639	4640	4643	4644	4645	4646	4656	4657	4659	4663	4665	4673
4677	4680	4681	4686	4687	4695	4696	4697	4698	4699	4700	4702
4711	4713	4716	4717	4722	4726	4727	4728	4729	4731	4733	4737
4738	4739	4742	4743	4749	4751	4765	4766	4767	4769	4770	4771
4778	4793	4797	4801	4807	4818	4828	4831	4835	4839	4848	4852
4855	4858	4869	4870	4871	4873	4918	4928	4940	4941	4942	4943
4944	4946	4947	4964	4967	4979	4980	4985	4987	4992	4993	4996
5010	5014	5017	5024	5025	5028	5030	5032	5039	5040	5041	5042
5043	5044	5045	5046	5048	5049	5050	5051	5052	5053	5054	5056
5057	5058	5059	5060	5061	5062	5063	5064	5094	5102	5103	5104
5105	5106	5107	5109	5110	5115	5138	5147	5148	5149	5150	5151
5153	5154	5156	5159	5160	5161	5196	5200	5206	5207	5208	5211
5244	5249	5250	5251	5258	5259	5260	5266	5269	5271	5272	5273
5274	5275	5276	5277	5278	5280	5285	5286	5289	5290	5291	5292

Continued on next page

Continued from previous page

5293	5297	5313	5314	5319	5321	5324	5333	5336	5337	5342	5355
5362	5367	5378	5379	5381	5382	5383	5387	5388	5394	5395	5397
5399	5417	5429	5430	5432	5434	5435	5438	5442	5443	5445	5457
5497	5500	5501	5503	5504	5505	5506	5523	5525	5531	5532	5533
5537	5541	5548	5552	5560	5563	5567	5568	5570	5571	5573	5576
5579	5582	5583	5584	5595	5598	5600	5610	5611	5614	5617	5625
5632	5637	5640	5654	5655	5671	5697	5698	5699	5701	5707	5710
5721	5723	5739	5765	5767	5768	5771	5773	5783	5785	5791	5824
5859	5868	5869	5872	5874	5878	5883	5885	5886	5887	5892	5893
5895	5898	5910	5912	5913	5914	5927	5929	5934	5941	5942	5949
5954	5955	5956	5957	5958	5962	5970	5980	5984	5985	5987	5993
6000	6007	6021	6034	6035	6037	6040	6041	6043	6044	6045	6046
6063	6064	6068	6069	6072	6076	6077	6078	6084	6085	6086	6100
6101	6102	6108	6111	6136	6137	6138	6151	6152	6154	6159	6160
6161	6162	6163	6164	6165	6166	6167	6171	6180	6183	6187	6188
6190	6191	6192	6195	6199	6216	6222	6223	6227	6236	6264	6265
6278	6282	6295	6300	6302	6311	6316	6318	6319	6322	6325	6406
6407	6428	6442	6443	6444	6447	6448	6449	6456	6477	6486	6487
6490	6491	6501	6504	6505	6511	6518	6519	6522	6523	6545	6548
6555	6557	6558	6574	6575	6578	6586	6589	6592	6594	6600	6606
6612	6620	6621	6622	6623	6637	6641	6649	6669	6673	6679	6683
6685	6687	6691	6698	6699	6707	6711	6715	6733	6734	6740	6741
6744	6747	6750	6753	6771	6783	6787	6800	6803	6804	6814	6815
6816	6839	6840	6841	6842	6843	6851	6862	6867	6872	6881	6897
6910	6914	6919	6921	6925	6936	6946	6954	6956	6959	6979	6980
7004	7014	7029	7030	7039	7046	7079	7101	7104	7105	7134	7147
7148	7154	7155	7174	7175	7176	7177	7178	7179	7190	7191	7193
7199	7231	7233	7234	7236	7252	7268	7272	7275	7276	7277	7278
7286	7287	7296	7299	7300	7301	7302	7304	7305	7306	7308	7309
7310	7311	7312	7313	7315	7316	7327	7328	7329	7330	7333	7337
7338	7341	7370	7371	7376	7377	7379	7395	7402	7403	7404	7406
7407	7412	7413	7446	7453	7454	7457	7463	7481	7487	7491	7497
7498	7502	7508	7520	7521	7522	7533	7534	7539	7540	7541	7542
7544	7552	7555	7556	7577	7588	7592	7593	7596	7597	7598	7607

Continued on next page

Continued from previous page

7611	7612	7615	7616	7621	7629	7637	7638	7640	7641	7653	7655
7657	7658	7659	7671	7679	7686	7688	7689	7690	7693	7704	7710
7712	7716	7717	7718	7719	7722	7730	7731	7735	7741	7742	7744
7759	7760	7763	7770	7771	7774	7775	7781	7787	7793	7794	7795
7797	7798	7803	7811	7823	7826	7863	7864	7866	7868	7869	7874
7875	7896	7897	7918	7921	7923	7939	7940	7941	7943	7946	7952
7956	7957	7958	7959	7962	7963	7964	7965	7966	7967	7968	7969
7970	7982	8000	8001	8003	8005	8006	8008	8012	8016	8017	8018
8019	8020	8021	8022	8023	8024	8025	8026	8027	8028	8029	8030
8031	8033	8035	8036	8037	8038	8039	8044	8045	8046	8047	8048
8049	8058	8062	8064	8069	8070	8071	8077	8080	8082	8083	8089
8092	8108	8109	8110	8111	8114	8170	8176	8211	8216	8219	8222
8225	8251	8254	8257	8263	8266	8272	8299	8302	8305	8315	8333
8336	8342	8350	8354	8362	8373	8383	8385	8387	8394	8395	8397
8414	8424	8428	8429	8432	8433	8434	8435	8436	8437	8438	8439
8440	8441	8442	8443	8444	8445	8446	8447	8448	8449	8450	8459
8468	8480	8481	8493	8510	8513	8514	8515	8517	8518	8519	8526
8532	8533	8540	8542	8543	8546	8547	8551	8553	8556	8566	8570
8576	8577	8578	8596	8603	8608	8620	8622	8627	8628	8631	8637
8638	8639	8640	8649	8657	8658	8659	8660	8667	8675	8677	8683
8688	8697	8698	8701	8702	8703	8704	8717	8739	8742	8750	8751
8752	8753	8770	8774	8775	8777	8779	8780	8785	8786	8788	8795
8801	8805	8806	8817	8827	8830	8839	8842	8851	8855	8856	8860
8861	8863	8865	8874	8877	8878	8879	8891	8896	8902	8915	8936
8942	8944	8955	8957	8958	8959	8961	8962	8963	8965	8966	8968
8969	8971	8972	8983	8992	8994	8995	8996	8998	8999	9000	9019
9020	9025	9026	9027	9028	9031	9032	9051	9052	9055	9059	9060
9064	9066	9067	9069	9070	9074	9075	9076	9078	9086	9088	9089
9096	9099	9102	9105	9106	9109	9111	9114	9115	9119	9120	9121
9128	9129	9131	9133	9136	9140	9141	9142	9143	9144	9148	9150
9152	9155	9156	9157	9159	9161	9164	9167	9168	9170	9172	9174
9175	9176	9178	9179	9180	9182	9183	9184	9185	9186	9188	9191
9205	9209	9212	9213	9214	9218	9222	9223	9230	9252	9254	9267
9277	9283	9284	9289	9291	9300	9302	9303	9314	9315	9316	9329

Continued on next page

Continued from previous page

9330	9331	9357	9358	9359	9360	9364	9367	9368	9369	9378	9380
9381	9383	9419	9420	9427	9428	9471	9475	9483	9521	9526	9527
9529	9562	9564	9567	9568	9573	9575	9576	9577	9579	9580	9581
9582	9583	9584	9585	9586	9587	9588	9589	9592	9593	9594	9595
9597	9598	9599	9600	9603	9612	9613	9614	9615	9624	9629	9641
9642	9645	9646	9647	9648	9649	9651	9652	9653	9654	9655	9657
9658	9659	9660	9661	9670	9672	9682	9685	9687	9691	9692	9693
9695	9696	9697	9700	9701	9703	9706	9707	9708	9709	9710	9711
9712	9715	9718	9719	9720	9722	9724	9726	9728	9731	9732	9734
9742	9746	9748	9750	9753	9758	9762	9764	9765	9768	9773	9775
9776	9777	9778	9780	9791	9797	9798	9800	9801	9802	9803	9804
9805	9806	9809	9811	9814	9822	9824	9825	9826	9827	9829	9830
9837	9840	9845	9846	9847	9848	9849	9851	9854	9856	9857	9859
9860	9861	9862	9863	9865	9866	9868	9869	9870	9871	9873	9877
9878	9880	9881	9887	9888	9891	9892	9893	9894	9897	9901	9903
9909	9910	9911	9916	9918	9921	9929	9930	9941	9942	9944	9945
9946	9952	9953	9954	9955	9957	9960	9963	9966	9970	9985	9986
9987	9993	9995	9998	9999	10000	10001	10003	10007	10008	10010	10012
10013	10017	10018	10019	10020	10021	10022	10023	10025	10027	10028	10029
10031	10037	10046	10047	10050	10051	10052	10053	10057	10058	10059	10060
10063	10066	10071	10074	10078	10080	10083	10085	10089	10091	10094	10096
10098	10099	10100	10101	10102	10103	10104	10107	10109	10110	10112	10113
10122	10124	10125	10133	10134	10135	10136	10139	10140	10142	10149	10150
10152	10153	10155	10156	10157	10159	10160	10163	10165	10169	10172	10174
10175	10176	10179	10183	10184	10185	10188	10189	10190	10192	10199	10200
10201	10205	10208	10213	10216	10220	10221	10230	10232	10233	10236	10237
10239	10240	10242	10245	10248	10251	10252	10253	10254	10261	10264	10265
10268	10273	10274	10276	10277	10280	10281	10284	10285	10287	10290	10291
10292	10295	10298	10299	10301	10303	10305	10307	10308	10311	10313	10315
10317	10318	10321	10324	10326	10328	10331	10333	10339	10340	10341	10343
10345	10357	10361	10365	10367	10369	10370	10371	10373	10376	10377	10380
10381	10382	10385	10386	10389	10392	10393	10395	10397	10398	10405	10408
10411	10412	10414	10417	10418	10422	10423	10425	10426	10429	10431	10432
10435	10438	10440	10447	10449	10456	10463	10465	10469	10473	10474	10477

Continued on next page

Continued from previous page

10478 10481 10485 10486 10489 10491 10492 10494 10501 10502 10503 10506
10507 10510 10513 10515 10516 10517 10519 10523 10524 10526 10527 10528
10531 10532 10535 10536 10537 10538 10540 10541 10543 10545 10546 10548
10549 10554 10555 10556 10557 10558 10561 10565 10566 10568 10569 10571
10572 10573 10575 10576 10577 10578 10581 10583 10584 10586 10591 10598
10600 10601 10603 10604 10605 10606 10611 10615 10616 10618 10619 10621
10623 10627 10630 10632 10633 10634 10636 10637 10638 10641 10646 10647
10652 10657 10660 10662 10664 10665 10669 10670 10671 10672 10673 10678
10679 10680 10688 10689 10691 10693 10694 10695 10697 10698 10700 10701
10702 10707 10712 10713 10714 10716 10720 10721 10722 10723 10724 10725
10726 10728 10731 10735 10741 10756 10759 10762 10769 10770 10772 10775
10778 10780 10781 10782 10787 10790 10792 10799 10802 10803 10804 10807
10812 10816 10817 10824 10827 10828 10834 10835 10839 10841 10842 10843
10846 10847 10848 10850 10854 10860 10863 10866 10867 10868 10870 10876
10877 10878 10886 10889 10896 10897 10908 10910 10911 10914 10917 10918
10920 10921 10922 10923 10925 10927 10928 10930 10936 10938 10942 10944
10946 10947 10948 10952 10953 10956 10958 10960 10961 10963 10965 10971
10974 10975 10976 10984 10989 10990 10991 10996 10998 11000 11002 11003
11004 11005 11006 11008 11009 11011 11012 11014 11015 11018 11019 11020
11021 11023 11028 11030 11031 11034 11037 11038 11039 11044 11045 11048
11053 11054 11057 11059 11060 11063 11066 11068 11073 11083 11086 11089
11095 11098 11102 11103 11104 11105 11107 11110 11113 11114 11115 11117
11123 11125 11126 11127 11130 11131 11133 11135 11139 11140 11145 11146
11147 11148 11150 11151 11158 11159 11160 11161 11167 11174 11175 11176
11178 11182 11185 11188 11189 11191 11192 11196 11197 11205 11206 11208
11210 11212 11215 11216 11217 11218 11219 11220 11222 11227 11228 11229
11231 11235 11239 11241 11243 11246 11247 11249 11251 11253 11256 11257
11260 11261 11262 11263 11268 11269 11270 11272 11274 11275 11277 11279
11280 11286 11288 11294 11299 11301 11304 11307 11310 11316 11317 11319
11321 11334 11335 11336 11338 11342 11344 11345 11346 11351 11352 11356
11357 11361 11364 11366 11368 11369 11370 11372 11373 11375 11383 11385
11392 11396 11397 11398 11399 11400 11403 11407 11409 11410 11414 11419
11421 11426 11427 11430 11433 11434 11435 11439 11440 11441 11442 11445
11446 11448 11450 11452 11459 11461 11468 11474 11476 11477 11480 11485

Continued on next page

Continued from previous page

11486 11487 11488 11489 11491 11492 11494 11495 11497 11501 11502 11504
11505 11507 11509 11510 11512 11514 11521 11523 11526 11527 11530 11532
11533 11537 11539 11540 11543 11547 11550 11551 11553 11555 11561 11570
11572 11574 11576 11579 11582 11589 11591 11594 11595 11600 11613 11617
11620 11622 11626 11628 11630 11632 11634 11636 11638 11642 11650 11653
11654 11662 11664 11665 11668 11677 11679 11680 11686 11688 11689 11693
11699 11701 11702 11706 11707 11709 11719 11720 11722 11724 11725 11726
11727 11728 11731 11733 11734 11739 11741 11744 11745 11746 11749 11756
11757 11759 11765 11767 11769 11772 11776 11777 11782 11783 11784 11787
11791 11792 11794 11797 11798 11799 11800 11801 11805 11808 11809 11810
11811 11812 11814 11816 11820 11835 11837 11840 11841 11847 11851 11852
11860 11865 11866 11868 11873 11878 11879 11880 11885 11889 11891 11892
11894 11896 11897 11901 11902 11908 11909 11910 11913 11914 11915 11917
11920 11924 11925 11926 11928 11929 11932 11933 11939 11944 11945 11947
11948 11949 11950 11951 11956 11958 11960 11961 11967 11969 11972 11973
11988 11993 11995 11998 11999 12001 12002 12003 12004 12005 12009 12014
12018 12020 12023 12026 12027 12031 12032 12035 12036 12039 12041 12043
12048 12050 12052 12053 12054 12055 12059 12063 12065 12073 12074 12075
12077 12080 12085 12093 12096 12099 12101 12108 12109 12113 12115 12117
12120 12124 12126 12130 12132 12133 12137 12138 12140 12142 12143 12144
12148 12149 12151 12152 12154 12158 12164 12165 12169 12170 12172 12175
12179 12180 12182 12183 12185 12186 12188 12190 12192 12193 12195 12197
12203 12204 12205 12206 12210 12212 12214 12218 12219 12220 12221 12222
12224 12227 12232 12234 12235 12237 12241 12242 12243 12247 12253 12255
12257 12258 12259 12260 12261 12266 12267 12270 12273 12276 12279 12283
12287 12290 12291 12293 12297 12298 12299 12301 12312 12319 12321 12323
12324 12327 12330 12331 12332 12337 12340 12342 12344 12347 12348 12350
12355 12356 12357 12358 12360 12369 12374 12376 12377 12380 12382 12383
12386 12390 12394 12397 12399 12400 12401 12406 12407 12414 12417 12424
12426 12428 12430 12432 12433 12435 12437 12439 12444 12447 12448 12456
12458 12460 12461 12463 12465 12466 12468 12469 12471 12476 12477 12478
12490 12491 12493 12495 12497 12498 12499 12501 12502 12503 12512 12514
12516 12520 12524 12526 12527 12530 12532 12536 12541 12547 12551 12552
12553 12554 12555 12556 12559 12561 12562 12563 12564 12567 12569 12575

Continued on next page

Continued from previous page

12576 12578 12580 12582 12583 12587 12588 12590 12592 12593 12594 12596
12597 12601 12602 12606 12609 12611 12613 12615 12618 12622 12623 12634
12635 12637 12638 12640 12642 12644 12645 12648 12650 12651 12655 12656
12658 12660 12661 12662 12665 12666 12670 12672 12674 12675 12681 12682
12685 12686 12688 12689 12694 12697 12702 12706 12707 12713 12719 12723
12726 12727 12733 12744 12746 12747 12750 12752 12760 12763 12766 12767
12770 12771 12773 12775 12777 12779 12794 12804 12806 12807 12808 12811
12812 12816 12818 12822 12825 12831 12832 12833 12835 12837 12841 12842
12844 12846 12849 12852 12854 12856 12857 12858 12860 12862 12863 12866
12871 12872 12874 12875 12876 12877 12879 12882 12883 12884 12885 12888
12889 12890 12891 12892 12898 12904 12905 12914 12918 12919 12922 12925
12927 12928 12930 12931 12937 12938 12940 12942 12943 12944 12945 12947
12950 12954 12956 12959 12961 12963 12969 12974 12982 12986 12987 12992
12994 12996 12998 13002 13003 13016 13024 13025 13026 13027 13031 13035
13036 13038 13039 13045 13046 13052 13054 13055 13057 13065 13067 13070
13075 13076 13079 13081 13089 13094 13096 13097 13101 13105 13110 13118
13121 13122 13123 13125 13127 13133 13134 13140 13143 13147 13151 13159
13160 13161 13164 13166 13168 13170 13172 13174 13178 13180 13181 13182
13183 13184 13186 13196 13199 13202 13205 13214 13216 13219 13222 13231
13232 13235 13242 13243 13246 13248 13261 13267 13269 13278 13286 13290
13292 13295 13297 13299 13302 13307 13309 13315 13316 13317 13320 13321
13322 13323 13325 13327 13328 13334 13335 13339 13340 13343 13346 13348
13351 13352 13353 13356 13357 13358 13360 13361 13364 13366 13369 13371
13375 13377 13386 13387 13391 13394 13395 13397 13401 13406 13408 13409
13410 13411 13416 13417 13421 13423 13424 13425 13426 13431 13433 13434
13440 13442 13444 13446 13448 13450 13451 13453 13454 13456 13457 13460
13461 13466 13467 13473 13474 13479 13480 13486 13488 13489 13496 13498
13500 13502 13508 13509 13511 13516 13521 13528 13536 13537 13538 13543
13548 13550 13553 13554 13559 13563 13564 13565 13566 13568 13574 13578
13581 13588 13591 13595 13596 13604 13605 13606 13609 13612 13614 13616
13619 13620 13621 13622 13624 13625 13630 13631 13634 13635 13640 13644
13651 13652 13653 13654 13658 13661 13663 13664 13665 13666 13667 13669
13670 13679 13682 13683 13686 13687 13689 13692 13693 13695 13700 13701
13702 13704 13708 13709 13710 13711 13715 13720 13724 13730 13734 13736

Continued on next page

Continued from previous page

13739 13749 13751 13754 13755 13757 13759 13760 13764 13766 13768 13777
13778 13782 13783 13784 13785 13788 13789 13791 13798 13802 13807 13813
13817 13818 13820 13827 13834 13839 13840 13842 13846 13847 13854 13857
13859 13874 13875 13877 13886 13888 13892 13894 13897 13898 13902 13903
13906 13907 13910 13912 13913 13919 13934 13936 13938 13939 13941 13948
13950 13951 13952 13954 13955 13958 13961 13965 13967 13980 13982 13983
13986 13987 13990 13992 14013 14016 14021 14025 14026 14027 14033 14047
14055 14056 14064 14065 14072 14076 14079 14083 14086 14088 14089 14091
14093 14098 14101 14108 14111 14112 14115 14118 14119 14126 14127 14131
14132 14136 14144 14147 14152 14154 14157 14159 14163 14165 14170 14171
14174 14179 14181 14185 14187 14191 14192 14203 14204 14209 14211 14213
14214 14218 14222 14224 14226 14228 14233 14239 14240 14247 14250 14253
14254 14265 14267 14284 14291 14304 14316 14317 14318 14322 14327 14328
14329 14333 14335 14336 14348 14350 14351 14354 14356 14358 14360 14361
14365 14367 14373 14375 14388 14394 14398 14402 14404 14410 14419 14428
14430 14431 14439 14444 14445 14449 14452 14454 14458 14461 14468 14471
14473 14476 14480 14481 14482 14485 14486 14489 14493 14494 14499 14506
14508 14509 14511 14514 14521 14522 14525 14529 14531 14536 14537 14539
14541 14546 14547 14548 14551 14556 14557 14561 14562 14563 14568 14570
14582 14583 14586 14594 14596 14599 14600 14606 14607 14610 14611 14614
14617 14626 14627 14632 14635 14638 14641 14642 14644 14647 14648 14649
14655 14659 14661 14662 14670 14671 14673 14674 14680 14692 14714 14715
14722 14736 14758 14761 14765 14766 14767 14768 14785 14788 14789 14798
14802 14803 14804 14807 14810 14816 14819 14820 14823 14824 14826 14830
14834 14835 14836 14837 14840 14847 14849 14866 14867 14870 14875 14877
14881 14883 14885 14886 14888 14890 14896 14898 14899 14900 14902 14904
14905 14907 14911 14924 14925 14926 14928 14932 14934 14944 14945 14950
14952 14957 14971 14974 14983 14986 14990 14993 14994 14995 14998 14999
15004 15005 15008 15010 15012 15015 15016 15018 15019 15020 15024 15026
15027 15030 15035 15039 15042 15051 15052 15057 15059 15071 15075 15076
15080 15081 15082 15084 15085 15086 15087 15091 15092 15093 15094 15097
15098 15102 15107 15112 15114 15115 15116 15119 15122 15123 15127 15128
15130 15131 15134 15136 15138 15139 15150 15151 15154 15158 15162 15166
15168 15169 15173 15175 15180 15184 15185 15187 15188 15193 15197 15198

Continued on next page

Continued from previous page

15202 15203 15205 15207 15214 15215 15216 15217 15220 15221 15222 15225
15229 15230 15232 15233 15234 15243 15244 15247 15248 15250 15254 15258
15259 15264 15270 15273 15278 15282 15283 15285 15290 15291 15294 15301
15302 15303 15304 15308 15309 15310 15318 15319 15320 15322 15326 15327
15330 15331 15332 15334 15338 15339 15343 15347 15349 15353 15361 15364
15367 15373 15375 15376 15378 15381 15387 15390 15392 15397 15400 15407
15408 15409 15416 15419 15420 15425 15426 15432 15442 15453 15454 15485
15496 15497 15501 15525 15538 15591 15593 15607 15608 15610 15618 15627
15640 15641 15650 15651 15663 15696 15698 15701 15710 15724 15725 15731
15741 15742 15743 15754 15763 15793 15805 15810 15812 15814 15823 15845
15846 15852 15854 15859 15863 15896 15902 15903 15919 15928 15943 15952
15970 15971 15972 15973 15977 15980 15981 15982 16004 16009 16014 16026
16043 16046 16068 16069 16071 16074 16082 16099 16112 16139 16143 16145
16147 16148 16161 16185 16215 16216 16238 16246 16276 16279 16282 16283
16284 16303 16306 16308 16314 16317 16320 16326 16327 16348 16353 16354
16355 16356 16357 16358 16379 16452 16461 16466 16470 16477 16510 16530
16531 16538 16539 16542 16543 16544 16562 16567 16570 16571 16574 16575
16576 16578 16579 16580 16582 16585 16586 16587 16589 16594 16595 16596
16597 16600 16602 16605 16606 16607 16609 16618 16620 16622 16631 16639
16658 16684 16706 16712 16720 16733 16735 16742 16758 16759 16762 16763
16764 16765 16767 16768 16769 16770 16771 16772 16773 16774 16775 16779
16782 16796 16798 16799 16800 16801 16802 16803 16804 16806 16807 16808
16809 16813 16814 16816 16817 16818 16820 16822 16823 16824 16825 16826
16827 16828 16829 16830 16831 16832 16833 16834 16835 16836 16837 16838
16839 16840 16841 16842 16843 16844 16845 16846 16847 16848 16849 16850
16851 16867 16871 16872 16875 16886 16887 16902 16911 16917 16937 16947
16950 16964 16974 16976 16982 16993 16994 16995 16996 17000 17003 17004
17005 17006 17009 17011 17012 17013 17014 17015 17016 17017 17018 17019
17020 17022 17023 17026 17027 17029 17030 17031 17032 17034 17035 17036
17037 17041 17042 17044 17047 17050 17051 17052 17053 17057 17068 17069
17070 17071 17072 17076 17084 17092 17093 17102 17108 17111 17117 17119
17121 17123 17125 17126 17132 17133 17555 17556 17564 17565 17620 17621
17634 17635 17640 17647 17650 17666 17687 17701 17729 17758 17760 17776
17796 17826 17840 17866 17917 17918 17919 17920 17921 17922 17923 17924

Continued on next page

Continued from previous page

17925 17926 17927 17928 17929 17930 17985 17999 18003 18043 18048 18084
18108 18132 18134 18172 18180 18185 18190 18191 18192 18200 18201 18210
18211 18212 18213 18214 18215 18216 18217 18218 18224 18261 18267 18289
18295 18296 18307 18325 18327 18328 18342 18344 18362 18366 18374 18380
18401 18428 18439 18456 18525 18529 18543 18548 18578 18592 18596 18603
18616 18617 18618 18648 18650 18654 18663 18664 18665 18666 18667 18668
18669 18670 18671 18674 18697 18698 18701 18702 18713 18714 18715 18717
18718 18719 18720 18721 18722 18723 18724 18725 18726 18727 18728 18729
18730 18731 18732 18733 18734 18735 18736 18737 18738 18739 18740 18741
18742 18743 18744 18746 18748 18756 18757 18758 18762 18768 18797 18813
18814 18815 18827 18828 18833 18840 18844 18845 18846 18847 18848 18849
18850 18851 18852 18853 18856 18857 18858 18859 18860 18872 18873 18877
18879 18887 18888 18908 18913 18918 18936 18950 18951 18970 18983 18985
19001 19002 19008 19011 19012 19014 19015 19021 19027 19028 19029 19036
19037 19038 19040 19041 19042 19043 19044 19045 19047 19049 19059 19060
19062 19070 19071 19073 19079 19090 19092 19095 19097 19098 19099 19101
19114 19115 19131 19135 19136 19149 19174 19210 19221 19228 19237 19262
19267 19281 19298 19315 19327 19359 19367 19368 19369 19375 19377 19385
19387 19394 19401 19403 19412 19413 19415 19451 19462 19469 19470 19480
19499 19514 19517 19518 19519 19526 19543 19549 19589 19590 19595 19596
19597 19605 19607 19613 19616 19625 19630 19632 19636 19637 19639 19641
19643 19657 19658 19661 19663 19666 19668 19669 19689 19699 19702 19704
19705 19714 19718 19719 19720 19750 19755 19759 19762 19764 19765 19774
19775 19783 19787 19818 19833 19834 19838 19842 19848 19855 19867 19870
19872 19886 19887 19888 19895 19899 19909 19935 19944 19949 19950 19977
19984 19991 20034 20047 20066 20068 20072 20076 20077 20079 20103 20107
20120 20136 20139 20164 20165 20179 20180 20186 20187 20191 20209 20222
20223 20225 20259 20260 20263 20265 20270 20272 20273 20274 20276 20278
20296 20303 20311 20325 20359 20362 20368 20395 20396 20397 20419 20423
20452 20460 20565 20567 20571 20584 20598 20601 20605 20606 20608 20619
20633 20645 20646 20647 20654 20657 20658 20659 20660 20661 20662 20663
20664 20665 20667 20668 20679 20680 20702 20705 20708 20711 20719 20729
20735 20776 20813 20814 20835 20842 20845 20850 20862 20865 20867 20868
20871 20874 20875 20889 20901 20925 20928 20931 20932 20933 20934 20954

Continued on next page

Continued from previous page

20955 20956 20973 20979 20987 20988 21004 21005 21006 21015 21021 21044
21046 21060 21069 21081 21086 21087 21104 21108 21110 21115 21116 21117
21118 21119 21127 21134 21162 21204 21205 21206 21207 21208 21218 21219
21231 21233 21241 21246 21257 21261 21308 21327 21328 21334 21337 21339
21341 21356 21358 21360 21385 21396 21399 21402 21403 21405 21412 21413
21414 21429 21438 21478 21502 21509 21527 21539 21543 21544 21545 21549
21550 21551 21552 21553 21563 21565 21566 21567 21568 21569 21574 21575
21577 21581 21592 21610 21613 21614 21616 21618 21628 21629 21633 21640
21641 21643 21644 21645 21646 21660 21662 21664 21665 21666 21668 21669
21670 21671 21672 21673 21680 21681 21682 21683 21684 21694 21697 21701
21702 21703 21707 21715 21718 21719 21724 21739 21740 21741 21743 21744
21745 21746 21752 21753 21758 21769 21772 21780 21781 21783 21785 21795
21796 21804 21808 21809 21825 21826 21841 21842 21843 21845 21846 21847
21848 21856 21857 21858 21859 21875 21876 21877 21881 21882 21903 21904
21905 21914 21915 21924 21930 21969 21970 21971 21981 21985 21990 21999
22023 22039 22047 22048 22049 22059 22064 22065 22066 22071 22075 22088
22089 22097 22099 22100 22102 22104 22111 22113 22114 22129 22135 22137
22139 22144 22148 22184 22219 22224 22230 22232 22234 22236 22244 22247
22248 22250 22251 22252 22265 22266 22267 22268 22269 22270 22279 22286
22287 22296 22297 22298 22299 22321 22322 22323 22325 22326 22327 22329
22330 22331 22332 22333 22337 22344 22345 22347 22360 22364 22367 22369
22370 22381 22383 22391 22392 22393 22396 22397 22398 22399 22404 22455
22461 22463 22466 22467 22468 22470 22471 22472 22474 22475 22476 22479
22482 22483 22484 22485 22487 22488 22489 22490 22491 22492 22493 22494
22496 22497 22498 22499 22505 22517 22523 22538 22544 22557 22559 22563
22564 22565 22566 22567 22569 22571 22573 22575 22576 22577 22578 22616
22619 22637 22639 22643 22644 22652 22654 22655 22656 22681 22683 22689
22690 22691 22698 22705 22707 22708 22710 22711 22713 22714 22721 22740
22741 22746 22747 22750 22752 22753 22754 22755 22756 22764 22765 22766
22776 22779 22792 22793 22794 22796 22799 22807 22808 22824 22834 22845
22846 22847 22848 22849 22850 22851 22852 22853 22854 22855 22856 22857
22858 22860 22887 22889 22890 22904 22908 22917 22939 22963 22964 22965
22966 22967 22968 22970 22972 22975 22979 23005 23009 23012 23015 23021
23022 23023 23024 23027 23055 23056 23057 23058 23059 23061 23062 23072

Continued on next page

Continued from previous page

23073 23085 23086 23092 23116 23163 23190 23198 23212 23213 23214 23229
23239 23242 23243 23257 23272 23274 23275 23283 23299 23304 23321 23325
23326 23335 23342 23343 23354 23357 23358 23370 23389 23392 23394 23400
23418 23426 23428 23431 23435 23456 23457 23458 23459 23460 23461 23462
23463 23465 23474 23475 23538 23552 23562 23565 23566 23569 23591 23604
23605 23629 23631 23641 23645 23646 23648 23658 23659 23660 23664 23673
23675 23678 23679 23680 23686 23691 23693 23697 23706 23713 23718 23719
23721 23735 23736 23738 23739 23752 23755 23756 23757 23760 23767 23769
23770 23782 23785 23789 23790 23794 23797 23798 23799 23801 23811 23815
23819 23824 23831 23832 23834 23839 23840 23842 23847 23855 23860 23861
23876 23885 23886 23891 23892 23894 23900 23915 23920 23926 23927 23928
23930 23931 23932 23933 23935 23936 23940 23941 23963 23967 23981 23982
23983 23989 23990 23991 23993 23994 23995 23996 23998 23999 24000 24001
24012 24013 24014 24022 24024 24034 24049 24056 24060 24061 24062 24063
24064 24070 24071 24072 24073 24075 24135 24145 24147 24178 24190 24193
24194 24199 24200 24208 24214 24226 24228 24253 24256 24263 24274 24276
24278 24280 24286 24293 24299 24310 24313 24314 24316 24327 24343 24354
24356 24361 24363 24370 24378 24407 24414 24429 24430 24446 24447 24467
24470 24471 24482 24486 24487 24488 24489 24492 24493 24496 24510 24521
24534 24536 24538 24551 24559 24560 24561 24563 24564 24570 24586 24591
24596 24600 24605 24615 24619 24621 24628 24629 24642 24653 24658 24660
24661 24694 24698 24699 24708 24754 24760 24798 24801 24807 24809 24810
24833 24834 24836 24859 24861 24863 24870 24875 24878 24879 24880 24897
24898 24903 24904 24905 24913 24914 24915 24918 24933 24938 24941 24949
24970 24984 24985 24993 24994 24997 25000 25006 25012 25013 25019 25022
25023 25025 25026 25027 25036 25045 25046 25047 25054 25056 25060 25061
25062 25099 25115 25135 25139 25143 25154 25155 25181 25184 25190 25191
25204 25209 25213 25214 25218 25220 25242 25243 25244 25259 25268 25269
25274 25286 25308 25310 25311 25326 25327 25329 25337 25339 25344 25356
25358 25366 25369 25388 25393 25394 25398 25416 25419 25420 25425 25483
25513 25526 25527 25529 25534 25535 25542 25543 25547 25552 25553 25566
25612 25621 25625 25628 25636 25638 25639 25640 25643 25660 25667 25668
25680 25687 25688 25693 25696 25706 25708 25739 25743 25755 25762 25764
25767 25772 25781 25785 25786 25787 25800 25806 25807 25809 25811 25812

Continued on next page

Continued from previous page

25814 25816 25839 25841 25848 25854 25873 25877 25881 25882 25883 25890
25894 25911 25913 25939 25940 25941 25942 25943 25944 25945 25947 25948
25949 25951 25952 25953 25955 25963 25964 25983 25993 26005 26012 26014
26022 26029 26032 26033 26035 26039 26042 26043 26044 26045 26046 26048
26050 26060 26064 26066 26076 26078 26088 26092 26093 26104 26123 26127
26143 26147 26150 26151 26162 26163 26170 26186 26190 26200 26203 26204
26208 26211 26214 26215 26216 26217 26218 26219 26220 26222 26230 26231
26233 26234 26235 26237 26238 26244 26246 26251 26252 26257 26259 26261
26265 26266 26268 26269 26272 26273 26274 26278 26290 26291 26299 26302
26306 26307 26308 26312 26314 26329 26332 26343 26346 26354 26356 26358
26360 26361 26368 26391 26394 26397 26398 26399 26401 26408 26410 26417
26423 26429 26430 26435 26445 26448 26450 26458 26469 26471 26477 26479
26482 26488 26489 26490 26491 26514 26515 26516 26529 26530 26531 26541
26542 26544 26545 26549 26552 26571 26588 26603 26632 26633 26640 26641
26642 26643 26644 26645 26646 26647 26648 26649 26650 26651 26652 26657
26667 26668 26670 26672 26674 26687 26689 26690 26691 26692 26699 26701
26711 26713 26714 26717 26718 26729 26733 26734 26735 26740 26749 26750
26757 26767 26787 26788 26790 26795 26796 26799 26807 26810 26816 26822
26830 26833 26834 26835 26836 26838 26847 26861 26869 26870 26886 26890
26891 26892 26893 26895 26896 26898 26903 26904 26905 26907 26909 26910
26911 26912 26913 26914 26915 26916 26917 26918 26919 26920 26929 26932
26933 26934 26945 26948 26953 26954 26957 26958 26961 27001 27003 27004
27013 27021 27022 27023 27024 27038 27039 27040 27044 27045 27047 27051
27052 27063 27064 27068 27072 27076 27077 27078 27079 27080 27081 27095
27101 27110 27121 27128 27129 27130 27133 27141 27142 27144 27146 27147
27148 27162 27163 27164 27165 27166 27167 27168 27176 27178 27179 27181
27182 27183 27184 27185 27186 27188 27189 27190 27191 27196 27207 27215
27225 27228 27244 27245 27250 27252 27253 27254 27255 27256 27260 27278
27279 27280 27287 27289 27290 27292 27293 27294 27307 27308 27346 27347
27351 27356 27362 27369 27373 27380 27381 27382 27388 27396 27398 27399
27415 27421 27422 27444 27461 27471 27486 27506 27514 27572 27577 27604
27614 27628 27632 27636 27651 27657 27658 27669 27670 27681 27682 27685
27691 27716 27717 27719 27725 27739 27764 27765 27771 27783 27792 27796
27804 27811 27814 27817 27834 27835 27836 27837 27838 27839 27840 27842

Continued on next page

Continued from previous page

27866 27868 27871 27872 27877 27883 27889 27894 27904 27912 27913 27914
27926 27928 27929 27936 27940 27952 27958 27976 27999 28006 28008 28011
28012 28015 28016 28019 28023 28028 28036 28037 28038 28039 28040 28041
28042 28044 28045 28046 28047 28048 28052 28065 28068 28072 28073 28077
28088 28089 28093 28102 28115 28117 28118 28119 28122 28130 28131 28137
28139 28154 28156 28157 28158 28159 28160 28163 28166 28167 28186 28189
28193 28204 28213 28214 28220 28223 28226 28249 28268 28269 28270 28271
28275 28281 28282 28283 28284 28289 28292 28298 28304 28314 28316 28317
28318 28319 28320 28323 28324 28325 28339 28350 28355 28361 28374 28395
28396 28397 28398 28411 28414 28437 28442 28443 28453 28455 28457 28464
28465 28471 28472 28473 28474 28477 28481 28484 28486 28504 28511 28527
28528 28529 28530 28532 28541 28543 28544 28551 28553 28557 28565 28566
28569 28576 28579 28580 28581 28584 28591 28594 28596 28601 28602 28603
28604 28606 28608 28616 28618 28620 28626 28627 28629 28630 28631 28640
28646 28650 28656 28657 28664 28667 28669 28670 28676 28679 28681 28683
28685 28693 28695 28697 28698 28701 28710 28717 28720 28737 28741 28742
28743 28747 28748 28749 28765 28772 28773 28774 28775 28778 28780 28787
28801 28802 28814 28828 28851 28866 28869 28943 28950 28956 28963 28965
28970 28971 28985 28987 28993 28994 28995 28997 28998 28999 29042 29051
29070 29071 29078 29080 29081 29082 29083 29084 29085 29086 29087 29088
29118 29119 29120 29121 29124 29126 29127 29128 29129 29138 29144 29146
29147 29159 29181 29182 29188 29192 29197 29214 29215 29228 29229 29237
29254 29257 29258 29262 29266 29268 29270 29271 29285 29288 29289 29322
29323 29324 29325 29328 29334 29352 29356 29357 29358 29373 29376 29377
29378 29379 29407 29413 29415 29417 29418 29425 29434 29448 29473 29481
29499 29500 29504 29506 29507 29508 29509 29510 29517 29518 29519 29520
29521 29522 29523 29524 29527 29532 29535 29537 29538 29539 29540 29541
29544 29545 29549 29575 29596 29598 29599 29605 29606 29607 29608 29609
29613 29618 29625 29626 29627 29628 29629 29631 29632 29636 29637 29638
29639 29641 29659 29661 29666 29667 29670 29673 29680 29681 29688 29692
29693 29695 29696 29701 29702 29709 29713 29714 29716 29717 29718 29719
29720 29726 29728 29730 29740 29741 29751 29758 29761 29762 29763 29769
29777 29778 29779 29792 29820 29825 29826 29833 29835 29837 29844 29845
29854 29856 29857 29858 29864 29866 29868 29886 29887 29893 29906 29909

Continued on next page

Continued from previous page

29912 29914 29921 29928 29947 29953 29966 29967 29969 29977 29980 29985
29989 29990 29991 29993 29996 29999 30011 30019 30020 30021 30045 30048
30078 30079 30080 30083 30084 30085 30086 30088 30089 30090 30091 30092
30103 30106 30117 30126 30127 30131 30137 30139 30144 30154 30156 30157
30158 30160 30168 30173 30175 30176 30180 30182 30183 30184 30185 30187
30188 30189 30190 30191 30192 30193 30194 30195 30196 30197 30198 30199
30200 30201 30202 30203 30204 30205 30206 30207 30208 30209 30210 30211
30212 30214 30216 30217 30218 30219 30220 30221 30222 30223 30224 30225
30226 30227 30228 30229 30230 30231 30232 30233 30234 30235 30236 30237
30238 30239 30240 30241 30242 30243 30244 30245 30246 30248 30249 30250
30251 30253 30254 30255 30256 30257 30258 30259 30260 30261 30262 30263
30264 30265 30266 30267 30268 30269 30271 30273 30274 30292 30301 30307
30308 30310 30312 30313 30315 30317 30318 30320 30321 30322 30323 30324
30326 30327 30328 30329 30330 30331 30333 30335 30336 30337 30338 30340
30342 30343 30346 30347 30348 30350 30351 30352 30354 30356 30357 30358
30359 30360 30361 30362 30363 30364 30365 30367 30369 30370 30371 30372
30373 30374 30375 30377 30379 30380 30381 30382 30383 30385 30386 30389
30390 30391 30392 30393 30394 30395 30397 30398 30399 30400 30404 30405
30406 30409 30410 30411 30412 30413 30414 30415 30416 30417 30418 30419
30420 30421 30422 30423 30424 30426 30427 30428 30429 30430 30431 30432
30433 30434 30435 30436 30437 30441 30442 30443 30446 30447 30449 30450
30452 30453 30454 30455 30456 30458 30459 30460 30461 30464 30465 30466
30467 30468 30469 30470 30471 30472 30473 30474 30475 30478 30480 30483
30486 30487 30490 30491 30492 30493 30496 30497 30498 30501 30504 30505
30507 30508 30509 30510 30511 30512 30516 30527 30534 30535 30536 30537
30539 30543 30544 30545 30546 30547 30549 30550 30551 30553 30555 30556
30557 30558 30559 30560 30561 30563 30564 30565 30566 30567 30569 30571
30572 30573 30574 30575 30577 30578 30579 30584 30585 30587 30588 30589
30591 30592 30593 30595 30596 30597 30598 30599 30600 30601 30603 30605
30606 30607 30608 30609 30610 30611 30612 30613 30614 30615 30616 30617
30618 30620 30621 30622 30623 30624 30625 30628 30629 30630 30631 30633
30634 30635 30637 30638 30639 30640 30642 30643 30644 30645 30646 30647
30648 30649 30650 30651 30652 30653 30654 30655 30656 30657 30658 30659
30660 30661 30662 30663 30665 30666 30667 30668 30669 30670 30671 30672

Continued on next page

Continued from previous page

30673 30674 30675 30676 30678 30679 30680 30681 30683 30684 30685 30686
30687 30688 30690 30693 30694 30695 30696 30697 30698 30699 30700 30701
30702 30714 30715 30716 30718 30719 30720 30722 30725 30726 30727 30728
30730 30731 30732 30733 30735 30736 30737 30738 30739 30740 30741 30742
30743 30744 30747 30748 30749 30750 30751 30752 30753 30754 30755 30756
30757 30758 30759 30760 30762 30763 30765 30767 30768 30769 30770 30771
30772 30773 30774 30775 30777 30778 30779 30780 30781 30782 30783 30784
30785 30786 30787 30789 30790 30791 30792 30794 30795 30796 30797 30798
30799 30800 30801 30802 30806 30807 30808 30809 30810 30812 30815 30818
30819 30821 30824 30825 30826 30827 30828 30829 30830 30831 30832 30833
30835 30836 30851 30861 30864 30865 30868 30869 30871 30872 30874 30882
30885 30886 30887 30898 30913 30923 30942 30955 30956 30957 30958 30960
30963 30966 30974 30976 30977 30978 30989 30990 30991 30993 31005 31006
31007 31039 31045 31048 31049 31073 31084 31087 31089 31090 31093 31094
31099 31106 31116 31117 31118 31119 31132 31148 31152 31164 31173 31178
31179 31192 31198 31200 31204 31208 31210 31211 31212 31213 31214 31215
31217 31218 31219 31223 31225 31226 31229 31231 31232 31233 31234 31235
31236 31237 31238 31242 31243 31248 31250 31253 31257 31258 31265 31266
31269 31270 31271 31272 31273 31274 31275 31276 31279 31286 31288 31289
31296 31297 31299 31301 31304 31305 31306 31308 31309 31311 31326 31328
31329 31333 31334 31336 31341 31343 31347 31349 31353 31354 31355 31356
31368 31369 31370 31373 31375 31384 31385 31386 31390 31392 31393 31394
31397 31398 31404 31406 31409 31411 31412 31416 31420 31421 31422 31426
31427 31433 31434 31436 31437 31440 31441 31442 31445 31447 31451 31460
31461 31463 31464 31466 31467 31468 31471 31476 31479 31480 31481 31485
31486 31487 31488 31489 31490 31491 31493 31494 31495 31496 31497 31501
31506 31508 31510 31514 31515 31516 31520 31524 31526 31527 31528 31531
31532 31533 31534 31535 31536 31538 31539 31540 31543 31544 31545 31547
31548 31549 31551 31552 31556 31561 31563 31564 31567 31570 31572 31574
31584 31586 31587 31591 31592 31603 31606 31608 31610 31620 31621 31623
31632 31633 31634 31635 31636 31637 31638 31640 31643 31648 31650 31651
31652 31653 31655 31657 31659 31661 31663 31664 31667 31670 31671 31672
31680 31683 31684 31688 31690 31691 31694 31699 31700 31701 31704 31705
31711 31713 31716 31717 31720 31722 31725 31726 31729 31730 31731 31733

Continued on next page

Continued from previous page

31734 31735 31736 31737 31739 31744 31745 31753 31762 31764 31765 31768
31774 31776 31777 31780 31786 31790 31791 31794 31795 31802 31804 31807
31811 31812 31815 31816 31817 31825 31830 31834 31835 31840 31842 31845
31849 31850 31853 31854 31855 31856 31857 31858 31859 31860 31863 31864
31866 31874 31879 31884 31886 31887 31888 31890 31892 31894 31895 31898
31899 31907 31908 31914 31920 31923 31925 31927 31929 31931 31932 31935
31936 31937 31938 31939 31946 31947 31951 31952 31957 31959 31962 31963
31969 31972 31974 31976 31977 31978 31979 31980 31988 31990 31991 31993
31994 31995 31996 31997 31998 31999 32003 32005 32006 32007 32011 32012
32013 32014 32016 32019 32020 32021 32029 32032 32033 32034 32035 32049
32056 32059 32061 32062 32063 32065 32070 32071 32077 32079 32080 32081
32082 32083 32084 32085 32086 32087 32088 32089 32090 32091 32092 32094
32097 32098 32099 32102 32103 32104 32107 32108 32109 32113 32114 32115
32116 32117 32120 32126 32127 32128 32134 32135 32139 32142 32143 32148
32153 32154 32155 32156 32158 32161 32162 32163 32164 32167 32168 32171
32172 32173 32174 32175 32176 32177 32178 32179 32180 32181 32182 32186
32187 32192 32193 32197 32200 32208 32209 32210 32213 32214 32216 32218
32219 32220 32221 32223 32228 32229 32230 32234 32237 32238 32239 32240
32242 32244 32246 32248 32250 32252 32253 32256 32260 32261 32262 32265
32266 32270 32271 32272 32273 32274 32275 32281 32282 32286 32287 32289
32290 32291 32292 32294 32297 32298 32301 32302 32303 32304 32305 32307
32308 32311 32312 32313 32314 32315 32317 32321 32323 32324 32325 32327
32332 32333 32337 32339 32342 32343 32345 32346 32348 32349 32350 32352
32353 32354 32355 32357 32358 32361 32362 32363 32364 32367 32368 32370
32371 32372 32374 32375 32377 32378 32379 32380 32384 32386 32390 32392
32395 32396 32397 32398 32399 32400 32406 32407 32408 32412 32415 32420
32422 32423 32424 32425 32426 32429 32430 32431 32434 32435 32436 32440
32443 32444 32445 32446 32447 32448 32449 32450 32451 32452 32453 32454
32455 32456 32457 32458 32459 32460 32461 32462 32463 32464 32474 32476
32482 32484 32487 32490 32492 32493 32494 32495 32496 32497 32498 32499
32500 32501 32502 32504 32505 32507 32508 32509 32510 32511 32514 32515
32521 32522 32523 32524 32525 32526 32527 32528 32529 32530 32531 32532
32533 32534 32535 32538 32539 32540 32544 32545 32546 32547 32549 32550
32551 32552 32553 32554 32555 32556 32559 32560 32562 32563 32564 32565

Continued on next page

Continued from previous page

32566 32571 32572 32573 32574 32575 32576 32580 32582 32583 32584 32585
32588 32589 32592 32593 32602 32603 32607 32609 32610 32611 32613 32614
32615 32617 32618 32621 32622 32623 32624 32627 32630 32632 32633 32635
32636 32637 32638 32640 32644 32645 32646 32648 32649 32653 32654 32655
32657 32658 32660 32661 32662 32664 32670 32673 32674 32678 32684 32685
32686 32687 32688 32689 32691 32695 32697 32699 32704 32707 32710 32711
32712 32713 32715 32717 32722 32723 32724 32725 32727 32728 32729 32731
32732 32733 32735 32736 32739 32740 32742 32744 32745 32748 32750 32751
32755 32757 32758 32759 32760 32762 32763 32766 32768 32769 32772 32775
32777 32778 32780 32781 32782 32783 32784 32785 32788 32789 32790 32796
32797 32800 32804 32807 32808 32809 32810 32818 32823 32824 32825 32826
32828 32834 32838 32839 32840 32841 32842 32843 32845 32848 32851 32852
32853 32856 32857 32859 32860 32861 32862 32864 32865 32870 32871 32872
32873 32874 32875 32877 32878 32879 32881 32882 32884 32890 32892 32893
32894 32895 32897 32903 32906 32909 32911 32912 32915 32917 32918 32919
32920 32921 32923 32924 32925 32928 32929 32931 32932 32936 32937 32939
32942 32943 32944 32947 32950 32957 32960 32961 32962 32963 32966 32967
32968 32969 32970 32972 32973 32974 32982 32984 32994 32996 32997 33000
33002 33003 33005 33006 33008 33015 33017 33018 33019 33022 33024 33025
33026 33027 33029 33034 33036 33037 33038 33040 33041 33043 33044 33046
33047 33051 33061 33062 33069 33070 33073 33075 33079 33080 33085 33086
33088 33090 33091 33095 33098 33099 33100 33102 33103 33104 33105 33106
33113 33114 33118 33124 33126 33127 33128 33131 33134 33135 33139 33140
33141 33145 33147 33150 33151 33158 33163 33165 33167 33168 33170 33173
33175 33177 33181 33182 33190 33193 33199 33201 33211 33228 33230 33233
33239 33246 33249 33250 33259 33260 33261 33265 33266 33267 33281 33282
33283 33284 33285 33286 33293 33297 33302 33303 33308 33314 33315 33316
33322 33323 33324 33335 33365 33391 33406 33407 33408 33415 33421 33424
33466 33472 33476 33477 33478 33479 33480 33481 33482 33483 33484 33485
33486 33487 33488 33489 33490 33491 33493 33495 33505 33506 33507 33515
33524 33526 33540 33548 33549 33550 33557 33583 33586 33646 33647 33650
33651 33652 33653 33654 33656 33657 33658 33660 33661 33662 33664 33666
33667 33669 33672 33674 33675 33676 33678 33679 33680 33681 33682 33688
33700 33724 33725 33726 33727 33728 33729 33736 33737 33740 33741 33765

Continued on next page

Continued from previous page

33766 33767 33769 33770 33771 33772 33773 33774 33793 33804 33806 33808
33813 33814 33826 33834 33835 33860 33889 33898 33905 33906 33909 33913
33914 33938 33943 33952 33956 33961 33963 33965 33966 33968 33972 33976
33985 33986 33993 33994 34005 34017 34018 34027 34028 34035 34038 34051
34054 34069 34083 34088 34089 34090 34100 34101 34102 34107 34115 34116
34168 34169 34170 34171 34186 34187 34188 34189 34192 34195 34202 34203
34217 34220 34222 34223 34239 34242 34244 34245 34246 34254 34299 34310
34315 34316 34319 34320 34321 34322 34348 34359 34360 34362 34363 34364
34368 34379 34381 34382 34397 34412 34413 34416 34417 34428 34429 34434
34444 34445 34447 34449 34455 34459 34462 34470 34480 34483 34484 34493
34494 34495 34501 34502 34503 34504 34506 34519 34520 34521 34527 34539
34544 34545 34568 34569 34570 34571 34572 34577 34601 34605 34642 34646
34647 34649 34650 34657 34659 34660 34661 34664 34699 34702 34706 34716
34718 34719 34721 34743 34745 34746 34747 34748 34750 34751 34752 34753
34755 34757 34777 34790 34823 34827 34829 34832 34833 34834 34838 34855
34856 34869 34947 34948

G. Bond Order Assignment Validation Molecules

Table G.1: List of the names of molecules in the MMFF94 validation set. Molecules are sourced from `http://www.ccl.net/cca/data/MMFF94/`.

AGLYSL01	AMHTAR01	AMPTRB10	AN05A	AN06A	AN08A	AN11A	AN12A	AR14A
ARGIND11	BAOXLM01	BBSPRT10	BEVJER10	BEWCUB	BEWKUJ04	BIHKEI01	BIPDEJ02	BIPJUF10
BIPYCL01	BITNAT10	BIYBIU10	BODKOU	BODKOU	BSALAP01	BUPSLB10	BUPSLD10	BUYTIY10
BUYTOE10	BUYXEY10	BYITOT02	CA04A	CABWEH10	CAFORM07	CAGREH10	CALXES20	CAMALD03
CE05A	CEFMEN	CETROI01	CEWCUC10	CEWVIJ10	CEWYIM30	CIHWUL10	CIJXOI10	CIKSEU10
CILBII	CILDOQ	CILWUP11	CIMRUL10	CINVIE	CIPVOM	CIPYAB10	CISMOG	CISPOJ
CITDIS	CITNOI10	CITPEA10	CITSED10	CIVCEP02	CIVLAU02	CIXWAH	CIYNUT	CIZFIA
CIZWUD	CIZYEP	CIZZUG	CO01A	CO08A	COBKIN01	COCXUN	COGDEH	COGYAY
COHKOZ	COJFIQ	COKDEL	COKROJ	COLZUY	COMDIR	COMKAQ	COMWOQ	COMWUW
CONBAI	CONFAM	CONLIA	CORDOC	CORWUB10	COSFAR	COSSEI	COSWIQ	COTMON
COTPEG	COTRIM	COVHUQ	COVMAB	COVXIU	COWTIR	COXBAS	COXZEU	COYMOS
COYNAF	COYVIV	CUBTUO	CUCDAF	CUCHOX	CUCHUD	CUDJAM	CUDPAS	CUDPOG
CUDREY	CUFFAK	CUGBEL	CUGGOA	CUGLOF	CUJYUB10	CULGEV10	CULHIA10	CULVEK
CUNVAI	CUNVEM	CURZIY	CUVFOO	CUVGAB	CUVJOS	CUYRAP	CYANAM01	CYGUAN01
DABHAP	DABLIB	DACSAB	DACYIP	DADDAN	DADLAV	DADLEZ	DAFKIE	DAFPUV
DAGTUA	DAHBAP	DAJXER	DAKBAS	DAKCEX	DAKDOI	DAPSUO03	DARDEF	DARXID
DARZEB	DAVWEC	DAVXED	DAWXII	DAWYUV	DAYWEF	DAZVEF	DEBMOM01	DECJAW
DECKUR	DECRIM	DEDCIY	DEFGIE	DEFLEF	DEFPUZ	DEFTUD	DEFVAL	DEFYUI
DEGLUW	DEGRIQ	DEKRUG	DEMBIG	DEPKEO	DERZUV	DESWUT	DESYOP	DEWHOC
DEXCIS	DEXGIW	DEZDUH	DEZNIF	DEZXEL	DHOADS01	DICKIJ	DICPUA	DICRAI
DICYOD	DIDYOE	DIFSIU	DIGCOL	DIGCUR	DIGLEK	DIHTET	DIKGAF	DIKGEJ
DIKWID	DIKYUR	DILCOQ	DIMYIH10	DIPDAH10	DIPDIP10	DIRMIA	DISHES	DISJOE
DITRAZ	DITYAG10	DIVJUN	DIVTUX	DIVVEJ	DIVWEK	DIWCOB	DIXJEZ	DIYDIY
DIYPOQ	DIYPUW	DIZPUX	DMEOXA01	DOCCIH	DOCFIK	DOCWUN	DODNOZ	DODNUF
DOJPAT	DONFOB	DOSNOO	DOTNIJ	DOTVEN	DOTWOY	DOWDEY	DOXXAP	DOXZOF
DOZFON	DOZNIP	DUBNET	DUDMUK	DUGMUN	DUGWIL01	DUJHEV	DUJMEA	DUKVAG
DUKWUB	DULTIN	DUMHIC	DUMPAC	DUPHEB	DUPTAJ	DURDID	DUTHIJ	DUVHUX10
DUVXIB	DUWGAD	DUWKUB	DUWRIW	DUXTIZ	DUXWUO	DUXXAV	DUYNOA	DUYPES
DUYRAQ	ERULE_01	ERULE_02	ERULE_04	ERULE_05	ERULE_06	ERULE_07	ERULE_08	FACMIF
FACREG	FACYAJ	FADMIG	FADVEL	FADVUB	FAGBUK	FAGCOF	FAGLII	FAGVEO
FAGZOC	FAHPUZ	FAHSUC	FAHYUI	FAHZET	FAJWIW	FAMHAC	FAMYUN	FAPLUD
FARMAM	FARSOG	FARWEA	FASGUB	FASJIS	FATLIV	FAXFUF10	FAXVAB	FAXVEF
FAXVIJ	FAZBAJ	FBATNB	FECXEQ	FEGSEP	FEHDAX	FEJJEJ	FEJKIO	FELYIE
FELYUQ	FEMGEJ	FENCOQ	FENHAH	FENJIR	FENJOX	FENJUD	FENNUH	FENYIG
FEPWAY	FEPWOM	FESCAH	FESMIZ	FETRUR	FETWOQ	FEVNUP	FEYLUQ	FEZPOP
FEZRUX	FIBLIL	FICDOK	FIFGUW	FIGYID	FIHXID	FIKJAK	FIKZOO10	FILGEM
FILNOD	FINBIN	FINPEX	FITGIY	FITSEG	FITTIL	FIVNUT	FIVRAD	FIXPIL
FIYBIY	FIZGEA	FIZGOK	FIZJED	FOBJUB01	FODTUN	FOGVIG01	FOHXEF	FOHYAC
FOJBEL	FOJPAV	FONCOA	FORHEZ	FORJIF	FORJUR	FORTAH	FOSDIA	FOVHUT
FOVJIJ	FOVRAJ	FOVRUD	FOWBEY	FOWPOW	FOWVES	FOWZAS	FOYMAH	FOYNUC
FUCMUL	FUCTIG01	FUCWIJ	FUCWOP	FUDPOJ	FUDXUX	FUFDIT	FUGWIN	FUHFAP
FUHSEG	FULRAF	FUNSIQ	FUNXOB	FUPJUV	FUPKIK	FUPKOQ	FUPTOZ	FUPZEV
FUSPEO	FUTCEC	FUVDOP	FUVMUE	FUVNEP	FUVXOJ	FUWMOZ	FUWTUM	FUXXAX
FUXZED	GADHEY	GAFNUW	GAHPIO	GAJTEQ	GAKGOO	GAKNEL	GAKNIP	GAKPEN
GAKTAN	GANHUY	GAPMEP	GAVKOD	GAVMEV	GAWWOQ	GEHBOK	GEHPUE	GEHXEW
GEJYOJ	GEKXEZ	GEMCEG	GEMCOQ	GEMDAD	GERCUB	GESCIQ	GESNIB	GESSUS
GETFIU	GETFOA	GETJOE	GEWTAD	GEXGIZ	GEYWOW	GICTIV01	GIDJUY	GIDMEL

Continued on next page

Continued from previous page

GIDTIW	GIFRAO	GIGCEE	GIGMUE	GIHZEC	GIJMOB01	GIKJIT	GIKNOD	GIKTUP
GIMJIV	GINMUL	GIPHES	GIRDOA01	GOHVUU	GOJCIR	GOJKIZ	GUANCH01	H3OPW1
HL08A	HL11A	HL13A	HYTPRD01	IM02A	ISTZCN10	JABGAU	JADLIJ	JADXER
JAHKOS	JAHTOB	JAHYEW	JAKGUX	JAKJOU	JALSOE	JAMREU	JANDOR	JANMAM
JAPFAH	JATBIP	JATCOW	JATLOF	JATMEW	JAVGAO	JAWJIA	JAWMAV	JAWVEI
JAWZEM	JAZGOG	JAZVIP10	JAZZOZ10	JEBFEB01	JECVES	JECVUI	JECYIZ	JEFRAN
JEHCUU01	JEHXOJ	JELKUG	JELREX	JELRIB	JEMHIS	JEMWUT	JETJUN	JEVXIR
JEWFAS	JEWPIK	JEXREJ	JEYBUK	JIDHIN	JIFYUS	JIGCIL	JIGRAS	JIHVEB
JILWUW	JINDAL	JINDOZ	JIRJID	JISZAM	JITMII	JIWKOP	JIXBAT	JIYJAC
JIYREO	JIYTOA	KAFXIY	KAGBOJ	KAKGOS	KAMCUW	KAMJAJ	KANWEB	KANZOO
KAPCUZ	KARYAD	KASBAH	KASBOV	KATNAU	KAVFUI	KAVTEG	KECSIU	KECSUG
KEDYAT	KEFJEK	KEJFOU	KEMFAJ	KENHOA	KEPKIZ	KESNEB	KEWJIF	KHDFRM11
KIBDII	KIBFAC	KICCUU	KICGAE	KICLAJ	KIGKIU	KIKVUV	KIMLEX01	KINKUN
KINTUW	KINWEJ	KINWIN	KIRCAP	KIRCOD	KITREK	KIYGAA	KOBXOO	KOBYOP
KOBZEG	KOCWUU	KOFKIZ	KOFMEX	KOFNIC	KOHVEI	KOHVIM	KOJGOF	KOJKID
KOJZOY	KOKMIG	KOLCUJ	MAPMIP03	MENBZS01	METBZC10	NAESCB01	NC10A	NC13A
NH10A	NH20A	NH22A	NH23A	NHOXAL06	NO03A	NX02A	OC02A	OH10A
PHOSLA10	PHOSLB10	PIMTAZ01	PO02A	PO05A	PR01A	PR02A	PR03A	PR04A
QUICNA01	SABNOY	SACXAV	SADXAW	SAFFOU	SAFFUA	SAFKAL	SAHSOJ	SAHSUP
SAKGUG	SALVEG	SAMFUH	SAMXUZ	SANKEX10	SAVDOI	SAWKEG10	SEBPEU01	SECDAF
SEFRAW	SEFYIL	SEGFIT	SEGLAR	SEGNEX	SEHBEM	SEJDAM	SEKKIC	SEKPED
SEKPIH	SEMDIX	SEMXOX	SETHAA	SETLIM	SEYVUN	SEYWUO	SEZMEP	SICNUN
SICPEZ	SICSEC	SIDRUS	SINMIL	SIYLOB	SIZJIU	SIZWUT	SLFNMB04	SO07A
SO12A	SO15A	SO16A	SO18A	SOGVOZ	SOHXOC	SOJNEK	SOMKIO	SONZIE
SOPZEC	SORBIK	SR05A	SR07A	SURDOX02	TACGIN	TACLEO	TAFKIU	TAFXIH
TAFZIJ	TAGVIG	TAHMOE	TAJPUP	TAJSUS	TAJVUV	TAJWAC	TAKHES	TAMMAV
TANHAR	TAPJUP	TAPSAE	TCYMPH02	TMTCHD01	VABLIT	VABROF	VACRUM	VAJFAN
VALTEH	VALWOU	VAPZOB10	VASDOI	VATKAC	VAWDUS	VAWMOV	VAWWAR	VAYKUB
VAZHUZ	VECSAX	VEDTED	VEHZOX	VEJWOW	VEKMON	VENYUI	VETWAS	VEVDIJ
VEWZOM	VEXKOY	VEXMOA	VEYWAX	VEZBUX	VICGAP	VICGET	VICKIB	VICPOM
VIDKUO	VIFFEV	VIGPEG	VIGTUA	VIHHID	VIKVIU	VIKYAP	VIMHII	VIPXAT
VIRBON	VIWCOT	VIXRID	VIXXOP	VIYPAU	VOBLAZ	VOBWOY	VOFBOH	VOFCAU
VOJGEG	VOJJIN	VUWXUG	VUXGOK	VUXPUZ	VUXREL	ZZZIZA01	ZZZMVU10	

Table G.2: List of the IDs for molecules in the KEGG Drug Database validation set. Molecules ending in *_x* indicate they are the *x*'th connected component in the source file. Molecules are sourced from `ftp://ftp.genome.jp/pub/kegg/medicus/drug`.

D00002	D00005	D00006_1	D00007	D00008	D00009	D00010	D00012	D00013
D00014	D00015	D00016	D00018	D00019	D00020	D00021	D00022	D00023
D00025	D00026	D00027	D00028	D00029	D00030	D00031	D00032	D00033
D00034	D00035	D00036	D00037	D00038	D00039	D00040	D00041	D00043
D00044	D00045	D00047	D00048	D00049	D00050	D00051	D00052	D00054
D00055	D00057	D00058	D00059	D00060	D00061	D00062	D00063	D00064
D00065	D00066	D00067	D00068	D00070	D00073	D00075	D00076	D00077
D00078	D00079	D00080	D00081	D00082_1	D00084	D00086	D00087	D00088
D00089	D00091	D00093	D00094	D00095	D00097	D00098	D00099	D00100
D00101_1	D00101_2	D00103	D00104	D00105	D00106	D00107_2	D00109	D00110
D00111	D00112	D00113	D00114	D00117	D00118	D00119	D00120	D00121
D00122_2	D00124	D00125	D00126	D00127	D00129	D00130	D00131	D00132
D00133	D00134	D00135_2	D00136	D00137	D00138	D00139	D00140	D00141
D00142	D00143	D00144	D00145	D00148	D00149	D00151	D00152	D00153
D00154	D00155	D00156	D00157	D00158_1	D00159	D00160	D00161_2	D00162
D00163	D00164	D00165	D00167	D00168	D00169_1	D00170	D00171	D00174
D00175	D00176	D00177_1	D00178_2	D00179	D00180	D00181_1	D00182	D00183
D00184	D00185	D00186	D00187	D00188	D00189	D00190_1	D00192	D00193
D00194	D00195_2	D00196	D00197	D00198	D00199	D00200_1	D00201	D00202
D00203	D00204	D00205_1	D00206_1	D00208	D00209	D00210	D00211	D00212
D00213_2	D00214	D00215	D00216	D00217	D00218	D00219	D00220	D00221
D00222	D00223	D00224	D00225	D00226_2	D00227_1	D00227_2	D00228	D00229_1
D00230_1	D00231	D00232_1	D00233	D00234	D00235	D00236	D00238	D00240
D00241	D00242	D00243	D00244	D00245	D00246	D00247	D00248	D00249
D00250	D00251	D00252	D00254	D00255	D00256	D00257	D00258	D00259
D00260	D00261_1	D00262	D00263	D00264	D00265	D00266	D00267	D00268

Continued on next page

Continued from previous page

D00269	D00270	D00271	D00272	D00273_1	D00274	D00276	D00277	D00278
D00279	D00280	D00281	D00282	D00283	D00284	D00286	D00287_1	D00288
D00289	D00290_1	D00291	D00293	D00294	D00295	D00296	D00297	D00298
D00299	D00300	D00301_2	D00302	D00303	D00304_1	D00304_2	D00305_1	D00306
D00307_1	D00308	D00309	D00310	D00311	D00312_1	D00313	D00314_1	D00315
D00316	D00317	D00319	D00320	D00321	D00322	D00323	D00324_2	D00325
D00326	D00327	D00328	D00329	D00330	D00331	D00332	D00333	D00334
D00335	D00336	D00337	D00338	D00340	D00341	D00343	D00345	D00346
D00347	D00349	D00350	D00351	D00352	D00353	D00354	D00355	D00356
D00357_1	D00358	D00359	D00360	D00362_1	D00363	D00364	D00365	D00367
D00368	D00369	D00370	D00372	D00373	D00374	D00375	D00376	D00377
D00378_2	D00379	D00380	D00381	D00382	D00383	D00384	D00385	D00386
D00387	D00388_1	D00389	D00390	D00391	D00392	D00393	D00394	D00395
D00396	D00397	D00398_2	D00400	D00401	D00402	D00403	D00404	D00405_1
D00407	D00408	D00409	D00410	D00411	D00412	D00413	D00414	D00415
D00416	D00418	D00419	D00420	D00421	D00422	D00423	D00424	D00425
D00426	D00427	D00428	D00429	D00430	D00431	D00432	D00433_1	D00434
D00435	D00436	D00437	D00438	D00439	D00440	D00441	D00442	D00443
D00444	D00445	D00448	D00449	D00450	D00451	D00452	D00453	D00454
D00455	D00456	D00457	D00458_1	D00458_2	D00459_1	D00460	D00461	D00462
D00463	D00464	D00465	D00466_1	D00467	D00468	D00469_1	D00470	D00471
D00472	D00473_1	D00474	D00475	D00476	D00477_1	D00478_1	D00479_2	D00480_1
D00481_2	D00482_2	D00483_1	D00484_1	D00486_1	D00486_2	D00487_1	D00488	D00489_1
D00489_2	D00490	D00491	D00492_1	D00495	D00496	D00497_1	D00498	D00499
D00500_1	D00501	D00502_1	D00502_2	D00503	D00504	D00505_1	D00506	D00507_1
D00508	D00509_1	D00510	D00511_2	D00512	D00513	D00514	D00515	D00516
D00517	D00518	D00519	D00520_1	D00521	D00522	D00523	D00524_1	D00525
D00526_3	D00527_1	D00528	D00529_1	D00530	D00531	D00532	D00533	D00534
D00535	D00536	D00537	D00539	D00540_1	D00541	D00542	D00543	D00544
D00545	D00546	D00547	D00548	D00549	D00550	D00551	D00552	D00553
D00554	D00555	D00556	D00557	D00558_2	D00559_1	D00560	D00561	D00562
D00563	D00564_1	D00565	D00566_1	D00567	D00568	D00569	D00570	D00571_1
D00572	D00573_2	D00574	D00575	D00576	D00577	D00578	D00579_4	D00580
D00583	D00584	D00585	D00586	D00587	D00589_1	D00590	D00591	D00593
D00594	D00595	D00596_1	D00597_1	D00598_1	D00599_2	D00600_1	D00601_2	D00602_2
D00605_1	D00606_1	D00607_1	D00608_1	D00609_1	D00610_1	D00611_1	D00612_1	D00612_2
D00613_2	D00615_1	D00615_2	D00616_1	D00617_1	D00618	D00619_1	D00620_1	D00621_1
D00622_1	D00623_1	D00624_1	D00624_2	D00625	D00626	D00627	D00628	D00629
D00630	D00631_1	D00632_1	D00633_1	D00634_2	D00635_1	D00637_1	D00638_1	D00639_1
D00640_1	D00641	D00642_1	D00642_2	D00643_2	D00644_2	D00645_1	D00645_2	D00646
D00647	D00648_2	D00649_1	D00650	D00651	D00652	D00653_1	D00654	D00656
D00657	D00658	D00659_2	D00661_1	D00662_2	D00663_1	D00664_1	D00665_2	D00666_2
D00667_1	D00670_2	D00671_1	D00672_1	D00674_1	D00675_1	D00677_2	D00679_1	D00680_1
D00681_1	D00682_2	D00683_1	D00684_2	D00685_1	D00686_2	D00687_1	D00687_2	D00688_1
D00689	D00690	D00691	D00694_1	D00697	D00698	D00699_1	D00700	D00701_1
D00702_1	D00703	D00704	D00705	D00706_1	D00707	D00708	D00709	D00711_1
D00712_1	D00713_1	D00714_1	D00715_2	D00716_2	D00717_2	D00718_1	D00720_2	D00721_2
D00723_2	D00724_2	D00725_2	D00726	D00727_1	D00729_2	D00730	D00732_1	D00733
D00735_1	D00736_1	D00737_1	D00738_1	D00739_1	D00740_1	D00743_2	D00744	D00749
D00751_1	D00752	D00753	D00754	D00755_1	D00756_1	D00756_2	D00757_2	D00758_1
D00758_2	D00760_1	D00761_3	D00762	D00763_1	D00764_1	D00765_1	D00766_1	D00767_1
D00768	D00769_1	D00770	D00771	D00772_1	D00773	D00774_2	D00775	D00776_1
D00777_1	D00778_2	D00779	D00780_1	D00781	D00782_2	D00784_1	D00785_1	D00786
D00787_2	D00788_1	D00790	D00791_1	D00792	D00793	D00794_2	D00795_1	D00796_1
D00797_2	D00799_1	D00801_1	D00802	D00803_1	D00805	D00806_1	D00807	D00808_1
D00809_1	D00810_2	D00811_1	D00812_1	D00813_1	D00814_2	D00815_1	D00816_1	D00817_1
D00818_2	D00819_1	D00820_2	D00821_2	D00822_1	D00824_2	D00825_1	D00826_1	D00828
D00829_1	D00830	D00831_1	D00833	D00834_1	D00834_2	D00835_3	D00836_2	D00837_1
D00838	D00839_2	D00840_2	D00841_1	D00842_3	D00843_2	D00844_2	D00845_2	D00846_2
D00847_1	D00848_3	D00849_2	D00850_1	D00851_1	D00851_2	D00852	D00853	D00855_1
D00856_2	D00857_1	D00858_1	D00859	D00862	D00863_2	D00865_1	D00866_1	D00867_2
D00868_1	D00869_1	D00870	D00871_1	D00873_1	D00874_2	D00875_1	D00877	D00878_1
D00879	D00880_1	D00881_1	D00883_1	D00884	D00885_1	D00886_2	D00887_1	D00888
D00889_1	D00890	D00891_2	D00892_2	D00893_1	D00894	D00895_1	D00896	D00897_1
D00898	D00899_2	D00900_2	D00901_1	D00902	D00903_2	D00905_1	D00907_1	D00908_1
D00909_1	D00910	D00911_1	D00912_1	D00913_1	D00914	D00915_1	D00916_1	D00917
D00918_1	D00919_1	D00920	D00921_1	D00922	D00923_1	D00924_1	D00925_1	D00927_1

Continued on next page

Continued from previous page

D00928_1	D00929_1	D00930	D00931_1	D00933_1	D00934_1	D00935_1	D00936_2	D00937_1
D00938_1	D00939_3	D00941_4	D00942_2	D00944_1	D00945_1	D00946	D00947	D00948_1
D00949	D00950	D00951	D00952	D00953	D00955	D00956	D00957	D00958
D00959	D00960	D00961	D00962_2	D00963	D00964	D00965	D00966_2	D00967_2
D00968_3	D00969	D00970_1	D00971_1	D00972_1	D00973	D00974	D00976	D00977_1
D00978_1	D00979	D00980	D00981_1	D00982	D00983	D00984	D00985	D00986
D00987	D00988_2	D00989_2	D00990_1	D00993	D00994_2	D00995_1	D00997_2	D00998_2
D00999_1	D01000_1	D01001_3	D01002_2	D01003_1	D01004_2	D01005_2	D01006_2	D01007_2
D01008_2	D01009	D01017_2	D01018_1	D01019_2	D01020_2	D01022_2	D01024_2	D01025_2
D01026_1	D01028_1	D01029_1	D01031	D01034	D01035	D01036	D01037	D01038
D01039	D01040	D01041	D01042	D01043	D01044	D01045_1	D01046_1	D01047
D01048	D01049	D01050	D01051	D01052	D01053_1	D01054	D01055	D01056
D01057	D01058_1	D01059_1	D01061_3	D01062_3	D01063_2	D01064	D01065	D01066_1
D01067	D01068_2	D01069_1	D01070	D01071	D01072	D01073	D01074	D01075
D01076	D01077_2	D01078	D01079	D01080	D01081_1	D01082_1	D01085	D01086
D01087	D01088_1	D01090	D01091_1	D01092	D01093_1	D01094	D01095	D01097_1
D01098	D01100	D01101	D01102	D01103_2	D01104_1	D01105_2	D01106_1	D01107_1
D01109	D01110_1	D01111	D01112	D01113	D01115	D01116	D01117	D01118_1
D01119_1	D01120	D01121	D01122	D01123	D01124_2	D01125	D01126_1	D01127_1
D01127_2	D01128	D01129	D01130_2	D01131	D01132	D01133	D01134	D01135
D01136	D01137_1	D01138	D01139_1	D01140_1	D01141	D01142	D01143_1	D01144
D01145	D01146	D01147	D01149	D01150	D01151	D01152	D01153	D01155_2
D01156	D01157_4	D01158	D01159	D01161	D01162_1	D01162_2	D01163_1	D01164
D01165_1	D01171_1	D01172_1	D01173	D01174_1	D01175_1	D01176	D01177_2	D01178_1
D01179_1	D01180	D01182_1	D01183	D01184	D01185	D01186_2	D01187_2	D01188
D01189	D01190	D01191	D01192_2	D01193	D01194_1	D01195	D01196_1	D01197_2
D01198	D01199	D01200	D01201_1	D01202	D01203_1	D01204	D01206	D01207_1
D01209_1	D01210	D01211_1	D01212	D01213	D01214	D01215_1	D01216	D01217
D01218	D01219	D01220	D01221	D01223	D01224_2	D01225_2	D01226	D01227_1
D01228_3	D01229	D01230	D01231_1	D01232_1	D01233_2	D01234	D01235	D01236_2
D01237	D01238_1	D01239_1	D01240	D01241	D01245	D01246	D01247	D01248_1
D01249_1	D01250_1	D01252	D01253	D01254	D01255	D01256	D01257	D01259
D01260_1	D01262_1	D01263_1	D01264_1	D01265	D01266	D01267	D01268	D01269_2
D01270	D01271	D01272	D01273	D01274	D01275_1	D01276_1	D01277	D01278
D01279	D01281	D01282_1	D01282_2	D01283_1	D01284	D01285_1	D01286	D01287_1
D01288_1	D01289	D01290	D01291	D01292	D01293	D01294	D01295	D01296_1
D01297_2	D01299	D01300	D01301	D01302_1	D01303_1	D01304_1	D01305	D01306
D01307_2	D01308	D01309	D01310_2	D01314_1	D01315_2	D01316	D01317_3	D01318_2
D01319	D01320	D01321	D01322	D01323	D01324	D01325	D01326_2	D01327
D01328	D01329	D01330	D01331	D01332_2	D01333	D01334	D01335	D01336_2
D01337_2	D01338	D01339	D01340_2	D01341_1	D01342_1	D01343_1	D01344	D01346
D01347_2	D01348	D01350_2	D01351	D01353	D01354	D01355_1	D01356_1	D01357
D01358_2	D01359	D01360_2	D01361	D01362_1	D01364_1	D01364_2	D01365	D01366
D01367	D01368	D01369_1	D01370	D01371	D01372	D01373_2	D01374	D01375
D01377_1	D01378	D01379	D01380	D01381	D01382	D01383_2	D01384	D01385
D01387	D01389_1	D01390_2	D01391	D01392	D01394	D01395	D01396	D01397
D01400	D01401_2	D01402	D01403	D01404	D01405_2	D01406_1	D01407	D01409
D01410_2	D01411_2	D01412_1	D01413	D01414	D01415_1	D01417_2	D01418	D01419
D01420	D01421_1	D01422	D01423_1	D01425	D01426	D01427_2	D01428_2	D01429
D01430	D01431	D01432_3	D01433	D01434	D01435_1	D01436	D01437	D01438
D01439_1	D01440_1	D01441_1	D01442	D01443	D01444_2	D01445_1	D01447_1	D01449_2
D01451_1	D01452	D01454_2	D01455_2	D01456	D01457_2	D01458	D01459_2	D01460_2
D01461	D01462_1	D01463_2	D01464	D01465_2	D01466	D01467_1	D01468	D01469_1
D01470	D01472_1	D01473_1	D01475	D01476	D01477_1	D01478	D01479_1	D01480_2
D01481_1	D01482_1	D01483_1	D01484_1	D01485	D01486_2	D01487_2	D01488_1	D01489_1
D01490_2	D01491_1	D01492_2	D01493_1	D01495_1	D01495_2	D01496	D01497_1	D01497_2
D01498	D01499_1	D01500_2	D01501_1	D01504_2	D01505_2	D01506_1	D01507_1	D01508
D01509_1	D01510_2	D01511	D01512_1	D01513	D01514	D01515_1	D01516	D01517_1
D01518	D01519	D01521_2	D01522_1	D01523_2	D01524_1	D01525_1	D01526	D01527
D01528	D01529	D01530	D01531_1	D01532_1	D01533	D01534	D01535	D01536
D01537_1	D01538_2	D01539	D01541	D01542	D01543_1	D01544_1	D01545	D01546_1
D01547	D01548_1	D01549_2	D01550	D01551_1	D01552	D01553_1	D01554_1	D01556_1
D01556_2	D01557	D01558	D01559_1	D01560_2	D01562	D01563_1	D01564_1	D01565
D01566_1	D01567	D01568_2	D01569_1	D01569_2	D01571_1	D01573_1	D01574_2	D01575_3
D01576_1	D01577	D01578	D01579	D01580	D01581	D01582	D01583	D01584
D01585	D01586_1	D01587	D01588_1	D01589_1	D01590	D01591	D01592_2	D01593
D01594	D01597	D01598_1	D01599	D01600_2	D01601	D01602	D01603_1	D01604_2

Continued on next page

Continued from previous page

D01605	D01606_2	D01607	D01608_2	D01611_1	D01612_3	D01613_2	D01614	D01615
D01616_3	D01617	D01619	D01620_2	D01621	D01622_1	D01623_3	D01624_1	D01625_2
D01628	D01629_1	D01630	D01631	D01632	D01633	D01634	D01635_2	D01636
D01638	D01639	D01640_1	D01640_2	D01641	D01643_1	D01645_1	D01646_1	D01647
D01649_1	D01650_2	D01651_2	D01652	D01653_1	D01654_2	D01655	D01656_2	D01657
D01658_1	D01659	D01660_1	D01661	D01662	D01663	D01664_1	D01665	D01666
D01667_2	D01668_3	D01669	D01670_1	D01671_1	D01672_1	D01673_1	D01674	D01675
D01676	D01677	D01678	D01680_1	D01681_2	D01682_1	D01683_1	D01684_1	D01685_1
D01686	D01687_1	D01688	D01689	D01690	D01691	D01692_1	D01693	D01694
D01695	D01696_1	D01697_1	D01698	D01699_2	D01700_1	D01701_1	D01702_1	D01703
D01704	D01705_1	D01706_2	D01707_1	D01707_3	D01708	D01709_3	D01710	D01713_1
D01715	D01716	D01717_1	D01718	D01720_1	D01721	D01722	D01733	D01734
D01735	D01736_1	D01737	D01738	D01740	D01741_1	D01742_1	D01742_2	D01743
D01744	D01745	D01747_2	D01748_1	D01750	D01751	D01752	D01753_2	D01754
D01756_1	D01758	D01759_1	D01760	D01762_3	D01763	D01764	D01765_1	D01766_1
D01767	D01768_1	D01769_2	D01770	D01771	D01772	D01773	D01775	D01778_2
D01782_2	D01783	D01784	D01785_1	D01786_1	D01786_2	D01787	D01788_1	D01789_1
D01790_2	D01791	D01792	D01793_1	D01794_1	D01794_2	D01795_1	D01798_1	D01799
D01800	D01801_2	D01802_1	D01803	D01804	D01805	D01806_2	D01807	D01808_1
D01809	D01810	D01811	D01812_1	D01813	D01814	D01815_2	D01816_1	D01816_2
D01818	D01819_1	D01820	D01821_1	D01822	D01823	D01824	D01825	D01826_2
D01827	D01828	D01829	D01830_2	D01831_2	D01832	D01834	D01835_1	D01836_2
D01837_2	D01838_1	D01839_1	D01840_2	D01841	D01842	D01843	D01844_1	D01845
D01846	D01847_2	D01848_1	D01849_2	D01850_1	D01851	D01852_1	D01853	D01854_1
D01855_1	D01856_2	D01857_3	D01858	D01859	D01861_2	D01862_2	D01863_1	D01864
D01865	D01866	D01868_1	D01869	D01870	D01871_1	D01872_3	D01873_2	D01874
D01875_1	D01876_2	D01877	D01878_1	D01879_2	D01881_2	D01882	D01883	D01885
D01886	D01887_1	D01888_1	D01889	D01890	D01891	D01892	D01894	D01895
D01896	D01897	D01898	D01899_1	D01900	D01902_1	D01903	D01904_1	D01907
D01908	D01909_1	D01911_1	D01912_1	D01913	D01914	D01915_1	D01918_2	D01919
D01920	D01922	D01923_1	D01924	D01925_2	D01926_3	D01928_1	D01929_1	D01930_1
D01932	D01933_1	D01935_2	D01938_1	D01939_3	D01941_1	D01941_2	D01942	D01943_1
D01944_1	D01946_1	D01947	D01949	D01950_1	D01951_1	D01952_1	D01953	D01954
D01955_1	D01955_2	D01956	D01957	D01958_2	D01959	D01961_3	D01962	D01963
D01964	D01965	D01966	D01967_1	D01968_2	D01969_2	D01970	D01971	D01972
D01973_2	D01974	D01975	D01977	D01978_1	D01980	D01981	D01982_2	D01983_2
D01983_3	D01985_1	D01986	D01987_2	D01988	D01989	D01990_1	D01991_2	D01992_2
D01994_3	D01995_2	D01995_3	D01996_1	D01997	D01998	D02000_2	D02001_1	D02002_1
D02003_1	D02004_3	D02005_2	D02008	D02009_1	D02011_1	D02013_1	D02014	D02016_2
D02017_1	D02018	D02020	D02021_1	D02021_2	D02024_1	D02026_1	D02027	D02029_2
D02030_1	D02031_1	D02032	D02034_2	D02035	D02039_1	D02040_1	D02041_2	D02042_1
D02045_1	D02046	D02047_1	D02048	D02049_1	D02050_1	D02054_1	D02059_1	D02061
D02062	D02063_2	D02064_2	D02067	D02068_1	D02070_1	D02073	D02074_1	D02075_1
D02076_2	D02080_1	D02081_2	D02082_2	D02084_2	D02087_1	D02088_1	D02089_1	D02090_2
D02091_1	D02093_1	D02095_2	D02096_3	D02097_1	D02098_1	D02099_1	D02102_2	D02103_2
D02104_2	D02108_1	D02110_1	D02114_2	D02115_1	D02116_1	D02117_1	D02119_1	D02123_1
D02124_1	D02125_2	D02126_1	D02134_1	D02137_1	D02140_1	D02142_1	D02143	D02145_2
D02148_1	D02156	D02157_1	D02157_3	D02158_1	D02159	D02160	D02164_1	D02165_1
D02166_1	D02167_2	D02168_1	D02169_1	D02170_1	D02171	D02173_2	D02176	D02177_2
D02178_1	D02179_1	D02181_1	D02185_1	D02186_1	D02187_3	D02189_2	D02190_1	D02191_1
D02192_1	D02193_1	D02194_1	D02195_2	D02196_1	D02197_2	D02198_1	D02199_1	D02201_1
D02202_2	D02203_1	D02204_1	D02205_2	D02207_4	D02208_2	D02209_1	D02210_3	D02211_1
D02212_2	D02217_2	D02218_1	D02219_2	D02221_1	D02222_2	D02223_1	D02228_1	D02229_1
D02230_1	D02231_1	D02232_2	D02233_1	D02234_3	D02237_1	D02238_2	D02241_1	D02242_1
D02245_2	D02249_2	D02251_2	D02252_1	D02255_1	D02260_2	D02264_1	D02265_1	D02266_1
D02268_1	D02270_1	D02274_10	D02276_1	D02277_1	D02278_1	D02280_1	D02284_1	D02285_3
D02286	D02287	D02288	D02289	D02290_2	D02292_2	D02293_1	D02294_1	D02296
D02298_1	D02299	D02301	D02302	D02303	D02305	D02306	D02309	D02310
D02311	D02312_1	D02314	D02316	D02319	D02321	D02322	D02323	D02324
D02327_1	D02328	D02330	D02331	D02333	D02334	D02335	D02339	D02341
D02344	D02345	D02347	D02348	D02350	D02355	D02361	D02363	D02365
D02367	D02371_1	D02372_1	D02373	D02376	D02377	D02378	D02379	D02380
D02381	D02382	D02384	D02385	D02386	D02387_1	D02388	D02389	D02391
D02392	D02393	D02396_1	D02398_1	D02401	D02404_2	D02405_2	D02406	D02407_1
D02410	D02411_1	D02412	D02413	D02414_1	D02415_1	D02417_2	D02420	D02423_1
D02424	D02425	D02426	D02427	D02428	D02429	D02430	D02431	D02432_1
D02433_2	D02434	D02435	D02437	D02438	D02439	D02440	D02441	D02442

Continued on next page

Continued from previous page

D02443_1	D02445_1	D02446	D02447_1	D02449	D02450_1	D02451_1	D02452_2	D02453
D02454_1	D02455_1	D02456	D02458_1	D02459_1	D02460	D02462	D02463_1	D02465_1
D02466_3	D02468	D02469_1	D02470_1	D02471_1	D02472_1	D02473	D02474	D02475_1
D02476	D02477_2	D02478	D02479	D02480	D02481	D02482	D02483	D02484_2
D02485_2	D02486	D02487_2	D02489	D02490	D02492	D02494	D02495	D02496
D02497	D02500	D02501_2	D02503_1	D02503_2	D02504_2	D02507	D02508_1	D02510
D02511	D02512_2	D02513	D02514	D02515	D02516	D02517	D02518	D02519
D02520_1	D02521_1	D02522	D02523_1	D02524_1	D02526_2	D02527_1	D02528	D02529_1
D02535_2	D02537	D02538_2	D02539_1	D02540_1	D02546_2	D02547	D02548_1	D02549
D02550	D02551	D02552	D02553	D02554	D02555	D02556	D02557	D02558_1
D02559	D02560	D02561	D02562_1	D02563	D02564_1	D02565	D02567_2	D02568
D02569_1	D02570_3	D02571_1	D02572_1	D02573_1	D02574_1	D02575	D02576	D02577
D02578	D02579	D02580	D02581	D02582	D02584	D02585	D02586	D02587_2
D02588	D02589	D02590	D02591	D02592	D02593_3	D02594	D02595	D02600
D02602	D02603_1	D02604_1	D02605_1	D02606	D02607_1	D02609_2	D02610_1	D02611
D02613	D02614	D02616	D02617	D02618	D02619	D02620	D02621	D02625
D02626	D02627	D02628	D02629	D02630	D02631_1	D02632_1	D02633_2	D02634_1
D02635	D02636_2	D02637_1	D02638	D02639	D02640	D02641_1	D02642	D02644_1
D02645_2	D02646	D02647	D02648_2	D02649	D02650_1	D02651_1	D02652_1	D02653_1
D02654	D02655_1	D02657	D02658	D02659	D02660_1	D02661	D02662	D02663_1
D02665	D02666	D02667_1	D02668	D02669_1	D02670_2	D02672	D02673	D02674_1
D02675	D02676	D02677_2	D02678_1	D02679	D02680	D02681_1	D02682	D02684_1
D02685_1	D02686	D02687_1	D02688_1	D02689	D02690_1	D02693	D02694	D02696
D02697	D02698	D02699_1	D02700_1	D02701	D02702_1	D02703_1	D02704_2	D02705_2
D02708_3	D02709	D02710	D02712_1	D02714	D02715_1	D02716	D02717	D02718
D02719	D02720	D02721	D02722	D02723	D02724	D02725	D02726	D02727_1
D02728_1	D02729_1	D02730_1	D02731_1	D02732_1	D02733_1	D02738	D02739	D02741_1
D02742	D02746_1	D02747	D02748	D02750	D02751	D02752_1	D02753	D02754
D02755	D02757_1	D02758_1	D02759_1	D02759_2	D02760	D02761_1	D02762	D02763
D02764_1	D02765_2	D02766	D02767_1	D02768	D02769	D02770	D02772_2	D02773
D02774	D02775_1	D02780_1	D02785	D02787	D02797	D02801	D02803_2	D02804
D02806	D02808	D02809	D02812	D02813	D02814_1	D02817	D02818	D02822_2
D02824	D02826_1	D02828	D02829_2	D02830	D02833	D02834_1	D02835_2	D02836_1
D02838	D02839_2	D02840	D02841	D02846	D02847	D02848_1	D02849	D02850
D02851	D02867_3	D02876_2	D02877_1	D02878_2	D02880_1	D02881	D02883	D02884_2
D02885	D02886	D02887	D02888	D02890	D02891_1	D02892	D02893	D02894_1
D02896	D02897	D02898_1	D02898_2	D02899	D02900	D02901_2	D02902	D02904
D02906_1	D02908_1	D02909	D02911_2	D02912_1	D02913	D02919_3	D02924_1	D02925_1
D02926	D02927	D02928_1	D02929_1	D02930	D02931	D02932	D02933_1	D02936_1
D02939	D02940	D02941	D02943_1	D02944	D02948	D02950_1	D02951	D02952
D02964	D02965	D02966	D02968	D02972_1	D02973_1	D02974	D02975	D02976_1
D02978	D02984	D02985	D02987	D02989	D02991_2	D02992_1	D02993_1	D02995_1
D02996	D02998	D02999	D03001_1	D03002	D03003_1	D03005_1	D03007	D03008
D03009_1	D03010	D03011_2	D03012	D03014_1	D03015	D03016	D03017	D03018
D03019	D03021	D03022_2	D03023	D03024_1	D03025_1	D03026_1	D03027_1	D03028
D03029	D03030	D03031	D03032	D03034	D03035	D03036	D03037_2	D03038_2
D03039_1	D03040	D03041_3	D03042	D03046_1	D03051_1	D03052	D03056	D03057
D03059_1	D03060_1	D03061	D03062_2	D03064_2	D03067	D03070	D03071	D03072_1
D03073	D03074	D03075	D03076_2	D03077	D03078_1	D03079	D03080	D03081
D03083	D03086	D03087_1	D03088_1	D03089_1	D03090	D03092	D03093	D03094_3
D03097	D03098	D03099_1	D03100	D03101	D03103_1	D03104_1	D03105	D03106
D03107	D03108_1	D03110_1	D03112_1	D03113	D03114_1	D03116	D03118	D03120
D03121_1	D03126_1	D03127_1	D03128_1	D03130_1	D03132_1	D03133_2	D03134_4	D03135_1
D03137	D03144	D03145	D03146	D03147	D03148	D03152	D03153_2	D03154_1
D03155	D03156_2	D03158	D03159	D03160_2	D03161	D03163_2	D03164	D03166_1
D03167	D03168	D03169_1	D03170_1	D03171	D03172	D03174	D03175_2	D03176
D03178	D03179	D03180	D03181	D03182	D03183_1	D03184_1	D03185_2	D03186
D03187	D03188_2	D03189	D03190	D03191	D03193_1	D03193_2	D03194	D03195
D03196_1	D03198_1	D03199_1	D03200	D03202	D03206_1	D03208	D03209_1	D03210
D03211	D03212	D03213	D03214_1	D03217	D03218	D03219	D03220	D03221
D03225_1	D03227	D03233_1	D03239_2	D03241	D03246_1	D03246_3	D03248	D03249
D03252	D03253	D03254_1	D03255_2	D03264_1	D03267_2	D03268_1	D03272_1	D03281_1
D03286	D03290_2	D03292_3	D03294_4	D03298_1	D03299	D03301	D03307_1	D03319
D03340	D03342_2	D03349	D03350_1	D03351_1	D03355_1	D03359	D03361_2	D03363
D03365	D03366_2	D03367_1	D03369	D03370	D03371	D03373_1	D03376_1	D03377
D03382	D03384_1	D03385_1	D03386	D03387_1	D03389_1	D03390	D03392	D03399
D03402_2	D03405_2	D03406_1	D03407	D03408	D03409	D03410	D03412_2	D03413

Continued on next page

Continued from previous page

D03414_2	D03416	D03419	D03421	D03422_1	D03423	D03424	D03425_2	D03426_1
D03427	D03428	D03429_2	D03430_2	D03431	D03432	D03433_1	D03440	D03442
D03447_1	D03448_1	D03449_2	D03450_2	D03451_1	D03452	D03459_2	D03460	D03462
D03463_1	D03464	D03467	D03468	D03470	D03471_1	D03472	D03473	D03476_2
D03485_2	D03486	D03487	D03489	D03490_1	D03493	D03494_1	D03495	D03497
D03499_1	D03500	D03501	D03502	D03504	D03506_2	D03507_1	D03508	D03509
D03510	D03512	D03513	D03514	D03515	D03516	D03517	D03518_1	D03519_1
D03520	D03521	D03522_2	D03523	D03526	D03527	D03528_2	D03529_1	D03531
D03532	D03533	D03534	D03536_2	D03537	D03540	D03541	D03542	D03543_1
D03544_1	D03545	D03546	D03547_2	D03548	D03549	D03550	D03551	D03552
D03553	D03554	D03555	D03558_1	D03559	D03560	D03561	D03563	D03564
D03565	D03566_2	D03568_1	D03568_2	D03569	D03571_1	D03572	D03581	D03583
D03584	D03585	D03587_1	D03588_1	D03589	D03594	D03595	D03596_1	D03600
D03601	D03602_1	D03607_1	D03610	D03614_1	D03618	D03619	D03620_1	D03623
D03625	D03627	D03628	D03629_1	D03630	D03631	D03632_1	D03633	D03634
D03636	D03640	D03641_1	D03642	D03643	D03646_1	D03648_2	D03649_2	D03650
D03652_1	D03653_2	D03655	D03656	D03657	D03658	D03659_1	D03660_1	D03661
D03662_2	D03663_2	D03664_1	D03665	D03666	D03667	D03668_1	D03669	D03671
D03672_1	D03675	D03678_1	D03679_1	D03681	D03685_1	D03687	D03688	D03689
D03690	D03691	D03693	D03694	D03696	D03697	D03698	D03699	D03700_1
D03702_1	D03705_2	D03706	D03707_1	D03709_1	D03710	D03712	D03713_1	D03714
D03716	D03717_1	D03720	D03722_1	D03724_1	D03727	D03728_1	D03730	D03733_1
D03734	D03735	D03736	D03738	D03741	D03743	D03745_1	D03747	D03749
D03751_1	D03754	D03757	D03758	D03759_1	D03760	D03762_1	D03763	D03764
D03765_1	D03766	D03767_1	D03769_1	D03771	D03772	D03773	D03774	D03775
D03776_1	D03777	D03780	D03781_2	D03782	D03783_1	D03786	D03787_1	D03788
D03789	D03790	D03791	D03792	D03793	D03794	D03795	D03796	D03797
D03798	D03799	D03800	D03801_1	D03802	D03803_1	D03804	D03805	D03806
D03807	D03808_1	D03809	D03810_1	D03811_1	D03812	D03813	D03815_1	D03816_1
D03817	D03818	D03820	D03823	D03824_2	D03828_1	D03829_2	D03831	D03838
D03839_1	D03840	D03842	D03843_1	D03845_1	D03846	D03847_1	D03848	D03853
D03864_1	D03865	D03867	D03871	D03873	D03874	D03875_1	D03877_1	D03878_2
D03882	D03883	D03884	D03887_2	D03889	D03890	D03891	D03893_1	D03894
D03897	D03898	D03900_1	D03903_2	D03904_1	D03906	D03907	D03908	D03909
D03910	D03911	D03913	D03915	D03917	D03920	D03921_2	D03924_1	D03929
D03930	D03932	D03933_2	D03934	D03937_1	D03938	D03941_1	D03942	D03947_1
D03948	D03950_1	D03953_1	D03954	D03955	D03956	D03961	D03962_1	D03963_2
D03964	D03966_2	D03967	D03968_1	D03970	D03971_1	D03973	D03975	D03977
D03978_2	D03979	D03981	D03982_2	D03984	D03985_2	D03986_1	D03987	D03988
D03990_1	D03991_2	D03992_1	D03993	D03994_1	D03995	D03996_1	D03997	D03998
D03999	D04001_1	D04001_2	D04002	D04003_1	D04004	D04005_1	D04006	D04007
D04008_2	D04009_1	D04010	D04012	D04013	D04014_1	D04017	D04020	D04021
D04023_1	D04024_2	D04025	D04027	D04028	D04029_1	D04031	D04033	D04034_2
D04035	D04037	D04039_1	D04041	D04042	D04043_5	D04048_3	D04049_1	D04050
D04051	D04054_1	D04058_1	D04061	D04063	D04064	D04065	D04066	D04067_1
D04068_2	D04069	D04072_1	D04074	D04075	D04076	D04077	D04078_1	D04079_1
D04080	D04085_1	D04086_1	D04087	D04088	D04089	D04090	D04092_1	D04093
D04094	D04096	D04097	D04098_1	D04100	D04101	D04102	D04103_1	D04104
D04105_2	D04107	D04108	D04110_1	D04111	D04112	D04115	D04116_2	D04117
D04118_2	D04120_1	D04122	D04124_1	D04125_1	D04126_1	D04127	D04128_1	D04129_1
D04130_1	D04131	D04132	D04135	D04138	D04139	D04140	D04141	D04142
D04143	D04144	D04145	D04146	D04147_1	D04149	D04150	D04151	D04152_1
D04153	D04155_1	D04156_1	D04158	D04159_1	D04160	D04161	D04162	D04163_1
D04164	D04165_1	D04174_2	D04178_1	D04179_1	D04183	D04184_1	D04185	D04186
D04187_1	D04188_1	D04189	D04190_1	D04191	D04192	D04193	D04194	D04195
D04196	D04197	D04198	D04199_1	D04200	D04201	D04202	D04203	D04204_1
D04205	D04206	D04207	D04208	D04209	D04210	D04211	D04212	D04213
D04214	D04215	D04217	D04218	D04219	D04220	D04221	D04222	D04223_1
D04224	D04225	D04226_1	D04227	D04228	D04229	D04230	D04231_1	D04232
D04233	D04234	D04235	D04236	D04237	D04238	D04239	D04241	D04243
D04244	D04245_1	D04247	D04248	D04254_1	D04255_1	D04256	D04257_1	D04258
D04260	D04262_1	D04263	D04264_1	D04270	D04273	D04274_1	D04276_1	D04277
D04278	D04279	D04280	D04281	D04282	D04283_3	D04286_3	D04288_3	D04294_1
D04298	D04300	D04301_1	D04302_1	D04303_1	D04308	D04309_1	D04310	D04312
D04314_1	D04315	D04316	D04317	D04320	D04322	D04327	D04328	D04332
D04334	D04350	D04353	D04354	D04355	D04356	D04361_3	D04368	D04373
D04376_1	D04378	D04384_1	D04386_1	D04387_1	D04398	D04399_1	D04401_1	D04409

Continued on next page

Continued from previous page

D04411	D04413_1	D04414	D04433_1	D04434_1	D04435_1	D04436_1	D04437	D04438
D04439	D04444_1	D04450	D04452_1	D04473_1	D04476	D04485	D04486_1	D04488
D04491_2	D04492_2	D04494	D04497	D04498	D04500	D04501_1	D04502_1	D04503
D04504	D04506	D04508_1	D04509	D04512_1	D04513	D04514_1	D04517_1	D04519
D04520	D04521_1	D04522	D04526_1	D04529	D04530	D04531	D04533_1	D04534
D04535_1	D04537	D04538	D04555	D04556_1	D04601_2	D04602_1	D04605	D04607
D04608	D04609_6	D04610_1	D04611_1	D04614	D04617	D04618	D04619_2	D04622
D04623	D04627	D04628_1	D04630_1	D04631_1	D04632	D04633	D04634_1	D04635
D04636	D04637	D04638	D04639	D04640_1	D04641	D04642	D04643	D04645
D04646	D04648	D04649	D04650	D04651	D04652_2	D04653	D04657	D04658_1
D04659	D04666_1	D04669_1	D04671	D04672_2	D04673	D04678	D04679_1	D04680_2
D04683_1	D04685_1	D04687	D04691	D04693	D04696	D04697_1	D04698_1	D04700_1
D04713_1	D04717_1	D04718_1	D04720	D04724	D04725	D04728	D04730	D04733
D04734_1	D04735	D04736	D04737	D04738	D04739	D04741	D04746	D04747_1
D04748	D04751_1	D04752	D04754	D04757	D04758	D04759	D04760	D04762_1
D04763_1	D04764_2	D04765_1	D04766_1	D04767	D04768	D04770	D04774_2	D04777
D04778_1	D04779	D04780	D04781	D04782_1	D04783_1	D04784_1	D04785_1	D04786
D04787	D04788_1	D04789	D04790	D04791_1	D04793	D04794_2	D04820_1	D04821_2
D04822_1	D04823_1	D04825_1	D04826	D04827	D04830_2	D04848_1	D04859	D04860_3
D04862	D04863	D04864	D04868_1	D04869	D04871_1	D04871_2	D04872_1	D04876
D04877	D04878	D04884	D04885	D04886_1	D04888	D04890_1	D04890_2	D04891
D04893_1	D04894	D04896_1	D04897	D04900	D04903	D04905_1	D04906_1	D04907_2
D04908_3	D04912	D04914	D04920_1	D04924_1	D04925	D04926	D04927	D04933
D04936	D04941	D04947	D04949_1	D04950	D04951	D04953_1	D04955_1	D04964_1
D04970_1	D04975	D04976	D04979	D04982	D04985	D04986	D04988_1	D04989
D04991	D04992	D04997_1	D05000	D05001_1	D05002_1	D05003	D05004	D05005
D05006_2	D05007_1	D05009	D05010	D05012	D05013_2	D05015	D05017	D05018
D05020_1	D05021	D05024_1	D05025	D05027	D05029	D05030	D05031_1	D05032
D05033_1	D05035_1	D05038_2	D05039	D05041	D05044	D05046_3	D05048_2	D05049_1
D05050_1	D05052	D05053_2	D05054	D05060	D05062_1	D05063	D05064_1	D05065
D05067	D05068	D05072	D05075	D05076_1	D05078	D05080_1	D05082	D05083
D05084	D05085_1	D05086	D05087	D05091	D05093	D05095_1	D05096	D05097
D05098	D05099	D05100_2	D05101_1	D05102	D05103_1	D05104_2	D05105_1	D05106_2
D05107_2	D05108	D05110_1	D05112_2	D05114	D05115_2	D05116	D05118_1	D05119_1
D05120_1	D05121	D05122	D05124_1	D05125	D05127	D05129	D05131_1	D05133_1
D05134	D05135_2	D05137	D05138_2	D05139_1	D05139_2	D05140	D05143	D05144
D05145_1	D05149	D05150	D05152	D05153	D05154_2	D05155	D05159	D05160
D05161	D05162	D05163	D05164	D05165	D05166	D05167	D05168	D05169
D05170	D05171_2	D05172	D05173	D05174	D05176	D05177	D05178_1	D05179_1
D05180_2	D05184	D05185	D05191	D05192_2	D05193	D05194	D05195_1	D05196
D05197	D05198	D05199_1	D05200_1	D05203_1	D05204	D05205	D05207	D05208
D05209	D05210	D05211	D05212	D05213	D05215_1	D05217_2	D05219	D05220
D05221	D05222_2	D05224	D05225	D05226	D05227	D05228	D05229	D05231_2
D05232	D05234	D05235_1	D05241_1	D05245	D05246	D05249	D05250	D05267_1
D05268_2	D05271_1	D05273	D05274	D05275	D05278	D05280_1	D05281_1	D05284_2
D05285	D05290_1	D05291	D05292	D05293	D05294	D05296	D05298_1	D05299
D05300	D05302_2	D05303	D05305	D05306_2	D05308	D05309	D05319_1	D05320
D05322_1	D05323	D05333	D05334	D05335	D05337	D05339	D05340	D05341
D05342_3	D05343_1	D05344	D05345_2	D05346	D05348	D05351	D05353	D05359_1
D05360_2	D05363	D05365	D05367_1	D05371_1	D05378	D05380_1	D05381	D05385
D05391	D05395_1	D05396	D05397	D05399	D05400	D05402	D05403_1	D05405_1
D05406	D05407	D05413_2	D05414	D05416	D05417	D05418_1	D05419_4	D05422
D05423	D05426_1	D05427	D05428_1	D05430	D05436	D05437	D05438	D05439
D05440	D05441	D05442_1	D05443	D05447	D05450_1	D05451_2	D05452	D05453_1
D05454_2	D05455_1	D05456	D05457	D05458	D05460	D05468	D05469_2	D05471_1
D05472_1	D05473_1	D05474	D05475_1	D05476_2	D05477	D05479_1	D05480	D05481_1
D05482_1	D05483	D05484	D05485_1	D05487	D05488_1	D05489	D05490_1	D05491_1
D05492_3	D05493	D05494_1	D05495	D05496_1	D05497_1	D05498	D05499_2	D05500_2
D05501_1	D05502	D05503	D05504	D05506	D05507_1	D05508	D05509	D05510_2
D05512	D05514	D05515	D05516	D05517	D05518_2	D05522	D05523	D05528
D05529	D05535_2	D05536	D05537_1	D05537_2	D05538	D05539	D05540_1	D05540_2
D05545	D05546_1	D05547	D05549	D05552	D05553_1	D05553_2	D05554	D05556_1
D05558_1	D05565	D05568	D05572	D05579_1	D05582_1	D05583_1	D05587	D05588_1
D05589	D05590_2	D05592_1	D05596_1	D05597_1	D05598_1	D05601	D05602	D05603
D05604	D05605	D05606_1	D05607_1	D05607_2	D05608_1	D05609	D05611_1	D05612
D05613_1	D05614_2	D05615_1	D05616	D05617	D05618_1	D05619	D05620_1	D05621
D05623_1	D05626	D05627	D05628_1	D05629	D05630	D05631_1	D05633	D05635

Continued on next page

Continued from previous page

D05637	D05638	D05639	D05640_1	D05641_1	D05642	D05645	D05646_1	D05653
D05656_1	D05656_2	D05658	D05661	D05663_1	D05664_1	D05665	D05666_1	D05667_2
D05668	D05669	D05670	D05671_1	D05672_1	D05673_1	D05674	D05675_1	D05676_1
D05677_1	D05678_1	D05678_2	D05679	D05680	D05682_1	D05683_1	D05684_3	D05685_1
D05690	D05692_2	D05694	D05695	D05700	D05703_1	D05708_1	D05710	D05711
D05713	D05714_1	D05715	D05717	D05718_1	D05719	D05720	D05722_1	D05726
D05727	D05728	D05729	D05730_1	D05731	D05732	D05733	D05735	D05736_1
D05737_1	D05738	D05739	D05741_1	D05742_3	D05743	D05744	D05745	D05746
D05747	D05748	D05749	D05750	D05751	D05752	D05753	D05754_2	D05755
D05756	D05757	D05758	D05759	D05767	D05768	D05770	D05772_2	D05775
D05776	D05778	D05779_1	D05780	D05784	D05787_1	D05789_2	D05790	D05795_1
D05796	D05797	D05798_1	D05799_1	D05800	D05801_1	D05802_1	D05804_1	D05805
D05806	D05807_1	D05809	D05810	D05811	D05812_1	D05813	D05817	D05818
D05819	D05820	D05822	D05824_1	D05825	D05826_1	D05828_1	D05829_1	D05830
D05832_1	D05833_1	D05834	D05842	D05843	D05845	D05846	D05848	D05856_1
D05857_1	D05858_2	D05861_1	D05864_1	D05866_1	D05867_1	D05868_2	D05871_1	D05872_1
D05873_1	D05876_2	D05879_1	D05880_2	D05883_1	D05891_1	D05892	D05893	D05894
D05895	D05896	D05897	D05898	D05899	D05900	D05901	D05902_2	D05903
D05904_1	D05905	D05906	D05907_1	D05908	D05910	D05911_1	D05912	D05914_1
D05915	D05916_1	D05922_3	D05924	D05925	D05926_1	D05927_1	D05928	D05929
D05930_1	D05931	D05932	D05935	D05936_1	D05937	D05939	D05940_4	D05941_1
D05942	D05943	D05945	D05946	D05947	D05948	D05950_1	D05951	D05952
D05953	D05954	D05955_1	D05956	D05960_1	D05962	D05964	D05965	D05966
D05967	D05968	D05969	D05970	D05971_1	D05973	D05974_1	D05975	D05977
D05979_1	D05980	D05981_1	D05984_1	D05985	D05986_1	D05988	D05992	D05993
D05995	D05996_1	D05997_1	D05998	D05999_1	D06000_2	D06001	D06002	D06003
D06004_1	D06005	D06007	D06008	D06009_1	D06010	D06011_1	D06012_1	D06013
D06014_2	D06015_2	D06016_1	D06017	D06018	D06019	D06020_1	D06021	D06027_1
D06029_1	D06030_2	D06034_1	D06034_4	D06039_1	D06047_1	D06047_3	D06047_4	D06049_1
D06049_3	D06051	D06055	D06057_1	D06059_1	D06060_1	D06061_1	D06062	D06063
D06064	D06065	D06066	D06067	D06068	D06069	D06072	D06073_1	D06074_1
D06075	D06076	D06077	D06079_1	D06081_1	D06082	D06083	D06084_1	D06085
D06086	D06087	D06092	D06093_1	D06094	D06095_1	D06096	D06097	D06099
D06100_1	D06101_1	D06101_2	D06102	D06104_2	D06105	D06106	D06107_1	D06109_1
D06110	D06111	D06112_1	D06113_1	D06114	D06115	D06117	D06123	D06124_1
D06125	D06126_1	D06127	D06129_1	D06130	D06131	D06132_1	D06133	D06134
D06135	D06136	D06137_1	D06138_1	D06139	D06140	D06141	D06143_1	D06144_2
D06145	D06146_1	D06147_1	D06148	D06149_1	D06150_1	D06151	D06152	D06153
D06154	D06156	D06157	D06158	D06159_1	D06160	D06161	D06162_1	D06163_1
D06164_1	D06165	D06166_1	D06167	D06170	D06171	D06173	D06174	D06175_2
D06176	D06177_1	D06178_1	D06180	D06181	D06182	D06183	D06186	D06187
D06189_2	D06190	D06191	D06192	D06195	D06196	D06197	D06198	D06199
D06200	D06201	D06202	D06203	D06204_1	D06205_1	D06206_2	D06208_2	D06209
D06211	D06212_1	D06213	D06214	D06215	D06216_3	D06217_1	D06218	D06219
D06220	D06221	D06223	D06224_2	D06225	D06226	D06227_1	D06228	D06229
D06230	D06231	D06232	D06234_1	D06235	D06237	D06238	D06240	D06241_1
D06245_2	D06247	D06251_1	D06252	D06253_2	D06254_1	D06255	D06257_2	D06260
D06263	D06265	D06266	D06267	D06268	D06272_2	D06273	D06274	D06275
D06277	D06278_2	D06280_2	D06281	D06282_1	D06283_2	D06285	D06288_1	D06289_1
D06290_1	D06291	D06292	D06293_1	D06294_4	D06295	D06296	D06297_1	D06300_1
D06303	D06305_2	D06306_1	D06307_1	D06309_1	D06310	D06315	D06316	D06317_1
D06318	D06320	D06328	D06330	D06332_1	D06334_2	D06335_1	D06336	D06340
D06341	D06342_1	D06343_3	D06344_1	D06346	D06353_3	D06354_1	D06355_1	D06357
D06358_2	D06360_1	D06361_1	D06364	D06366	D06367_1	D06372	D06373_1	D06374_1
D06376_1	D06377_2	D06378_4	D06379_10	D06380_1	D06382_1	D06383	D06384_1	D06386_2
D06387_2	D06394_1	D06395_1	D06396	D06402_1	D06403	D06405_2	D06406	D06407
D06413_1	D06465_4	D06490	D06543	D06551	D06552	D06553_1	D06554_1	D06556_1
D06557_1	D06559	D06560	D06562_2	D06564_2	D06566	D06568_1	D06569	D06570
D06573	D06574_1	D06575	D06576_1	D06578_1	D06579_1	D06582	D06590	D06597_1
D06598	D06600	D06607_1	D06609	D06612_2	D06613_1	D06618_1	D06619_3	D06625_1
D06627	D06629_1	D06630	D06631	D06632	D06634	D06637	D06638	D06640
D06641	D06645_1	D06646	D06647	D06648	D06650	D06651_1	D06652	D06656_1
D06658_1	D06659	D06660_1	D06661_1	D06662_1	D06664	D06665_1	D06670	D06671_1
D06672	D06673	D06674	D06675	D06676	D06677	D06678	D06876	D06877_1
D06878_2	D06881	D06882_1	D06883	D06884	D06885	D06887	D06888	D06890
D07058	D07059	D07061_1	D07062_1	D07063	D07064	D07067	D07068	D07069_1
D07070_1	D07071	D07072	D07073	D07074	D07075	D07077	D07078	D07079

Continued on next page

Continued from previous page

D07080	D07081_2	D07082_1	D07083_1	D07084_1	D07085_1	D07086	D07087	D07088
D07089	D07091	D07092	D07093	D07094_1	D07095	D07096	D07097_1	D07097_2
D07098	D07099_2	D07100_1	D07101	D07102	D07103_1	D07104	D07105	D07106
D07107	D07109	D07111	D07113	D07114	D07115	D07116	D07117	D07118
D07119	D07123_2	D07127	D07128	D07130	D07131	D07133_1	D07134	D07135
D07136	D07137	D07138	D07139	D07140	D07141	D07142	D07143	D07146
D07147	D07148	D07149	D07150	D07151	D07155	D07157_1	D07158	D07159
D07160	D07161	D07162	D07163	D07164	D07165	D07166	D07167	D07168
D07169	D07170	D07171	D07172	D07173	D07174	D07176	D07177	D07178
D07179	D07180	D07181	D07182	D07183	D07184	D07185	D07186	D07187
D07188	D07189	D07190	D07191	D07192	D07193	D07194	D07195	D07196
D07197	D07198	D07199	D07201	D07202	D07203	D07204	D07205	D07206
D07207	D07208	D07210_1	D07211	D07212	D07213	D07215	D07216	D07217
D07218	D07219	D07220	D07221	D07222	D07223	D07224_2	D07225	D07226
D07227	D07228	D07229	D07230	D07231	D07232_1	D07233_1	D07234	D07235
D07236	D07237	D07238	D07240	D07241	D07242	D07243	D07244	D07245
D07246	D07247	D07248	D07250	D07251	D07252	D07253	D07254	D07255
D07256	D07257	D07258	D07260	D07261	D07262	D07263_1	D07264	D07265
D07266	D07267	D07268	D07270	D07271	D07272	D07273_1	D07274	D07275
D07276	D07277	D07278	D07279	D07280	D07281	D07282	D07284	D07285
D07286	D07287	D07288	D07289	D07290	D07291	D07292	D07293_2	D07294
D07295	D07296	D07297	D07299	D07300	D07301	D07302	D07303	D07304
D07305	D07306	D07307	D07308	D07309	D07310	D07311	D07313	D07314
D07315	D07316	D07317	D07318	D07319	D07320	D07321	D07322	D07323
D07324	D07325	D07326	D07327	D07328	D07329	D07330	D07331	D07332
D07333	D07334	D07335	D07336	D07337	D07338	D07339	D07340	D07341
D07342	D07343	D07344	D07345	D07346	D07347	D07348	D07349	D07350
D07351	D07352	D07353	D07354	D07355	D07359	D07360	D07361	D07362
D07364	D07365	D07366	D07367	D07368	D07370	D07371	D07372	D07373
D07374	D07375	D07376	D07377	D07378	D07379	D07380	D07381	D07382
D07383	D07384	D07385	D07386	D07387	D07388_1	D07389	D07390	D07391
D07392	D07393	D07394	D07395	D07396	D07397	D07399	D07400	D07401
D07403	D07404	D07405	D07406	D07407	D07408	D07409	D07410	D07411
D07412	D07413	D07415	D07416	D07420_1	D07426	D07427_1	D07428	D07430
D07433	D07434	D07443	D07446	D07449	D07453	D07454	D07456	D07457
D07464	D07466	D07472	D07474	D07476	D07479	D07480_1	D07484_2	D07485
D07488	D07490	D07495	D07496	D07498	D07502_1	D07503_1	D07503_2	D07504
D07510	D07514	D07518	D07519	D07521	D07524_2	D07527	D07528	D07529
D07530	D07531	D07532	D07536	D07541	D07544	D07546	D07548_1	D07549_1
D07553	D07554_1	D07555	D07556	D07557_1	D07558_1	D07561	D07564_1	D07566
D07568	D07576	D07577	D07578	D07580_2	D07586_1	D07592_1	D07592_4	D07599
D07605_2	D07607	D07608	D07609_1	D07612_2	D07613	D07614	D07615	D07616
D07618_1	D07621	D07625	D07629	D07634	D07635	D07636	D07638	D07639
D07642	D07643	D07644	D07645	D07647	D07650	D07652	D07653	D07655
D07656	D07658	D07659	D07661	D07670	D07674	D07675	D07677	D07681
D07683_1	D07688	D07691_2	D07692_2	D07693	D07695	D07698	D07700	D07706
D07708	D07712_2	D07714	D07715	D07717	D07719	D07722	D07723	D07730
D07731	D07732	D07737_1	D07738_1	D07740_1	D07741	D07746	D07749	D07750
D07751	D07753	D07754	D07755	D07761	D07762	D07763	D07764	D07766
D07767	D07768	D07770	D07773	D07778_2	D07783	D07784	D07785	D07786
D07787	D07789	D07792	D07794_1	D07794_2	D07797	D07798_2	D07799	D07800
D07801	D07802	D07814	D07815	D07816	D07818_2	D07822_1	D07824_2	D07826_1
D07829	D07832_2	D07833_1	D07835	D07841	D07842_1	D07844	D07846	D07849_1
D07852_1	D07855	D07856	D07857	D07858	D07860_1	D07863	D07865	D07882
D07885	D07888	D07889	D07893	D07897	D07899	D07903	D07908	D07912
D07914	D07919	D07920	D07921_1	D07926	D07928	D07932	D07933	D07936
D07937	D07939	D07942	D07947	D07950	D07951_1	D07952	D07954	D07955_1
D07956_5	D07959	D07960_1	D07963	D07964	D07966	D07967	D07968	D07969
D07970	D07972	D07973	D07975	D07976	D07980	D07981	D07983	D07985_1
D07986	D07987_1	D07991	D07992	D07993	D07998	D07999_1	D08013	D08014
D08015_1	D08021	D08022_1	D08024	D08026	D08028	D08036	D08037	D08042
D08043	D08044	D08048	D08051	D08052	D08053	D08056	D08059_1	D08061
D08063	D08067	D08073	D08075	D08079	D08089	D08096	D08097_2	D08100
D08103_1	D08107	D08109	D08112	D08146	D08147	D08149	D08150	D08151
D08153	D08156	D08158	D08166	D08167	D08170	D08175	D08176_1	D08177_1
D08178	D08186_2	D08188	D08191_1	D08193	D08194	D08196	D08197	D08198
D08200	D08208	D08210	D08213	D08214	D08218	D08221	D08223_1	D08227

Continued on next page

Continued from previous page

D08228	D08229	D08235	D08242	D08243	D08247	D08250	D08251	D08260
D08272	D08274	D08275	D08277	D08279	D08281	D08282	D08283	D08285
D08292	D08295	D08298_2	D08299	D08301	D08302	D08307	D08314	D08320
D08326	D08328	D08329_1	D08331	D08332	D08336	D08337	D08338	D08341
D08342	D08345_1	D08350	D08351	D08354	D08357	D08359	D08361	D08367
D08369	D08370	D08372	D08373	D08374	D08383	D08385	D08388_1	D08392
D08399_1	D08404_2	D08406	D08409	D08410	D08412	D08413_1	D08414_1	D08415
D08416	D08429	D08431	D08433	D08437	D08438	D08442	D08450	D08452
D08455	D08463	D08467	D08470	D08475	D08476	D08477	D08482	D08484
D08487	D08492	D08494	D08498	D08502	D08503	D08507	D08515	D08518
D08520	D08527_1	D08527_2	D08532	D08534	D08537	D08538_1	D08539_1	D08541
D08542	D08543	D08545	D08548	D08550	D08553_1	D08565	D08567_1	D08568
D08573	D08574	D08575	D08580_1	D08581_1	D08582	D08583_1	D08584	D08586
D08589_1	D08591	D08599	D08601	D08603	D08605	D08610	D08612	D08615
D08616	D08627	D08628_2	D08630	D08631	D08632_2	D08633	D08634	D08635_1
D08637_1	D08640	D08646	D08647_1	D08650	D08655_2	D08656	D08657	D08659
D08660	D08661	D08665	D08671	D08672_1	D08674	D08676	D08677	D08682
D08688	D08691	D08692	D08735	D08836_1	D08837_1	D08838_1	D08840	D08845
D08850	D08852	D08855	D08856_1	D08859	D08860	D08861	D08862_2	D08863
D08864	D08865_1	D08868	D08869	D08870	D08871_2	D08872_1	D08873	D08876
D08878	D08879	D08880	D08881	D08884_1	D08885	D08886_2	D08888	D08890_1
D08892	D08893	D08897	D08899	D08900	D08902_1	D08904	D08905	D08906
D08907_1	D08909	D08911	D08912_2	D08913	D08914_1	D08915	D08916	D08917_1
D08919	D08924	D08925_1	D08928	D08931	D08932	D08935	D08937	D08938_1
D08939	D08940	D08945_1	D08949	D08950	D08951	D08954_1	D08955	D08956
D08958	D08963	D08964	D08965	D08967_2	D08969_1	D08970	D08971	D08973
D08974	D08975	D08976	D08978	D08979	D08981	D08983_1	D08984	D08986
D08987_2	D08988_1	D08989	D08991_1	D08994_1	D08995	D08996	D08998_1	D08999_1
D09002_1	D09003	D09008_1	D09009	D09012	D09014	D09017	D09018_1	D09019
D09020	D09022	D09024_1	D09026	D09028_2	D09032_1	D09033	D09035	D09036_1
D09038	D09198_1	D09200	D09202	D09205	D09208	D09209	D09215	D09216
D09318	D09320	D09322	D09323	D09324	D09330	D09335	D09336_1	D09338
D09344	D09346_1	D09347	D09348_1	D09349	D09350	D09351	D09353	D09354_1
D09357_1	D09358	D09360	D09361	D09362	D09364	D09365	D09366_1	D09367
D09369	D09375	D09377	D09378	D09380	D09381	D09382	D09385	D09386
D09388	D09389_1	D09390	D09391	D09392_1	D09393	D09394	D09396	D09399
D09401	D09402_1	D09404	D09410	D09535	D09537	D09539	D09544	D09546_1
D09547_2	D09566	D09567	D09568	D09569	D09570	D09571	D09572	D09573_1
D09576	D09581	D09582	D09584	D09585	D09586_1	D09589	D09592_2	D09598
D09599	D09602	D09604	D09607	D09608_1	D09609	D09610	D09612	D09613
D09614_1	D09616	D09617	D09618	D09619_1	D09621	D09625	D09627	D09634
D09635	D09636	D09637	D09639	D09640	D09642	D09644	D09645	D09647
D09648	D09650	D09655	D09658	D09664	D09666	D09667	D09671	D09672
D09673	D09676	D09677_1	D09678_1	D09679	D09683	D09684	D09685	D09686
D09687	D09689	D09691	D09692	D09693	D09695	D09696	D09697_2	D09698
D09701	D09702	D09703_1	D09707	D09708	D09709	D09712	D09714	D09717
D09720	D09721	D09722	D09724	D09730	D09731	D09732_1	D09738_1	D09740
D09750_1	D09751_1	D09755	D09757	D09758	D09765	D09768	D09769	D09770
D09772_1	D09773_1	D09774_1	D09779	D09780	D09781_1	D09782_1	D09783_1	D09784_2
D09785	D09786	D09787_1	D09789_1	D09792_1	D09793_2	D09796	D09797	D09799
D09801_1	D09804_2	D09806	D09807	D09816_1	D09817_1	D09821_1	D09824_1	D09828
D09830_2	D09832_1	D09835_1	D09837_3	D09840_1	D09841_1	D09843_1	D09843_2	D09849_1
D09850	D09855_2	D09861	D09862	D09863	D09864	D09866	D09869	D09870
D09872	D09876_1	D09878	D09881	D09882	D09883	D09884	D09890	D09893
D09899	D09902	D09903	D09904	D09906	D09915	D09916	D09917	D09919
D09921	D09922_1	D09923	D09925	D09928	D09929	D09931	D09933	D09935
D09938	D09942_2	D09949_2	D09950	D09953	D09955	D09957	D09959	D09962
D09963	D09964	D09965	D09968	D09969	D09971	D09972	D09973	D09974
D09976	D09978_2	D09981	D09983	D09985	D09986	D09987	D09990	D09991
D09992	D09994	D09995	D09996	D10002	D10004	D10005	D10006	D10007
D10008	D10009	D10013	D10016	D10017	D10019_1	D10020_1	D10021	D10022
D10026	D10027	D10028_1	D10029	D10032	D10055	D10060	D10062	D10064
D10065	D10066	D10068	D10069	D10073	D10075_1	D10076	D10078	D10079
D10081	D10082	D10085	D10086_1	D10087	D10088	D10090	D10093	D10097
D10098_1	D10099	D10102	D10107	D10108	D10113_1	D10117	D10119	D10121
D10122_1	D10125	D10126	D10127	D10128	D10134	D10135	D10136	D10137_2
D10140	D10141	D10143	D10146	D10147	D10151	D10154	D10155_1	D10158

Continued on next page

Continued from previous page

D10162	D10164_1	D10165	D10166	D10167	D10168	D10170	D10172	D10173
D10174	D10178	D10180	D10182	D10184	D10186	D10188	D10189	D10194_1
D10195_1	D10196	D10197	D10202	D10216	D10217_1	D10218	D10219	D10221
D10222	D10223	D10224	D10225	D10226_1	D10226_2	D10228	D10229	D10230
D10231	D10304	D10305_1	D10306	D10308	D10309	D10313	D10314	D10315
D10317	D10318	D10321	D10324	D10326	D10327	D10329	D10330	D10332
D10334	D10336	D10338	D10339	D10340_1	D10343	D10345	D10346	D10348
D10349	D10355	D10361	D10362	D10364	D10365	D10366	D10367	D10369
D10370	D10371	D10372	D10374	D10375_2	D10376_1	D10378	D10379_1	D10380
D10381	D10383	D10386	D10387	D10388	D10389_2	D10391	D10392_1	D10393
D10394_1	D10396_2	D10397	D10402	D10403	D10407	D10408	D10412	D10413
D10414	D10416	D10417	D10423	D10425	D10428	D10431	D10433	D10434
D10435	D10436	D10437	D10441	D10442	D10443	D10445	D10450_1	D10459
D10462_2	D10463_1	D10464	D10466_1	D10467_1	D10469_2	D10470_1	D10471_1	D10474_1
D10474_3	D10476	D10477	D10481	D10487	D10490	D10492	D10493	D10502
D10503	D10504	D10507	D10508	D10509	D10510	D10511	D10528	D10529
D10543	D10545	D10547	D10548	D10549_1	D10550	D10551	D10552	D10553
D10554	D10555	D10556	D10558	D10560	D10561	D10562_1	D10564	D10565
D10575	D10576	D10580	D10581_1	D10583	D10584	D10585	D10593	D10594_1
D10604	D10610	D10612	D10613	D10614	D10617	D10618	D10622_1	D10623
D10624	D10625	D10626	D10629	D10630	D10631	D10632	D10633	D10634
D10635	D10636	D10637	D10638	D10639_2	D10640	D10641	D10642_2	D10643_2
D10646	D10649	D10650_1	D10653	D10656	D10657_1	D10658	D10660	D10661
D10662	D10664	D10665	D10666	D10668	D10669	D10671	D10672	D10673
D10674	D10675	D10676	D10678	D10679	D10681	D10682	D10683_4	D10686
D10687	D10688	D10690_1	D10691	D10694_1	D10695	D10696	D10697	D10699
D10700	D10701	D10704	D10706	D10707	D10709	D10710	D10711	D10712
D10714	D10715	D10716	D10717	D10718	D10719	D10720_1	D10722	D10723
D10725	D10726_1	D10728	D10730	D10732	D10734	D10735	D10737	D10739
D10749	D10750	D10762_1	D10766_1	D10772_1	D10781	D10790_1	D10795_1	D10797_1
D10799_1	D10801	D10803	D10806	D10807	D10809_1	D10810	D10814	D10816
D10825	D10833_1	D10852	D10853	D10858	D10859	D10860	D10861	D10862
D10863	D10864	D10865	D10866	D10868	D10871	D10874	D10875	D10877
D10878	D10880							

Table G.3: List of the database IDs for molecules in the ZINC validation set. Molecules are sourced from `http://zinc15.docking.org/protomers/`.

324806330	324806332	324806367	324806456	324806474	324806529	324806715	324806717	324806796
324806907	324807021	324807054	324807077	324807081	324807121	324807300	324807313	324807316
324807454	324807576	324807653	324807735	324807820	324807926	324808087	324808175	324808227
324808337	324808456	324808598	324808630	324808697	324808735	324808801	324808885	324808938
324808975	324809101	324809115	324809125	324809133	324809166	324809187	324809371	324809884
324809999	324810332	324810474	324810563	324810607	324810646	324810700	324810715	324810719
324810751	324810826	324810846	324810850	324810984	324811033	324811173	324811358	324811428
324811447	324811449	324811736	324811882	324812045	324812097	324812220	324812227	324812267
324812290	324812406	324812454	324812461	324812534	324812545	324812560	324812703	324812715
324812737	324812811	324812839	324812996	324813097	324813249	324813275	324813411	324813478
324813587	324813600	324813604	324813784	324813799	324813817	324813846	324813851	324813868
324813914	324813997	324814181	324814221	324814287	324814294	324814301	324814342	324814369
324814448	324814476	324814617	324814751	324814755	324814838	324814908	324814916	324814959
324815173	324815276	324815389	324815437	324815446	324815497	324815622	324815772	324815805
324816066	324816081	324816089	324816174	324816187	324816327	324816449	324816453	324816639
324816724	324816774	324816799	324816873	324816897	324816955	324817071	324817091	324817135
324817178	324817384	324817388	324817632	324817720	324817910	324817916	324817942	324817982
324818050	324818108	324818144	324818185	324818188	324818246	324818318	324818331	324818393
324818506	324818544	324818546	324818778	324818852	324818882	324818957	324819393	324819432
324819436	324819513	324819704	324819740	324819759	324819828	324819896	324820079	324820102
324820127	324820128	324820187	324820202	324820290	324820310	324820313	324820333	324820342
324820544	324820622	324820722	324820744	324820770	324820845	324821065	324821113	324821204
324821240	324821346	324821398	324821482	324821532	324821550	324821644	324821744	324821767
324821790	324821914	324822053	324822076	324822346	324822403	324822443	324822543	324822662
324822796	324822841	324822861	324822862	324822916	324822921	324822981	324823061	324823193
324823238	324823287	324823326	324823334	324823455	324823503	324823575	324823589	324823634

Continued on next page

Continued from previous page

324823704	324823762	324823835	324823997	324824104	324824113	324824129	324824293	324824305
324824320	324824372	324824391	324824446	324824447	324824458	324824470	324824526	324824583
324824614	324824685	324824686	324824867	324824892	324824957	324825031	324825110	324825423
324825436	324825624	324825762	324825811	324825846	324825887	324825916	324825965	324825967
324825986	324826025	324826101	324826125	324826184	324826292	324826556	324826572	324826579
324826742	324826821	324826886	324827038	324827294	324827295	324827323	324827358	324827411
324827871	324828043	324828048	324828077	324828097	324828147	324828224	324828231	324828306
324828425	324828511	324828546	324828553	324828626	324828726	324828728	324828768	324828867
324828924	324828943	324829078	324829177	324829237	324829268	324829405	324829453	324829497
324829588	324829624	324829682	324829788	324829845	324829861	324829932	324829974	324830101
324830187	324830195	324830196	324830504	324830566	324830595	324830676	324830817	324831035
324831040	324831061	324831112	324831120	324831130	324831228	324831333	324831391	324831438
324831478	324831539	324831746	324831790	324831810	324831949	324831969	324832052	324832084
324832101	324832270	324832290	324832441	324832519	324832575	324832854	324832901	324832977
324833009	324833081	324833113	324833115	324833127	324833137	324833207	324833236	324833261
324833277	324833330	324833406	324833461	324833476	324833509	324833563	324833580	324833689
324833700	324833733	324833745	324833756	324833803	324833936	324834119	324834120	324834130
324834217	324834256	324834381	324834589	324834663	324834840	324834893	324834903	324834956
324835131	324835150	324835190	324835198	324835283	324835294	324835433	324835516	324835567
324835576	324835655	324835698	324835710	324835853	324836001	324836014	324836155	324836166
324836185	324836337	324836361	324836421	324836462	324836466	324836483	324836547	324836627
324836762	324836916	324837076	324837192	324837249	324837310	324837312	324837325	324837385
324837439	324837577	324837599	324837645	324837768	324837830	324837890	324837974	324838050
324838133	324838350	324838554	324838618	324838699	324838761	324838786	324838865	324838915
324838942	324838955	324839176	324839192	324839205	324839331	324839356	324839366	324839676
324839722	324839735	324839975	324840078	324840155	324840194	324840226	324840345	324840399
324840412	324840639	324840945	324841008	324841033	324841075	324841162	324841309	324841336
324841354	324841443	324841460	324841461	324841497	324841583	324841684	324841692	324841702
324841734	324841737	324841773	324841815	324842041	324842209	324842261	324842274	324842284
324842502	324842535	324842616	324842673	324842696	324842868	324842979	324843061	324843125
324843152	324843220	324843238	324843253	324843301	324843351	324843411	324843580	324843770
324843863	324844033	324844060	324844241	324844315	324844345	324844434	324844482	324844521
324844527	324844541	324844704	324844715	324844750	324844821	324845052	324845417	324845615
324845665	324845712	324845782	324845844	324845859	324845898	324846068	324846080	324846158
324846231	324846238	324846283	324846405	324846506	324846518	324846535	324846636	324846657
324846807	324846885	324847006	324847025	324847073	324847079	324847147	324847153	324847211
324847223	324847427	324847489	324847528	324847562	324847648	324847655	324847726	324847745
324847995	324848201	324848335	324848359	324848370	324848476	324848511	324848538	324848542
324848598	324848640	324848788	324848796	324848836	324848987	324849138	324849209	324849246
324849364	324849422	324849424	324849572	324849689	324849707	324849912	324849945	324849999
324850272	324850343	324850347	324850388	324850555	324850586	324850631	324850657	324850792
324850855	324850934	324850985	324851066	324851116	324851123	324851177	324851192	324851362
324851460	324851461	324851498	324851502	324851507	324851600	324851976	324852095	324852223
324852236	324852377	324852379	324852392	324852505	324852591	324852645	324852738	324852804
324852805	324852830	324852871	324852872	324852898	324853021	324853192	324853203	324853440
324853543	324853709	324853802	324853862	324853930	324853931	324854008	324854018	324854027
324854052	324854131	324854182	324854270	324854357	324854745	324854841	324854863	324854874
324854942	324854993	324855140	324855285	324855381	324855495	324855817	324855818	324855819
324855985	324856009	324856062	324856139	324856156	324856346	324856447	324856529	324856594
324856595	324856614	324856623	324856935	324857049	324857112	324857329	324857441	324857475
324857491	324857522	324857573	324857622	324857631	324857670	324857672	324857920	324858059
324858469	324858501	324858549	324858647	324858649	324858719	324858727	324858814	324859012
324859141	324859150	324859169	324859241	324859282	324859312	324859688	324859812	324859818
324859874	324859879	324859930	324860025	324860047	324860065	324860113	324860262	324860302
324860325	324860374	324860410	324860484	324860550	324860589	324860607	324860748	324861057
324861192	324861316	324861479	324861621	324861698	324861937	324862033	324862046	324862239
324862331	324862380	324862421	324862496	324862498	324862515	324862709	324862744	324862844
324862845	324862847	324862862	324862912	324862947	324862960	324863004	324863161	324863184
324863273	324863327	324863437	324863442	324863586	324863706	324863957	324863972	324864153
324864167	324864188	324864218	324864306	324864321	324864358	324864513	324864608	324864953
324865001	324865087	324865142	324865193	324865428	324865440	324865485	324865595	324865745
324865840	324866006	324866112	324866453	324866656	324866734	324866754	324867043	324867245
324867586	324867621	324867663	324867725	324867751	324867757	324867772	324867884	324867987
324868090	324868170	324868210	324868211	324868283	324868307	324868398	324868415	324868596
324868630	324868653	324868741	324868928	324868942	324868969	324868993	324869003	324869252
324869268	324869485	324869559	324869603	324869724	324869744	324869760	324869777	324869920
324869950	324870007	324870106	324870172	324870198	324870372	324870391	324870428	324870506

Continued on next page

Continued from previous page

324870636	324870671	324870764	324870793	324870794	324870855	324870986	324871036	324871089
324871134	324871187	324871280	324871322	324871478	324871492	324871542	324871590	324871675
324871713	324871798	324871828	324871867	324871893	324871917	324871934	324872288	324872431
324872435	324872539	324872577	324872681	324872773	324872830	324873081	324873112	324873160
324873176	324873221	324873253	324873344	324873432	324873537	324873608	324873683	324873690
324874220	324874235	324874326	324874363	324874407	324874549	324874625	324874836	324874837
324874885	324874943	324874995	324875058	324875070	324875085	324875216	324875283	324875323
324875427	324875457	324875540	324875603	324875788	324875831	324875987	324876018	324876153
324876181	324876242	324876527	324876584	324876645	324876731	324876743	324876782	324876950
324877173	324877658	324877709	324877711	324877729	324878171	324878275	324878330	324878342
324878682	324878793	324878854	324878855	324878894	324878898	324878948	324878975	324878999
324879066	324879076	324879185	324879260	324879328	324879469	324879651	324879688	324879702
324879948	324880010	324880070	324880097	324880111	324880116	324880193	324880230	324880278
324880432	324880493	324880507	324880518	324880903	324881185	324881303	324881306	324881349
324881624	324881627	324881669	324881723	324881790	324881808	324881823	324882061	324882201
324882259	324882301	324882327	324882426	324882460	324882659	324882735	324882853	324883021
324883089	324883208	324883256	324883281	324883288	324883401	324883411	324883676	324883694
324884002	324884023	324884058	324884092	324884148	324884351	324884532	324884601	324884710
324884793	324885177	324885204	324885337	324885438	324885446	324885487	324885554	324885725
324885753	324885785	324885891	324885906	324885955	324886256	324886281	324886312	324886654
324886664	324886914	324887070	324887100	324887107	324887167	324887277	324887291	324887443
324887482	324887558	324887563	324887645	324887688	324887739	324887791	324887860	324888005
324888097	324888177	324888244	324888250	324888307	324888324	324888370	324888372	324888400
324888414	324888416	324888550	324888559	324888674	324888741	324888789	324888795	324888958
324889082	324889142	324889313	324889469	324889482	324889539	324889672	324889690	324889709
324889829	324889947	324889954	324890071	324890158	324890191	324890206	324890307	324890315
324890316	324890423	324890524	324890540	324890560	324891110	324891160	324891406	324891624
324891765	324891778	324892102	324892127	324892326	324892578	324892595	324892737	324892745
324892813	324892823	324892836	324892933	324892969	324893175	324893443	324893447	324893644
324893688	324893919	324893928	324893994	324893995	324894066	324894171	324894376	324894521
324894565	324894734	324894784	324894815	324894969	324894994	324895106	324895277	324895283
324895323	324895379	324895684	324895743	324895764	324895919	324895977	324896096	324896097
324896118	324896144	324896242	324896292	324896368	324896503	324896618	324896622	324896636
324896645	324896699	324896702	324896745	324896882	324896896	324896901	324897140	324897164
324897230	324897281	324897284	324897307	324897368	324897454	324897486	324897779	324897792
324897816	324897827	324898033	324898220	324898385	324898393	324898492	324898505	324898506
324898525	324898606	324898634	324898664	324898871	324898939	324898993	324899023	324899042
324899113	324899116	324899210	324899355	324899809	324899852	324899859	324899900	324899943
324900060	324900069	324900252	324900276	324900291	324900382	324900395	324900484	324900566
324900831	324900888	324901078	324901118	324901185	324901526	324901533	324901559	324901594
324901765	324901884	324901977	324902119	324902208	324902239	324902322	324902390	324902399
324902622	324902805	324902848	324902896	324902920	324902946	324902947	324903093	324903205
324903544	324903669	324903722	324903761	324903768	324903865	324903902	324903950	324903989
324904075	324904147	324904149	324904200	324904215	324904237	324904283	324904571	324904606
324904628	324904722	324904735	324904761	324904783	324904877	324904957	324905095	324905159
324905217	324905384	324905406	324905409	324905471	324905525	324905653	324905664	324905855
324905859	324905886	324906067	324906072	324906126	324906215	324906277	324906360	324906384
324906521	324906547	324906593	324906641	324906779	324906804	324906816	324906946	324907049
324907154	324907161	324907365	324907462	324907549	324907553	324907670	324907739	324907762
324907767	324907972	324908342	324908490	324908558	324908620	324908662	324908665	324908763
324908986	324909011	324909061	324909071	324909105	324909140	324909148	324909154	324909209
324909308	324909380	324909554	324909596	324909604	324909605	324909655	324909697	324909703
324909889	324909912	324910117	324910189	324910207	324910220	324910323	324910488	324910509
324910617	324910654	324911056	324911073	324911084	324911085	324911150	324911281	324911380
324911617	324911654	324911687	324911730	324911825	324911826	324912008	324912058	324912142
324912180	324912183	324912195	324912251	324912352	324912387	324912410	324912586	324912732
324912741	324912759	324912890	324912946	324913122	324913213	324913233	324913316	324913339
324913568	324913669	324913819	324913888	324914088	324914158	324914202	324914385	324914411
324914570	324914817	324914841	324915037	324915052	324915059	324915141	324915279	324915297
324915495	324915548	324915556	324915577	324915613	324915634	324915988	324916047	324916116
324916128	324916162	324916218	324916241	324916310	324916346	324916407	324916439	324916445
324916531	324916571	324916656	324916767	324916789	324916963	324917016	324917034	324917116
324917157	324917192	324917264	324917375	324917410	324917423	324917471	324917591	324917594
324917715	324917885	324918038	324918083	324918101	324918144	324918293	324918415	324918556
324918614	324918649	324918697	324918755	324918871	324918957	324919056	324919074	324919105
324919163	324919176	324919252	324919279	324919326	324919356	324919384	324919425	324919446
324919508	324919537	324919635	324919897	324919928	324920026	324920114	324920167	324920267

Continued on next page

Continued from previous page

324920360	324920379	324920580	324920588	324920664	324920848	324921030	324921048	324921110
324921281	324921338	324921388	324921624	324921628	324921671	324921697	324921727	324921801
324921896	324921987	324922064	324922179	324922212	324922240	324922274	324922290	324922338
324922579	324922699	324922758	324922805	324922826	324922942	324923010	324923191	324923249
324923403	324923417	324923436	324923581	324923745	324923829	324923841	324923941	324923944
324924039	324924078	324924186	324924261	324924279	324924312	324924353	324924420	324924566
324924649	324924865	324924882	324924959	324925065	324925116	324925157	324925317	324925337
324925353	324925594	324925728	324925914	324925954	324925962	324926206	324926250	324926271
324926379	324926477	324926541	324926554	324926581	324926589	324926619	324926688	324926776
324926819	324926977	324926985	324927268	324927507	324927548	324927787	324927813	324927829
324927910	324927989	324927996	324928343	324928693	324928711	324928735	324928776	324928816
324928885	324928944	324929013	324929100	324929423	324929459	324929502	324929810	324929876
324929892	324929961	324930033	324930090	324930176	324930180	324930227	324930242	324930332
324930398	324930551	324930718	324930730	324930751	324930777	324930797	324930800	324930857
324930912	324931160	324931403	324931458	324931537	324931617	324931695	324931785	324932038
324932152	324932231	324932250	324932290	324932310	324932428	324932445	324932447	324932457
324932564	324932641	324932716	324932854	324932960	324933088	324933122	324933161	324933206
324933250	324933294	324933322	324933332	324933358	324933481	324933620	324933798	324933810
324933890	324933927	324933937	324933941	324934038	324934128	324934156	324934188	324934214
324934390	324934589	324934680	324934724	324934780	324934960	324935043	324935091	324935171
324935270	324935293	324935418	324935526	324935564	324935568	324935765	324935844	324935949
324935968	324936007	324936214	324936265	324936275	324936446	324936479	324936649	324936736
324936834	324937205	324937261	324937352	324937416	324937491	324937553	324937796	324937920
324937931	324937947	324938000	324938079	324938159	324938177	324938185	324938236	324938249
324938339	324938378	324938577	324938580	324938824	324938854	324938962	324939064	324939086
324939224	324939230	324939236	324939287	324939405	324939496	324939546	324939579	324939754
324939823	324940042	324940082	324940249	324940254	324940256	324940431	324940504	324940577
324940580	324940750	324940781	324940892	324940935	324941023	324941099	324941110	324941130
324941360	324941372	324941471	324941515	324941575	324941649	324941672	324941696	324941734
324942101	324942473	324942543	324942602	324942675	324942856	324942875	324942894	324942961
324942973	324942976	324942987	324943138	324943304	324943422	324943461	324943488	324943489
324943502	324943671	324943718	324943733	324943763	324943823	324943921	324943999	324944148
324944211	324944415	324944448	324944505	324944515	324944688	324944781	324944853	324944936
324945048	324945057	324945125	324945228	324945234	324945236	324945275	324945374	324945405
324945623	324945653	324945686	324945764	324945772	324945810	324945962	324946028	324946043
324946138	324946204	324946292	324946522	324946558	324946681	324946691	324946888	324946932
324947019	324947059	324947129	324947262	324947335	324947470	324947509	324947522	324947712
324947749	324947773	324947799	324947855	324947924	324947975	324948052	324948153	324948204
324948432	324948687	324948775	324948785	324948811	324948847	324948899	324949150	324949240
324949318	324949386	324949705	324949722	324949868	324949889	324949915	324949993	324950318
324950340	324950378	324950561	324950703	324950736	324950840	324950890	324950923	324950961
324950989	324950990	324951038	324951140	324951147	324951488	324951788	324951818	324951980
324952084	324952276	324952294	324952301	324952310	324952668	324952700	324952735	324952926
324953089	324953157	324953342	324953444	324953463	324953471	324953516	324953533	324953635
324953652	324953718	324953727	324953769	324953776	324953849	324953880	324953972	324954000
324954017	324954092	324954094	324954220	324954362	324954433	324954451	324954495	324954543
324954549	324954590	324954674	324954739	324954802	324954966	324955245	324955293	324955319
324955343	324955387	324955394	324955421	324955458	324955488	324955518	324955527	324955584
324955619	324955675	324955698	324955796	324955819	324955878	324955958	324955999	324956053
324956184	324956253	324956260	324956437	324956457	324956489	324956578	324956584	324956594
324956638	324956705	324956797	324956845	324956848	324956952	324956956	324957056	324957339
324957372	324957454	324957530	324957598	324957618	324957808	324957851	324958017	324958200
324958207	324958270	324958325	324958374	324958461	324958546	324958599	324958611	324958639
324958752	324958754	324958824	324958972	324958975	324959037	324959067	324959454	324959570
324959682	324959712	324959775	324959880	324960027	324960153	324960227	324960306	324960348
324960349	324960381	324960551	324960585	324960616	324960646	324960732	324960734	324960746
324960888	324960914	324961008	324961090	324961149	324961718	324961725	324961818	324961847
324962110	324962154	324962209	324962222	324962267	324962295	324962348	324962424	324962712
324962921	324962970	324963028	324963134	324963179	324963463	324963469	324963687	324963773
324963774	324963841	324963904	324964033	324964103	324964169	324964191	324964408	324964456
324964719	324964737	324964738	324964942	324964978	324965041	324965279	324965324	324965727
324965768	324965890	324966120	324966151	324966221	324966226	324966435	324966448	324966474
324966520	324966543	324966559	324966676	324966765	324966814	324966839	324966868	324966981
324967099	324967103	324967111	324967171	324967240	324967259	324967390	324967489	324967903
324967991	324968060	324968091	324968164	324968257	324968304	324968453	324968469	324968520
324968683	324968806	324968998	324969046	324969125	324969162	324969165	324969390	324969448
324969528	324969682	324969687	324969833	324969940	324969950	324969958	324970038	324970039

Continued on next page

Continued from previous page

324970054	324970210	324970278	324970302	324970374	324970459	324970511	324970639	324970695
324970749	324970808	324970812	324971045	324971072	324971113	324971208	324971392	324971472
324971531	324971538	324971540	324971605	324971649	324971764	324971927	324972069	324972099
324972141	324972144	324972287	324972511	324972555	324972663	324972669	324972744	324972750
324972752	324972970	324973052	324973135	324973136	324973254	324973524	324973625	324973722
324973764	324973770	324973916	324974045	324974163	324974182	324974203	324974259	324974267
324974372	324974423	324974434	324974499	324974522	324974696	324974727	324974763	324974820
324974870	324974968	324975078	324975105	324975142	324975230	324975292	324975494	324975616
324975638	324975675	324975784	324975796	324975821	324975823	324975873	324975886	324975914
324975971	324976020	324976045	324976063	324976146	324976152	324976243	324976370	324976474
324976650	324976671	324976730	324976781	324976864	324976948	324977033	324977191	324977211
324977227	324977259	324977295	324977310	324977340	324977432	324977457	324977555	324977571
324977588	324977596	324977709	324977821	324977897	324977922	324977992	324978030	324978077
324978113	324978421	324978445	324978509	324978784	324978884	324978996	324979147	324979151
324979303	324979489	324979629	324979649	324979686	324979698	324979700	324979863	324980001
324980194	324980412	324980557	324980839	324981129	324981192	324981204	324981535	324981569
324981621	324981683	324981796	324981865	324981870	324981892	324981896	324981907	324982066
324982070	324982110	324982142	324982143	324982204	324982330	324982351	324982491	324982572
324982603	324982665	324982731	324982804	324982889	324982964	324983085	324983128	324983252
324983261	324983263	324983269	324983286	324983294	324983409	324983455	324983539	324983602
324983843	324983893	324984037	324984089	324984166	324984168	324984220	324984611	324984899
324985003	324985016	324985114	324985148	324985171	324985174	324985321	324985379	324985518
324985574	324985600	324985759	324985975	324986099	324986121	324986188	324986407	324986520
324986567	324986802	324986844	324986881	324986927	324986929	324986969	324986977	324987123
324987227	324987361	324987457	324987535	324987688	324987707	324987854	324987870	324988049
324988054	324988092	324988138	324988303	324988359	324988365	324988380	324988633	324988641
324988686	324988699	324988774	324988780	324988866	324988921	324988938	324988947	324988957
324989047	324989109	324989198	324989341	324989369	324989503	324989561	324989615	324989621
324989798	324989835	324989860	324989884	324990052	324990074	324990223	324990241	324990269
324990469	324990605	324990816	324990843	324990927	324991020	324991054	324991121	324991147
324991199	324991434	324991490	324991662	324991739	324991820	324992024	324992240	324992243
324992269	324992342	324992362	324992383	324992508	324992513	324992555	324992559	324992634
324992677	324992794	324992854	324992885	324993021	324993045	324993132	324993174	324993312
324993498	324993543	324993560	324993577	324993589	324993666	324993694	324993723	324993770
324993779	324993941	324994079	324994105	324994142	324994308	324994403	324994449	324994561
324994602	324994625	324994652	324994758	324994803	324994837	324994847	324994911	324995081
324995126	324995204	324995228	324995415	324995482	324995608	324995690	324995721	324995929
324996025	324996097	324996129	324996248	324996355	324996389	324996421	324996432	324996460
324996617	324996629	324996697	324996707	324996822	324996845	324996923	324996997	324997020
324997130	324997195	324997211	324997226	324997322	324997328	324997484	324997557	324997869
324997929	324998111	324998195	324998215	324998288	324998477	324998480	324998506	324998586
324998774	324998839	324999014	324999158	324999475	324999706	324999789	324999903	324999963
324999969	325000062	325000102	325000167	325000192	325000199	325000336	325000339	325000387
325000390	325000411	325000415	325000640	325000726	325000795	325000808	325000912	325000950
325000967	325000997	325001030	325001032	325001352	325001365	325001369	325001592	325001642
325001853	325001879	325001883	325001961	325002167	325002343	325002486	325002520	325002602
325002629	325002649	325002742	325002906	325002949	325003001	325003121	325003129	325003233
325003289	325003295	325003461	325003469	325003488	325003505	325003524	325003548	325003745
325003836	325003884	325003941	325004109	325004240	325004384	325004424	325004462	325004613
325004661	325005068	325005070	325005093	325005270	325005380	325005454	325005477	325005563
325005576	325005628	325005637	325005656	325005747	325005874	325005905	325005914	325005944
325006103	325006258	325006291	325006402	325006466	325006567	325006692	325006697	325006713
325006898	325007180	325007223	325007243	325007375	325007381	325007467	325007516	325007581
325007618	325007672	325007701	325007818	325007843	325007905	325007943	325007968	325008050
325008085	325008118	325008127	325008272	325008347	325008420	325008496	325008608	325008652
325008656	325008702	325008761	325008773	325008845	325008848	325009037	325009134	325009163
325009177	325009243	325009387	325009405	325009424	325009561	325009590	325009676	325009800
325009892	325010204	325010279	325010304	325010325	325010612	325010616	325010667	325010733
325010791	325010844	325010864	325010926	325010946	325010980	325011151	325011156	325011188
325011286	325011319	325011428	325011601	325011688	325011769	325011871	325011896	325012111
325012150	325012183	325012219	325012391	325012400	325012503	325012626	325012728	325012900
325012913	325012975	325013045	325013134	325013236	325013277	325013413	325013482	325013495
325013603	325013634	325013651	325013727	325013737	325013811	325013896	325013946	325014037
325014116	325014124	325014281	325014298	325014336	325014677	325014871	325014894	325015016
325015033	325015084	325015101	325015167	325015196	325015202	325015230	325015386	325015517
325015545	325015569	325015640	325015738	325015872	325015962	325016072	325016107	325016171
325016211	325016229	325016243	325016313	325016510	325016528	325016622	325016677	325016736

Continued on next page

Continued from previous page

325016804	325016827	325016987	325017141	325017168	325017212	325017224	325017237	325017314
325017316	325017384	325017428	325017445	325017484	325017583	325017658	325017672	325017693
325017747	325017810	325018037	325018182	325018246	325018522	325018587	325018608	325018662
325018666	325018670	325018771	325018792	325018845	325018908	325018935	325018978	325019030
325019114	325019150	325019158	325019166	325019333	325019391	325019412	325019430	325019441
325019470	325019621	325019876	325019951	325019969	325020088	325020216	325020256	325020315
325020393	325020421	325020504	325020730	325020871	325020973	325021019	325021025	325021029
325021102	325021175	325021180	325021220	325021271	325021336	325021375	325021384	325021419
325021484	325021597	325021653	325021672	325021680	325021755	325021880	325021895	325021934
325021960	325021989	325021992	325022065	325022223	325022268	325022298	325022430	325022473
325022495	325022497	325022510	325022535	325022624	325022626	325022691	325022888	325022902
325022906	325022976	325023029	325023054	325023156	325023207	325023331	325023440	325023529
325023538	325023693	325023814	325023821	325024045	325024153	325024240	325024267	325024295
325024399	325024584	325024711	325024800	325024914	325024963	325025016	325025048	325025140
325025167	325025236	325025384	325025390	325025449	325025456	325025702	325025736	325025796
325025843	325025870	325025875	325025944	325025974	325026317	325026336	325026338	325026371
325026398	325026546	325026561	325026570	325026669	325026770	325026783	325027094	325027196
325027334	325027558	325027583	325027729	325027763	325028026	325028139	325028221	325028251
325028333	325028473	325028537	325028664	325028677	325028721	325028776	325028957	325029068
325029213	325029281	325029294	325029415	325029456	325029502	325029508	325029528	325029562
325029620	325029630	325029850	325029880	325029894	325030041	325030114	325030258	325030439
325030555	325030874	325031194	325031231	325031456	325031509	325031558	325031650	325031694
325031711	325031754	325031846	325031858	325031898	325031904	325031946	325032031	325032158
325032174	325032211	325032257	325032272	325032313	325032351	325032368	325032451	325032519
325032538	325032643	325032650	325032739	325032805	325032989	325033115	325033323	325033384
325033391	325033425	325033440	325033541	325033707	325033756	325033770	325033872	325034038
325034136	325034193	325034310	325034322	325034333	325034422	325034477	325034519	325034630
325034667	325034896	325034902	325034926	325034956	325035050	325035071	325035075	325035104
325035238	325035268	325035308	325035351	325035698	325035703	325035746	325035803	325035842
325035910	325035939	325036015	325036049	325036167	325036309	325036409	325036456	325036586
325036707	325036802	325036858	325036933	325037164	325037188	325037280	325037500	325037817
325037865	325038056	325038177	325038207	325038239	325038246	325038385	325038423	325038447
325038453	325038566	325038590	325038724	325038730	325038766	325038841	325038912	325038973
325038974	325039063	325039183	325039220	325039238	325039448	325039851	325040164	325040309
325040418	325040458	325040539	325040862	325040865	325040904	325040974	325040987	325041068
325041178	325041403	325041440	325041527	325041594	325041632	325041638	325041693	325041841
325041860	325041866	325041896	325041921	325041971	325042002	325042115	325042130	325042178
325042237	325042526	325042541	325042578	325042590	325042637	325042750	325042822	325042878
325042938	325042944	325042999	325043093	325043126	325043219	325043277	325043331	325043464
325043558	325043580	325043748	325043765	325043986	325044021	325044229	325044452	325044453
325044520	325044764	325044774	325044777	325044843	325044910	325045139	325045149	325045175
325045296	325045442	325045464	325045538	325045613	325045675	325045705	325045926	325045986
325046085	325046108	325046140	325046172	325046314	325046344	325046408	325046599	325046736
325046786	325046881	325046890	325047085	325047150	325047175	325047417	325047550	325047586
325047590	325047732	325047889	325047943	325047951	325048023	325048081	325048126	325048159
325048201	325048226	325048363	325048367	325048371	325048395	325048440	325048471	325048498
325048606	325048701	325048794	325048802	325048833	325048896	325048937	325048981	325049049
325049060	325049181	325049304	325049325	325049360	325049384	325049400	325049403	325049495
325049502	325049585	325049592	325049599	325049835	325049847	325049873	325050148	325050317
325050364	325050394	325050408	325050533	325050551	325050624	325050661	325050736	325050813
325050900	325051079	325051090	325051109	325051184	325051187	325051212	325051482	325051555
325051558	325051590	325051593	325051676	325051798	325051997	325052003	325052080	325052123
325052146	325052160	325052171	325052346	325052351	325052440	325052443	325052458	325052463
325052476	325052486	325052569	325052691	325052734	325052864	325053026	325053083	325053098
325053135	325053375	325053490	325053516	325053587	325053739	325053741	325053836	325053874
325053972	325054009	325054372	325054375	325054430	325054436	325054534	325054706	325054755
325054870	325054873	325054901	325054943	325055046	325055123	325055166	325055293	325055376
325055440	325055492	325055502	325055532	325055543	325055585	325055642	325055807	325055817
325055871	325055927	325055964	325056300	325056385	325056435	325056569	325056593	325056673
325056881	325056931	325057062	325057208	325057216	325057217	325057230	325057242	325057365
325057382	325057404	325057482	325057520	325057877	325057954	325058003	325058042	325058347
325058409	325058426	325058465	325058544	325058648	325058678	325058745	325058816	325058829
325058883	325058952	325058966	325059002	325059011	325059017	325059099	325059112	325059417
325059583	325059647	325059649	325059720	325059809	325060008	325060087	325060165	325060236
325060471	325060484	325060628	325060734	325060875	325060928	325060980	325061046	325061099
325061202	325061252	325061361	325061366	325061497	325061537	325061585	325061613	325061669
325061736	325061847	325061848	325061899	325061928	325062090	325062144	325062191	325062293

Continued on next page

Continued from previous page

325062327	325062363	325062635	325062738	325062758	325062934	325062969	325062976	325063070
325063087	325063110	325063255	325063364	325063365	325063484	325063510	325063563	325063592
325063601	325063609	325063618	325063637	325063663	325063868	325063994	325064312	325064352
325064363	325064389	325064416	325064513	325064696	325064795	325064953	325065024	325065276
325065282	325065292	325065346	325065428	325065437	325065452	325065470	325065533	325065546
325065653	325065896	325065909	325065967	325065999	325066015	325066339	325066447	325066481
325066522	325066535	325066545	325066696	325066751	325066876	325066940	325066963	325066968
325066998	325067052	325067139	325067190	325067284	325067362	325067397	325067419	325067441
325067457	325067743	325068050	325068070	325068193	325068213	325068215	325068406	325068413
325068512	325068529	325068580	325068659	325068714	325068811	325068818	325068931	325068960
325068980	325069009	325069099	325069104	325069133	325069166	325069420	325069512	325069556
325069663	325069726	325069800	325069849	325069884	325069899	325070235	325070275	325070276
325070472	325070515	325070520	325070570	325070616	325070680	325070708	325070713	325070731
325070780	325070862	325070907	325070926	325071081	325071187	325071437	325071451	325071471
325071520	325071591	325071819	325071861	325071881	325071922	325072108	325072293	325072476
325072610	325072737	325072776	325072932	325072938	325073067	325073119	325073246	325073345
325073391	325073526	325073625	325073817	325073906	325073968	325074000	325074141	325074273
325074298	325074302	325074429	325074528	325074608	325074617	325074746	325074756	325074789
325074949	325075028	325075068	325075135	325075140	325075599	325075634	325075636	325075643
325075694	325075712	325075826	325075992	325075994	325076038	325076391	325076481	325076509
325076525	325076531	325076755	325076799	325076828	325076888	325076933	325077045	325077103
325077116	325077166	325077187	325077283	325077434	325077484	325077497	325077513	325077525
325077581	325077641	325077708	325077781	325077819	325077821	325078057	325078136	325078291
325078342	325078720	325078844	325078861	325078912	325079058	325079322	325079444	325079530
325079652	325079672	325079781	325079794	325079841	325079958	325080059	325080091	325080105
325080129	325080312	325080315	325080328	325080453	325080514	325080520	325080576	325080619
325080631	325080636	325080686	325080871	325080874	325080932	325081051	325081060	325081078
325081283	325081301	325081340	325081415	325081448	325081494	325081592	325081657	325081685
325081743	325081761	325081798	325081808	325081810	325081875	325081994	325082021	325082026
325082286	325082313	325082423	325082459	325082613	325082660	325082705	325082724	325082750
325082844	325082974	325082991	325083031	325083094	325083095	325083157	325083322	325083435
325083468	325083633	325083672	325083704	325083768	325083775	325083993	325084080	325084093
325084151	325084158	325084238	325084266	325084295	325084333	325084378	325084488	325084509
325084521	325084540	325084656	325084674	325084677	325084719	325084854	325084887	325084989
325085135	325085157	325085158	325085323	325085350	325085471	325085577	325085623	325085765
325085842	325085847	325085888	325085894	325085938	325085986	325086031	325086102	325086244
325086251	325086281	325086338	325086362	325086451	325086461	325086497	325086604	325086605
325086680	325086687	325086755	325086834	325086865	325086913	325086977	325087108	325087137
325087140	325087283	325087304	325087324	325087330	325087332	325087347	325087354	325087425
325087652	325087672	325087719	325087753	325087760	325087978	325088073	325088104	325088120
325088139	325088231	325088240	325088427	325088459	325088636	325088719	325088774	325088894
325088899	325089015	325089088	325089112	325089219	325089326	325089386	325089476	325089560
325089570	325089589	325089608	325089714	325089775	325089794	325089828	325089891	325089978
325090017	325090018	325090129	325090219	325090236	325090344	325090375	325090406	325090527
325090637	325090829	325090831	325090910	325091055	325091067	325091091	325091104	325091205
325091300	325091360	325091368	325091482	325091483	325091587	325091606	325091789	325091807
325091882	325091939	325092041	325092064	325092084	325092109	325092146	325092159	325092249
325092316	325092447	325092473	325092639	325092655	325092663	325092680	325092687	325092898
325092911	325092925	325092929	325093004	325093018	325093052	325093079	325093095	325093120
325093191	325093333	325093411	325093422	325093441	325093466	325093500	325093512	325093706
325093729	325093871	325093890	325093891	325093961	325094040	325094244	325094259	325094287
325094325	325094357	325094387	325094456	325094485	325094533	325094576	325094773	325094814
325094827	325094896	325095080	325095218	325095226	325095273	325095280	325095348	325095361
325095402	325095507	325095509	325095554	325095572	325095616	325095645	325095769	325095773
325095815	325095817	325095859	325095900	325095932	325095944	325096093	325096149	325096212
325096314	325096364	325096456	325096774	325096934	325097011	325097025	325097119	325097173
325097318	325097530	325097535	325097578	325097708	325097859	325097941	325098005	325098316
325098472	325098509	325098596	325098633	325098678	325098803	325098812	325098920	325098942
325098944	325099123	325099245	325099249	325099327	325099370	325099427	325099497	325099566
325099699	325099893	325099915	325099941	325100039	325100106	325100147	325100305	325100336
325100379	325100392	325100418	325100460	325100499	325100616	325100627	325100700	325100756
325100765	325100777	325100813	325100854	325100869	325101115	325101122	325101196	325101450
325101515	325101659	325101689	325101786	325101805	325101828	325101846	325101912	325101945
325101975	325102195	325102223	325102452	325102463	325102514	325102521	325102556	325102571
325102683	325102817	325102854	325102879	325102932	325103047	325103109	325103181	325103269
325103298	325103331	325103367	325103368	325103421	325103621	325103662	325103925	325104107
325104157	325104259	325104337	325104407	325104453	325104515	325104516	325104546	325104571

Continued on next page

Continued from previous page

325104653	325104681	325104682	325104734	325104758	325104761	325104795	325104948	325104989
325105016	325105055	325105104	325105214	325105274	325105515	325105590	325105611	325105683
325105763	325106064	325106096	325106115	325106117	325106161	325106247	325106277	325106328
325106472	325106474	325106535	325106680	325106776	325106853	325107079	325107083	325107109
325107125	325107147	325107160	325107246	325107328	325107442	325107551	325107576	325107598
325107633	325107668	325107670	325107752	325107874	325108015	325108038	325108174	325108399
325108586	325108686	325108815	325108892	325108928	325108959	325108973	325109124	325109150
325109151	325109152	325109304	325109338	325109418	325109491	325109556	325109623	325109772
325110048	325110106	325110190	325110209	325110225	325110376	325110409	325110538	325110593
325110628	325110630	325110714	325110719	325110858	325111067	325111181	325111219	325111244
325111406	325111411	325111455	325111592	325111810	325111879	325112016	325112218	325112405
325112454	325112465	325112472	325112485	325112499	325112602	325112625	325112631	325112669
325112706	325112714	325112719	325112741	325112968	325113053	325113180	325113259	325113387
325113485	325113489	325113589	325113744	325113756	325113779	325113806	325113920	325113948
325114013	325114091	325114189	325114193	325114236	325114261	325114287	325114451	325114491
325114518	325114610	325114679	325114783	325114836	325114840	325114849	325115089	325115198
325115418	325115485	325115509	325115511	325115723	325115727	325115729	325115753	325115778
325115951	325115971	325115979	325116266	325116323	325116343	325116392	325116401	325116464
325116466	325116522	325116540	325116562	325116617	325116683	325116721	325116745	325116766
325116859	325116894	325116906	325116907	325116951	325117007	325117023	325117031	325117121
325117297	325117353	325117461	325117496	325117547	325117554	325117678	325117818	325117875
325118313	325118434	325118526	325118793	325118981	325119038	325119039	325119199	325119223
325119271	325119347	325119475	325119535	325119540	325119559	325119631	325119671	325119712
325119722	325119725	325119795	325119847	325119866	325119967	325120004	325120097	325120204
325120383	325120400	325120417	325120455	325120473	325120768	325120843	325121035	325121110
325121227	325121289	325121415	325121739	325121762	325121793	325121804	325121807	325121837
325121915	325122131	325122426	325122581	325122840	325123036	325123150	325123158	325123263
325123345	325123359	325123545	325123566	325123667	325123690	325123728	325123824	325123826
325123891	325123920	325123949	325123986	325124172	325124244	325124277	325124294	325124307
325124315	325124436	325124498	325124605	325124691	325124695	325124707	325124715	325124974
325125261	325125317	325125376	325125525	325125581	325125708	325125912	325126037	325126110
325126139	325126186	325126229	325126261	325126553	325126645	325126656	325126667	325126682
325126804	325126943	325126954	325126994	325127005	325127123	325127154	325127257	325127355
325127444	325127562	325127592	325127700	325127706	325127724	325127736	325127805	325127834
325127864	325128013	325128017	325128035	325128045	325128118	325128127	325128219	325128225
325128345	325128416	325128455	325128665	325128700	325128705	325128720	325128763	325128868
325129100	325129139	325129140	325129213	325129354	325129509	325129525	325129528	325129569
325129693	325129749	325129821	325129902	325129945	325129973	325130112	325130305	325130379
325130653	325130911	325130942	325130970	325131129	325131173	325131196	325131339	325131422
325131472	325131624	325131666	325131784	325131791	325131853	325131868	325131931	325132019
325132249	325132313	325132385	325132449	325132562	325132649	325132769	325132797	325132798
325132808	325132861	325132902	325133033	325133050	325133089	325133153	325133268	325133270
325133354	325133478	325133531	325133607	325133724	325133760	325133783	325133807	325133815
325133826	325133996	325134196	325134238	325134254	325134259	325134299	325134318	325134334
325134354	325134355	325134379	325134550	325134609	325134783	325134874	325134893	325134980
325135003	325135219	325135227	325135285	325135287	325135296	325135355	325135552	325135568
325135656	325135746	325135747	325135775	325135782	325135932	325135987	325136000	325136045
325136159	325136183	325136235	325136340	325136406	325136461	325136557	325136805	325136982
325136987	325136999	325137048	325137075	325137168	325137194	325137371	325137396	325137442
325137531	325137652	325137670	325137674	325137881	325137891	325138109	325138116	325138287
325138304	325138326	325138331	325138544	325138559	325138648	325138702	325138798	325138831
325138870	325138876	325138902	325138939	325138975	325139114	325139175	325139522	325139539
325139564	325139644	325139666	325139875	325139942	325140114	325140130	325140131	325140311
325140348	325140359	325140429	325140437	325140527	325140530	325140559	325140677	325140745
325140829	325140872	325140901	325140921	325141086	325141154	325141185	325141193	325141261
325141598	325141605	325141777	325141836	325141858	325141871	325141893	325141933	325142173
325142271	325142324	325142384	325142498	325142501	325142504	325142549	325142554	325142598
325142733	325142742	325142809	325142947	325143052	325143271	325143486	325143556	325143575
325143645	325143656	325143834	325143853	325143905	325143980	325143985	325144034	325144042
325144090	325144096	325144249	325144277	325144280	325144395	325144408	325144519	325144559
325144634	325144640	325144725	325144768	325144839	325144995	325145119	325145213	325146044
325146049	325146427	325146581	325146584	325146586	325146696	325146699	325146702	325146838
325146960	325147053	325147087	325147123	325147161	325147312	325147340	325147521	325147669
325147746	325147759	325147915	325147955	325148059	325148172	325148235	325148246	325148290
325148377	325148476	325148549	325148628	325148731	325149018	325149054	325149119	325149237
325149299	325149364	325149401	325149496	325149516	325149632	325149716	325149806	325149873
325149946	325149954	325149977	325150021	325150069	325150083	325150089	325150140	325150146

Continued on next page

Continued from previous page

325150241	325150323	325150440	325150611	325150627	325150852	325150971	325151051	325151133
325151155	325151286	325151312	325151519	325151527	325151642	325151655	325151721	325151775
325151810	325151995	325152007	325152134	325152158	325152245	325152309	325152387	325152404
325152472	325152477	325152623	325152633	325152963	325153083	325153144	325153181	325153188
325153261	325153289	325153391	325153435	325153497	325153511	325153587	325153622	325153701
325153727	325153744	325153773	325153774	325153916	325153952	325153980	325154011	325154176
325154213	325154345	325154393	325154425	325154520	325154522	325154616	325154759	325154803
325154899	325154906	325154935	325154942	325154974	325155169	325155176	325155199	325155365
325155446	325155592	325155769	325155842	325155883	325155893	325155972	325156022	325156045
325156136	325156156	325156358	325156443	325156535	325156596	325156674	325156746	325156788
325156861	325156999	325157085	325157096	325157174	325157247	325157273	325157313	325157613
325157723	325157775	325157810	325157812	325157833	325157848	325157894	325157943	325157963
325157986	325158154	325158263	325158312	325158332	325158459	325158500	325158514	325158551
325158617	325158673	325158728	325158743	325159099	325159225	325159328	325159343	325159406
325159409	325159425	325159466	325159467	325159532	325159598	325159627	325159658	325159689
325159794	325159893	325160005	325160043	325160168	325160183	325160252	325160360	325160405
325160467	325160543	325160653	325160834	325160878	325160913	325161065	325161163	325161183
325161212	325161430	325161510	325161638	325161749	325161800	325162085	325162124	325162274
325162282	325162409	325162532	325162620	325162656	325162763	325162784	325162843	325162849
325162995	325163161	325163572	325163596	325163795	325163929	325164018	325164119	325164251
325164453	325164559	325164675	325165177	325165210	325165428	325165515	325165772	325165782
325165787	325165887	325165890	325166102	325166164	325166322	325166402	325166407	325166423
325166526	325166555	325166714	325166863	325166978	325167056	325167133	325167136	325167142
325167250	325167422	325167564	325167590	325167749	325167762	325167834	325167849	325167885
325167903	325167966	325168084	325168121	325168138	325168200	325168448	325168454	325168490
325168509	325168780	325168784	325168789	325168808	325168892	325169141	325169169	325169172
325169215	325169231	325169315	325169453	325169499	325169540	325169705	325169725	325169745
325169808	325169950	325169974	325170107	325170143	325170280	325170332	325170337	325170663
325170698	325170736	325170950	325171030	325171077	325171219	325171238	325171352	325171440
325171577	325171628	325171670	325171672	325171698	325171843	325171950	325172157	325172206
325172258	325172446	325172642	325172668	325172669	325172696	325172839	325172865	325172889
325172899	325172908	325173020	325173053	325173092	325173403	325173584	325173640	325173658
325173701	325173708	325174196	325174273	325174411	325174498	325174544	325174651	325174713
325174775	325174901	325175100	325175106	325175158	325175375	325175393	325175519	325175597
325175803	325175816	325175845	325175922	325175990	325176005	325176180	325176298	325176310
325176313	325176449	325176546	325176646	325176656	325176692	325176696	325176779	325176811
325176859	325177124	325177133	325177135	325177184	325177239	325177348	325177547	325177602
325177661	325177744	325177808	325177819	325177846	325177972	325178025	325178047	325178203
325178223	325178326	325178420	325178491	325178496	325178535	325178541	325178892	325178909
325178959	325179008	325179020	325179075	325179118	325179227	325179349	325179390	325179424
325179602	325179735	325179760	325179778	325179952	325180021	325180142	325180192	325180195
325180239	325180287	325180310	325180324	325180469	325180503	325180515	325180604	325180680
325180802	325180805	325180836	325180847	325181015	325181078	325181120	325181166	325181297
325181303	325181362	325181370	325181490	325181525	325181550	325181585	325181598	325181668
325181673	325181899	325181984	325181989	325181997	325182007	325182150	325182189	325182203
325182226	325182264	325182323	325182369	325182396	325182403	325182437	325182451	325182619
325182635	325182652	325182733	325182807	325182901	325183186	325183189	325183330	325183372
325183403	325183471	325183556	325183557	325183624	325183736	325183847	325183858	325184078
325184196	325184316	325184387	325184474	325184493	325184543	325184567	325184593	325184628
325184637	325184895	325184918	325185069	325185120	325185178	325185250	325185274	325185320
325185479	325185520	325185556	325185641	325185683	325185819	325185871	325185924	325185934
325186065	325186081	325186090	325186113	325186167	325186235	325186260	325186285	325186388
325186430	325186544	325186545	325186607	325186905	325186940	325187181	325187192	325187196
325187197	325187205	325187246	325187273	325187326	325187420	325187502	325187558	325187652
325187722	325187893	325187900	325187911	325188095	325188099	325188121	325188191	325188431
325188538	325188565	325188569	325188720	325188731	325188980	325189046	325189074	325189183
325189193	325189387	325189496	325189563	325189599	325189604	325189907	325189909	325189917
325190020	325190034	325190135	325190260	325190274	325190420	325190558	325190565	325190600
325190636	325190729	325190811	325190961	325190997	325191101	325191160	325191220	325191227
325191239	325191269	325191318	325191359	325191378	325191423	325191564	325191611	325191681
325191724	325191928	325191962	325192020	325192137	325192226	325192339	325192374	325192418
325192450	325192480	325192642	325192900	325192902	325192942	325192996	325193346	325193530
325193697	325193734	325193761	325194087	325194222	325194314	325194441	325194526	325194528
325194665	325194781	325194981	325195114	325195291	325195337	325195362	325195393	325195418
325195448	325195482	325195631	325195714	325195894	325196016	325196359	325196423	325196518
325196534	325196683	325196692	325196759	325196815	325196850	325197052	325197143	325197156
325197239	325197246	325197279	325197355	325197377	325197604	325197709	325197805	325197967

Continued on next page

Continued from previous page

325197987	325198072	325198384	325198390	325198460	325198474	325198494	325198528	325198706
325198896	325198899	325199021	325199098	325199109	325199143	325199268	325199543	325199558
325199579	325199633	325199644	325199672	325199723	325199754	325199826	325199889	325199912
325200070	325200112	325200181	325200243	325200299	325200440	325200468	325200479	325200528
325200653	325200697	325200797	325201001	325201066	325201137	325201531	325201661	325201667
325201845	325202166	325202199	325202487	325202532	325202596	325202660	325202665	325202688
325202897	325203169	325203175	325203480	325203500	325203611	325203911	325203944	325204291
325204341	325204390	325204399	325204582	325204809	325204875	325204939	325204996	325205011
325205072	325205219	325205251	325205317	325205374	325205402	325205435	325205478	325205494
325205633	325205700	325205718	325205797	325206000	325206062	325206547	325206551	325206631
325206827	325206907	325206918	325206922	325206939	325206956	325207018	325207086	325207217
325207295	325207389	325207475	325207578	325207593	325207618	325207900	325207949	325207950
325207969	325207971	325208002	325208044	325208105	325208148	325208245	325208309	325208419
325208431	325208685	325208757	325208772	325208796	325208839	325208879	325208899	325208957
325209092	325209149	325209369	325209667	325209698	325209771	325209896	325209922	325209953
325209966	325210131	325210208	325210240	325210280	325210346	325210369	325210407	325210459
325210598	325210611	325210666	325210712	325210738	325210974	325211024	325211148	325211247
325211442	325211495	325211516	325211548	325211628	325211659	325211668	325211839	325211931
325211997	325212231	325212237	325212358	325212365	325212403	325212486	325212496	325212956
325213088	325213398	325213411	325213512	325213555	325213646	325213729	325213738	325213760
325213848	325213884	325213886	325213902	325214039	325214084	325214088	325214346	325214353
325214429	325214434	325214647	325214678	325214773	325214842	325214854	325214938	325215019
325215034	325215210	325215215	325215235	325215276	325215309	325215490	325215498	325215556
325215570	325215649	325215692	325215965	325216034	325216092	325216196	325216326	325216374
325216545	325216629	325216675	325216755	325216764	325216783	325216786	325216857	325216972
325217202	325217211	325217220	325217289	325217312	325217331	325217401	325217412	325217532
325217658	325217685	325217694	325217730	325217786	325217830	325217871	325217920	325218052
325218055	325218132	325218162	325218186	325218253	325218283	325218299	325218404	325218411
325218479	325218492	325218700	325218763	325218774	325218979	325218991	325218996	325219004
325219076	325219211	325219232	325219235	325219610	325219642	325219868	325220079	325220126
325220132	325220284	325220326	325220337	325220347	325220357	325220524	325220554	325220638
325220693	325220800	325220847	325220901	325220921	325220950	325220995	325221004	325221091
325221119	325221226	325221287	325221297	325221476	325221527	325221567	325221641	325221910
325222072	325222094	325222347	325222370	325222399	325222437	325222458	325222487	325222537
325222649	325222673	325222728	325222744	325222771	325222869	325222910	325222971	325222988
325223144	325223164	325223189	325223226	325223234	325223270	325223410	325223423	325223428
325223495	325223517	325223565	325223835	325223870	325223990	325224112	325224225	325224318
325224324	325224365	325224552	325224558	325224618	325224722	325224835	325224942	325225002
325225206	325225258	325225275	325225482	325225502	325225528	325225595	325225765	325225836
325225847	325225951	325225963	325226074	325226389	325226436	325226462	325226469	325226499
325226610	325226636	325226752	325226840	325226933	325227001	325227119	325227322	325227358
325227438	325227539	325227818	325227851	325227903	325227917	325227934	325228010	325228018
325228048	325228059	325228194	325228253	325228438	325228444	325228453	325228458	325228541
325228546	325228637	325228706	325228960	325229044	325229052	325229217	325229364	325229374
325229427	325229452	325229516	325230002	325230013	325230014	325230193	325230223	325230227
325230451	325230480	325230495	325230542	325230616	325230710	325230730	325230802	325230847
325230893	325231008	325231270	325231291	325231401	325231478	325231660	325231787	325231904
325231917	325232032	325232064	325232113	325232144	325232193	325232210	325232215	325232257
325232266	325232348	325232380	325232536	325232575	325232788	325232835	325232891	325232964
325233253	325233325	325233338	325233413	325233507	325233545	325233597	325233601	325233611
325233647	325233759	325233762	325233844	325233889	325233902	325233913	325233946	325234125
325234249	325234251	325234254	325234384	325234438	325234537	325234546	325234741	325234833
325234887	325234904	325234951	325234978	325234997	325235225	325235287	325235316	325235462
325235483	325235516	325235668	325235782	325236077	325236084	325236231	325236303	325236486
325236487	325236498	325236540	325236818	325236848	325236852	325236944	325237200	325237252
325237426	325237450	325237543	325237595	325237647	325237837	325238014	325238071	325238215
325238224	325238301	325238347	325238405	325238417	325238476	325238536	325238539	325238601
325238610	325238666	325238878	325238886	325238912	325238919	325238979	325238990	325239011
325239047	325239065	325239228	325239338	325239421	325239430	325239431	325239471	325239531
325239564	325239697	325239798	325239845	325239927	325239994	325240164	325240338	325240382
325240394	325240461	325240488	325240531	325240796	325240919	325241072	325241086	325241138
325241166	325241241	325241417	325241605	325241632	325241863	325241864	325242153	325242383
325242403	325242501	325242521	325242574	325242682	325242691	325242769	325242777	325242782
325242817	325242867	325242887	325243003	325243022	325243065	325243126	325243162	325243215
325243227	325243302	325243342	325243494	325243587	325243698	325243908	325243926	325244355
325244490	325244660	325244680	325244798	325244876	325244882	325244918	325245109	325245221
325245407	325245555	325245557	325245794	325245948	325245967	325246114	325246301	325246430

Continued on next page

Continued from previous page

325246475	325246765	325246774	325246838	325246951	325246990	325247185	325247244	325247260
325247348	325247416	325247439	325247702	325247734	325247790	325247798	325247861	325247926
325247994	325248025	325248054	325248073	325248160	325248209	325248371	325248511	325248625
325248677	325248727	325248730	325248745	325248816	325248990	325248999	325249025	325249045
325249145	325249223	325249248	325249302	325249348	325249445	325249502	325249527	325249710
325249827	325250129	325250132	325250224	325250225	325250242	325250331	325250450	325250465
325250527	325250555	325250579	325250630	325250725	325250949	325250993	325251048	325251065
325251131	325251185	325251224	325251261	325251282	325251454	325251502	325251548	325251559
325251854	325252259	325252377	325252399	325252529	325252538	325252539	325252691	325252859
325252873	325252956	325253138	325253153	325253198	325253268	325253446	325253543	325253760
325253762	325253767	325254002	325254033	325254106	325254154	325254341	325254399	325254436
325254626	325254705	325254832	325254876	325254955	325255006	325255052	325255074	325255124
325255140	325255656	325255666	325255753	325255758	325255777	325255809	325255865	325255877
325255918	325256058	325256171	325256252	325256306	325256543	325256555	325256626	325256638
325256725	325256787	325256894	325256899	325257038	325257061	325257083	325257122	325257457
325257461	325257463	325257837	325257969	325257975	325258131	325258241	325258246	325258325
325258341	325258391	325258405	325258472	325258532	325258609	325258627	325258733	325258781
325258783	325258811	325258861	325259027	325259100	325259111	325259118	325259127	325259173
325259177	325259191	325259240	325259309	325259385	325259587	325259596	325259648	325259813
325259818	325260045	325260219	325260347	325260375	325260435	325260438	325260524	325260541
325260771	325260815	325260836	325260839	325260885	325260976	325261192	325261329	325261456
325261483	325261716	325261849	325261874	325261891	325261927	325262003	325262010	325262019
325262103	325262107	325262133	325262142	325262280	325262365	325262393	325262518	325262525
325262535	325262878	325262883	325263028	325263159	325263190	325263208	325263247	325263291
325263312	325263314	325263390	325263426	325263450	325263496	325263690	325263819	325263853
325263854	325263891	325263934	325264084	325264240	325264390	325264410	325264501	325264571
325264881	325264904	325264987	325265323	325265365	325265420	325265549	325265628	325265671
325265694	325265708	325265763	325265875	325265936	325265994	325266171	325266175	325266551
325266968	325266977	325266980	325266995	325267007	325267071	325267191	325267204	325267252
325267292	325267299	325267331	325267435	325267526	325267528	325267708	325267720	325267891
325267923	325267994	325268153	325268173	325268211	325268277	325268412	325268437	325268511
325268588	325268679	325268738	325268913	325269012	325269153	325269178	325269208	325269415
325269448	325269623	325269660	325269662	325269724	325269778	325269904	325269915	325270051
325270166	325270422	325270478	325270569	325270882	325270885	325270902	325270924	325271091
325271148	325271165	325271263	325271312	325271323	325271487	325271494	325271505	325271614
325271829	325271872	325271896	325272253	325272321	325272420	325272442	325272542	325272595
325272673	325272865	325272897	325273022	325273205	325273224	325273469	325273555	325273614
325273742	325273753	325273765	325273805	325273851	325274210	325274215	325274231	325274258
325274298	325274330	325274418	325274510	325274618	325274704	325274706	325274851	325274919
325274937	325275002	325275187	325275233	325275303	325275309	325275448	325275456	325275491
325275633	325275660	325275694	325275831	325275940	325275985	325276167	325276188	325276246
325276298	325276376	325276477	325276654	325276722	325276784	325276786	325276812	325276857
325277067	325277127	325277271	325277450	325277487	325277524	325277528	325277548	325277576
325277765	325277803	325277806	325277852	325277874	325278084	325278085	325278103	325278142
325278304	325278419	325278455	325278468	325278480	325278819	325278854	325278859	325278938
325279006	325279193	325279195	325279201	325279225	325279312	325279456	325279461	325279508
325279520	325279684	325279835	325279978	325280042	325280052	325280069	325280209	325280251
325280437	325280666	325280688	325280799	325280854	325280967	325280984	325281123	325281333
325281423	325281437	325281584	325281684	325281913	325281925	325282019	325282055	325282122
325282123	325282134	325282340	325282451	325282539	325282591	325282712	325282742	325282774
325282959	325283149	325283174	325283351	325283354	325283616	325283802	325283841	325283842
325283847	325283899	325283946	325284072	325284101	325284155	325284284	325284295	325284405
325284467	325284497	325284626	325284634	325284642	325284657	325284672	325284750	325284759
325284766	325284839	325284862	325284864	325284946	325285122	325285187	325285245	325285297
325285407	325285485	325285779	325285857	325285912	325285986	325286014	325286015	325286286
325286344	325286386	325286520	325286532	325286655	325286662	325286722	325286904	325287048
325287076	325287098	325287217	325287299	325287501	325287583	325287853	325287943	325287974
325288161	325288174	325288194	325288309	325288552	325288608	325288615	325288623	325288661
325288718	325288952	325289328	325289390	325289400	325289454	325289571	325289700	325289874
325289983	325290308	325290351	325290395	325290716	325290774	325290879	325290893	325290914
325290978	325291009	325291013	325291123	325291144	325291175	325291256	325291312	325291347
325291485	325291489	325291563	325291599	325291725	325291903	325292025	325292146	325292149
325292225	325292238	325292365	325292398	325292455	325292459	325292558	325292609	325292926
325293233	325293295	325293299	325293502	325293577	325293615	325293644	325293652	325293694
325293703	325293717	325293997	325294012	325294225	325294298	325294338	325294375	325294397
325294407	325294423	325294563	325294670	325294720	325294732	325294936	325295193	325295243
325295301	325295341	325295508	325295605	325295622	325295831	325295883	325295893	325295903

Continued on next page

Continued from previous page

325295928	325296062	325296285	325296337	325296356	325296502	325296575	325296591	325296623
325296814	325296955	325296996	325297032	325297070	325297238	325297240	325297649	325297695
325297931	325297945	325297964	325297976	325298044	325298076	325298133	325298159	325298496
325298579	325298595	325298632	325298638	325298755	325298799	325298880	325299097	325299127
325299153	325299257	325299321	325299408	325299572	325299686	325299696	325299768	325299905
325299923	325299981	325300018	325300136	325300357	325300380	325300399	325300526	325300558
325300742	325300743	325300784	325300839	325300844	325300930	325300954	325301036	325301128
325301241	325301255	325301414	325301480	325301570	325301609	325301632	325301661	325301703
325301716	325301765	325301797	325302131	325302457	325302484	325302509	325302887	325302910
325302932	325303080	325303116	325303247	325303305	325303453	325303455	325303495	325303518
325303755	325303841	325303847	325303919	325304219	325304238	325304259	325304273	325304300
325304303	325304417	325304458	325304562	325304586	325304673	325304675	325304730	325305037
325305048	325305058	325305269	325305271	325305313	325305695	325305877	325305914	325305936
325305977	325306372	325306526	325306730	325306799	325306819	325306835	325306852	325307138
325307173	325307176	325307205	325307386	325307516	325307679	325307760	325307766	325307770
325307873	325308076	325308219	325308360	325308376	325308475	325308550	325308572	325308620
325308899	325308977	325309017	325309078	325309142	325309149	325309213	325309286	325309368
325309517	325309547	325309675	325309743	325309897	325310042	325310048	325310146	325310221
325310301	325310321	325310394	325310517	325310585	325310734	325310784	325310820	325310871
325310891	325310907	325310929	325310952	325311048	325311052	325311069	325311148	325311397
325311424	325311436	325311438	325311444	325311456	325311810	325311838	325312108	325312150
325312181	325312280	325312393	325312438	325312472	325312474	325312577	325312593	325312602
325312623	325312804	325312844	325312855	325312876	325312968	325312974	325312975	325313394
325313624	325313805	325313814	325313901	325314086	325314099	325314111	325314128	325314142
325314398	325314428	325314498	325314603	325314665	325314738	325314874	325314904	325314917
325315048	325315097	325315262	325315282	325315360	325315364	325315445	325315581	325315591
325315710	325315770	325315803	325316043	325316088	325316114	325316177	325316253	325316333
325316604	325316734	325316761	325316779	325317052	325317072	325317112	325317126	325317139
325317239	325317292	325317477	325317558	325317593	325317628	325317709	325317849	325317875
325317960	325318005	325318131	325318151	325318211	325318315	325318361	325318378	325318463
325318468	325318663	325318721	325318783	325318813	325318842	325318863	325319422	325319437
325319448	325319532	325319570	325319812	325319930	325319938	325319997	325320028	325320035
325320099	325320110	325320117	325320253	325320511	325320631	325320688	325320722	325320746
325320798	325320800	325320833	325320882	325320969	325320991	325321022	325321311	325321486
325321593	325321630	325321652	325321669	325321694	325321735	325321755	325321901	325322052
325322116	325322140	325322235	325322289	325322512	325322520	325322590	325322753	325322814
325322887	325322897	325322921	325322993	325322994	325323105	325323331	325323380	325323506
325323510	325323564	325323586	325323605	325323654	325323884	325323898	325323967	325324117
325324168	325324239	325324266	325324368	325324386	325324535	325324655	325324693	325324704
325324756	325324808	325324930	325324993	325325002	325325007	325325014	325325016	325325047
325325057	325325117	325325236	325325429	325325497	325325503	325325636	325325644	325325727
325325729	325325809	325325850	325325916	325325970	325326059	325326150	325326151	325326222
325326239	325326267	325326337	325326429	325326457	325326514	325326542	325326712	325326720
325326798	325326802	325326805	325326814	325326993	325327011	325327024	325327032	325327043
325327101	325327139	325327172	325327236	325327247	325327330	325327648	325327671	325327673
325327725	325327743	325327825	325327929	325327950	325327990	325328111	325328192	325328271
325328314	325328318	325328320	325328328	325328393	325328451	325328498	325328513	325328558
325328564	325328602	325328699	325328735	325328880	325328881	325328959	325329151	325329207
325329414	325329552	325329665	325329872	325329895	325329903	325329939	325330096	325330177
325330222	325330226	325330253	325330462	325330506	325330526	325330592	325330594	325330672
325330754	325330823	325330825	325330947	325330979	325331128	325331794	325331898	325332065
325332084	325332136	325332436	325332452	325332494	325332575	325332602	325332630	325332644
325332695	325332721	325332761	325332934	325333045	325333075	325333150	325333168	325333202
325333331	325333382	325333487	325333532	325333591	325333675	325333706	325333718	325333789
325333802	325333907	325333965	325334048	325334332	325334335	325334516	325334554	325334722
325334801	325334940	325335001	325335109	325335115	325335197	325335430	325335448	325335507
325335551	325335625	325335930	325336007	325336024	325336080	325336093	325336103	325336155
325336191	325336221	325336340	325336506	325336561	325336616	325336629	325336689	325336749
325336805	325336829	325336832	325336997	325337015	325337034	325337057	325337144	325337151
325337163	325337244	325337372	325337530	325337535	325337590	325337591	325337774	325337969
325338070	325338145	325338159	325338318	325338385	325338386	325338424	325338461	325338490
325338578	325338580	325338627	325338699	325338732	325338872	325338922	325338965	325339020
325339040	325339069	325339088	325339189	325339232	325339233	325339256	325339428	325339547
325339645	325339665	325339681	325339723	325339800	325339832	325339921	325339990	325339994
325340178	325340198	325340389	325340474	325340555	325340564	325340628	325340803	325340880
325341070	325341142	325341265	325341277	325341283	325341351	325341359	325341501	325341532
325341536	325341537	325341624	325341626	325341666	325341747	325341748	325341790	325341867

Continued on next page

Continued from previous page

325341878	325341920	325341930	325341956	325342187	325342191	325342194	325342276	325342285
325342292	325342356	325342420	325342440	325342496	325342616	325342617	325342655	325342859
325342956	325343033	325343044	325343116	325343132	325343158	325343162	325343302	325343391
325343402	325343428	325343463	325343481	325343498	325343561	325343594	325343675	325343737
325343811	325343812	325343835	325343846	325343851	325343874	325344099	325344238	325344391
325344404	325344544	325344558	325344627	325344782	325345094	325345146	325345197	325345237
325345352	325345370	325345427	325345430	325345497	325345656	325345665	325345798	325345799
325345827	325345964	325346096	325346382	325346424	325346445	325346700	325346737	325346775
325346783	325346857	325347069	325347250	325347292	325347416	325347448	325347459	325347506
325347523	325347634	325347661	325347687	325347778	325347799	325348024	325348333	325348393
325348423	325348500	325348601	325348723	325348724	325348997	325349172	325349219	325349233
325349258	325349430	325349457	325349738	325349892	325350022	325350114	325350122	325350219
325350230	325350567	325350609	325350754	325350774	325351236	325351403	325351630	325351633
325351703	325351796	325351841	325351851	325351865	325351876	325352056	325352063	325352126
325352127	325352195	325352422	325352499	325352592	325352628	325352664	325352671	325352703
325352801	325352945	325352998	325352999	325353024	325353102	325353152	325353233	325353448
325353480	325353719	325353829	325353891	325353895	325354034	325354037	325354168	325354249
325354378	325354457	325354492	325354748	325354869	325354885	325354886	325354962	325355026
325355037	325355112	325355331	325355350	325355353	325355432	325355600	325355715	325355777
325355874	325355903	325356055	325356231	325356324	325356349	325356424	325356586	325356602
325356625	325356688	325356709	325356828	325356846	325357068	325357070	325357134	325357166
325357227	325357281	325357322	325357375	325357446	325357458	325357708	325357752	325357783
325357977	325357999	325358001	325358007	325358188	325358232	325358375	325358417	325358610
325358627	325358678	325358818	325358997	325359110	325359293	325359365	325359445	325359502
325359526	325359553	325359617	325359682	325359736	325359873	325359960	325359974	325360204
325360214	325360247	325360252	325360304	325360493	325360506	325360598	325360681	325360728
325360796	325360810	325360854	325360886	325361117	325361155	325361169	325361259	325361298
325361367	325361369	325361395	325361404	325361520	325361633	325361737	325361824	325361891
325361961	325362099	325362188	325362279	325362323	325362358	325362572	325362585	325362632
325362643	325362753	325362779	325362891	325362917	325363098	325363151	325363264	325363500
325363550	325363654	325363683	325363765	325363859	325364459	325364559	325364613	325364629
325364645	325364669	325364763	325364945	325364954	325365004	325365097	325365162	325365256
325365304	325365309	325365388	325365440	325365518	325365684	325365723	325365811	325365838
325365843	325366093	325366230	325366248	325366259	325366300	325366315	325366460	325366931
325366979	325367006	325367088	325367136	325367197	325367232	325367259	325367268	325367281
325367286	325367295	325367423	325367456	325367461	325367675	325367728	325367948	325367970
325367993	325368058	325368090	325368101	325368171	325368201	325368243	325368313	325368390
325368471	325368496	325368507	325368524	325368541	325368675	325368702	325368819	325368823
325369123	325369186	325369213	325369382	325369640	325369725	325369731	325369767	325369805
325369871	325369891	325369946	325369950	325369987	325370009	325370048	325370051	325370075
325370269	325370312	325370321	325370387	325370394	325370442	325370497	325370540	325370557
325370636	325370725	325370789	325370812	325370888	325370922	325370939	325371077	325371114
325371244	325371343	325371362	325371566	325371658	325371743	325371765	325371990	325371998
325372064	325372087	325372274	325372280	325372393	325372413	325372582	325372902	325372970
325373072	325373219	325373239	325373241	325373275	325373305	325373405	325373421	325373470
325373475	325373478	325373633	325373823	325373952	325374007	325374035	325374152	325374419
325374625	325374696	325374787	325375079	325375104	325375227	325375240	325375344	325375374
325375452	325375497	325375554	325375624	325375627	325375752	325375816	325375900	325375938
325376020	325376174	325376269	325376287	325376326	325376425	325376509	325376558	325376622
325376678	325376690	325376699	325376767	325376958	325376967	325376980	325377104	325377120
325377232	325377239	325377262	325377310	325377324	325377361	325377368	325377370	325377448
325377451	325377471	325377536	325377636	325377659	325377700	325377709	325377787	325377811
325377843	325377913	325378025	325378033	325378069	325378094	325378205	325378383	325378386
325378387	325378416	325378597	325378611	325378650	325378702	325378714	325378717	325378797
325378881	325378882	325378942	325378965	325379001	325379036	325379370	325379378	325379385
325379469	325379518	325379533	325379594	325379655	325379714	325379756	325379808	325379840
325379873	325379986	325379990	325380060	325380118	325380145	325380271	325380284	325380364
325380457	325380621	325380632	325380645	325380776	325380795	325380876	325380890	325380910
325380987	325381040	325381181	325381198	325381240	325381273	325381316	325381323	325381327
325381361	325381478	325381572	325381680	325381689	325381825	325381826	325381841	325381865
325381942	325381977	325381978	325381988	325382028	325382031	325382155	325382172	325382223
325382272	325382413	325382637	325382913	325383039	325383052	325383099	325383158	325383184
325383262	325383364	325383604	325383692	325383704	325383823	325383852	325383996	325384038
325384082	325384114	325384353	325384424	325384472	325384525	325384581	325384621	325384691
325384717	325384732	325384838	325384980	325385010	325385025	325385077	325385118	325385188
325385276	325385378	325385457	325385460	325385473	325385601	325385612	325385709	325385722
325385754	325385874	325386197	325386805	325386807	325386841	325386850	325386930	325386985

Continued on next page

Continued from previous page

325386998	325387421	325387465	325387551	325387708	325387804	325387844	325387854	325387904
325388125	325388177	325388190	325388225	325388252	325388278	325388329	325388457	325388537
325388627	325388691	325388706	325388939	325388944	325388951	325388965	325389042	325389050
325389244	325389298	325389300	325389382	325389522	325389558	325389568	325389576	325389777
325389789	325389852	325389937	325390017	325390080	325390253	325390420	325390424	325390486
325390684	325390701	325390713	325390911	325390965	325391019	325391203	325391204	325391408
325391440	325391478	325391579	325391599	325391702	325391824	325391830	325391891	325391942
325392144	325392158	325392163	325392270	325392277	325392288	325392378	325392400	325392420
325392519	325392568	325392875	325392953	325392973	325393023	325393156	325393347	325393487
325393605	325393776	325393786	325393830	325393879	325393989	325393998	325394003	325394035
325394140	325394208	325394240	325394324	325394449	325394476	325394537	325394556	325394610
325394656	325394764	325394774	325394856	325395054	325395172	325395242	325395270	325395287
325395403	325395445	325395566	325395634	325395691	325395695	325395977	325395992	325396013
325396020	325396207	325396227	325396250	325396272	325396292	325396320	325396421	325396564
325396576	325396703	325396716	325396896	325396962	325397025	325397160	325397190	325397200
325397217	325397411	325397418	325397429	325397610	325397675	325397758	325398058	325398098
325398119	325398263	325398270	325398353	325398377	325398396	325398634	325398640	325398653
325398837	325398883	325398981	325398984	325399225	325399318	325399394	325399463	325399549
325399778	325399786	325399854	325400099	325400265	325400461	325400581	325400820	325400895
325401010	325401168	325401223	325401286	325401296	325401315	325401319	325401532	325401593
325401613	325401614	325401654	325401726	325401770	325401821	325402047	325402084	325402096
325402221	325402268	325402315	325402437	325402455	325402471	325402473	325402475	325402542
325402579	325402634	325402744	325402809	325402847	325402896	325403145	325403350	325403401
325403427	325403437	325403453	325403479	325403508	325403509	325403567	325403571	325403573
325403653	325403788	325403853	325403867	325403971	325404068	325404159	325404275	325404415
325404459	325404480	325404519	325404524	325404537	325404626	325404665	325404797	325404933
325404956	325405075	325405157	325405293	325405384	325405396	325405557	325405591	325405627
325405681	325405746	325405759	325405903	325406126	325406143	325406178	325406217	325406218
325406225	325406354	325406617	325407004	325407082	325407191	325407240	325407260	325407311
325407334	325407437	325407475	325407481	325407915	325407939	325407952	325407977	325408047
325408139	325408206	325408414	325408420	325408433	325408572	325408670	325408766	325408786
325408878	325408954	325409259	325409319	325409409	325409420	325409507	325409666	325409670
325409840	325409900	325409988	325410001	325410024	325410049	325410050	325410361	325410386
325410396	325410547	325410705	325410801	325411005	325411044	325411083	325411291	325411296
325411338	325411382	325411601	325411665	325411779	325411863	325411881	325411910	325411920
325411964	325411977	325411988	325412080	325412115	325412126	325412241	325412398	325412400
325412445	325412523	325412557	325412560	325412585	325412601	325412750	325412761	325412897
325412912	325413257	325413368	325413449	325413491	325413531	325413779	325413835	325413853
325414080	325414148	325414513	325414531	325414560	325414615	325414637	325414656	325414670
325414743	325414869	325414975	325415071	325415076	325415118	325415119	325415141	325415333
325415413	325415494	325415535	325415564	325415611	325415635	325415820	325415856	325415930
325415942	325415945	325415986	325415989	325416021	325416135	325416149	325416151	325416161
325416175	325416177	325416201	325416210	325416314	325416410	325416433	325416542	325416638
325416713	325416745	325416751	325416838	325416963	325417021	325417078	325417081	325417103
325417325	325417366	325417484	325417555	325417597	325417626	325417630	325417811	325417852
325417897	325417987	325418046	325418167	325418315	325418559	325418603	325418750	325418812
325418921	325419090	325419099	325419102	325419212	325419234	325419302	325419520	325419837
325419853	325420000	325420038	325420151	325420252	325420386	325420551	325420580	325420770
325420866	325421022	325421040	325421110	325421121	325421284	325421331	325421390	325421485
325421507	325421533	325421586	325421650	325421670	325421723	325421851	325421968	325421993
325422153	325422197	325422265	325422344	325422443	325422516	325422541	325422660	325422668
325422733	325422746	325422846	325422889	325423075	325423173	325423227	325423262	325423426
325423641	325423843	325424043	325424106	325424107	325424231	325424272	325424371	325424447
325424804	325424871	325424942	325425287	325425613	325425626	325425725	325425763	325425851
325425901	325425915	325425949	325426021	325426142	325426283	325426292	325426423	325426455
325426533	325426593	325426607	325426655	325426842	325426929	325426997	325427129	325427218
325427254	325427288	325427407	325427479	325427589	325427695	325427874	325427895	325427928
325427987	325427998	325428005	325428126	325428258	325428494	325428554	325428677	325428774
325428787	325428848	325428879	325429043	325429389	325429476	325429608	325429747	325429750
325429757	325429790	325430019	325430028	325430260	325430292	325430302	325430325	325430335
325430343	325430371	325430372	325430399	325430462	325430630	325430654	325430707	325430714
325430945	325431079	325431108	325431120	325431204	325431307	325431398	325431407	325431553
325431742	325431902	325431931	325432083	325432561	325432578	325432597	325432619	325432736
325432738	325432771	325432797	325432827	325432941	325432965	325432979	325433071	325433138
325433230	325433246	325433285	325433300	325433518	325433596	325433601	325433673	325433714
325433974	325433989	325434057	325434075	325434116	325434401	325434436	325434461	325434523
325434534	325434540	325434554	325434580	325434718	325434740	325434745	325434752	325434759

Continued on next page

Continued from previous page

325434814	325434824	325434837	325435031	325435197	325435295	325435298	325435320	325435492
325435494	325435572	325435613	325435615	325435716	325435764	325435777	325435784	325435806
325435958	325436016	325436017	325436055	325436083	325436248	325436352	325436378	325436396
325436495	325436537	325436549	325436583	325436705	325436763	325436868	325436928	325437082
325437151	325437155	325437309	325437522	325437533	325437561	325437580	325437733	325437834
325437988	325438073	325438215	325438278	325438303	325438313	325438336	325438402	325438419
325438492	325438579	325438633	325438787	325438897	325438942	325438956	325438981	325439259
325439309	325439321	325439357	325439394	325439503	325439504	325439558	325439583	325439628
325439703	325440022	325440160	325440245	325440407	325440512	325440562	325440610	325440645
325440758	325440786	325440993	325441038	325441043	325441157	325441194	325441388	325441530
325441566	325441602	325441672	325441838	325441951	325441967	325442016	325442190	325442212
325442253	325442263	325442265	325442426	325442564	325442636	325442819	325442927	325442966
325443016	325443037	325443052	325443086	325443181	325443231	325443449	325443603	325443831
325443840	325443897	325443976	325443985	325443995	325444063	325444100	325444179	325444218
325444255	325444521	325444538	325444637	325444662	325444665	325444683	325444791	325444807
325444819	325444865	325444938	325445055	325445180	325445194	325445243	325445297	325445353
325445441	325445541	325445804	325445810	325445880	325445887	325446102	325446211	325446244
325446557	325446651	325446662	325446670	325446691	325446857	325447087	325447209	325447366
325447382	325447465	325447505	325447509	325447510	325447534	325447548	325447639	325447706
325447731	325447783	325447865	325448001	325448004	325448052	325448075	325448106	325448198
325448259	325448276	325448284	325448416	325448474	325448553	325448564	325448713	325448919
325448920	325449069	325449112	325449132	325449178	325449189	325449364	325449902	325449946
325449970	325450042	325450465	325450564	325450566	325450619	325450662	325450694	325450698
325450708	325450838	325450967	325451027	325451060	325451196	325451380	325451488	325451523
325451593	325451615	325451691	325451703	325451705	325451799	325451810	325451980	325452084
325452109	325452181	325452185	325452419	325452548	325452556	325452728	325452760	325452931
325452998	325453096	325453111	325453218	325453335	325453443	325453451	325453499	325453630
325453730	325453761	325453813	325453823	325453872	325453884	325453885	325454100	325454144
325454274	325454276	325454304	325454439	325454534	325454541	325454617	325454705	325454718
325454729	325454791	325454988	325455013	325455056	325455184	325455242	325455296	325455331
325455532	325455644	325455701	325455831	325455912	325455965	325456021	325456063	325456105
325456225	325456454	325456505	325456523	325456585	325456612	325456717	325456846	325456929
325457047	325457082	325457322	325457381	325457425	325457436	325457471	325457484	325457503
325457530	325457649	325457658	325457744	325457749	325457785	325458120	325458227	325458302
325458343	325458431	325458550	325458570	325458658	325458660	325458805	325458816	325458848
325458915	325458975	325458986	325459054	325459062	325459182	325459238	325459367	325459382
325459552	325459579	325459644	325459709	325459729	325459771	325459899	325460077	325460105
325460109	325460111	325460204	325460271	325460413	325460489	325460577	325460592	325460669
325460793	325461123	325461189	325461217	325461243	325461311	325461405	325461662	325461734
325462171	325462203	325462321	325462354	325462358	325462373	325462402	325462691	325462776
325462957	325463063	325463131	325463155	325463168	325463194	325463213	325463484	325463647
325463652	325463700	325463778	325463831	325463927	325463928	325464155	325464176	325464207
325464232	325464389	325464439	325464457	325464524	325464566	325464765	325464933	325465003
325465070	325465095	325465098	325465123	325465158	325465231	325465620	325465681	325465735
325465777	325465874	325466007	325466023	325466076	325466117	325466211	325466314	325466384
325466400	325466412	325466466	325466621	325466705	325466708	325466826	325466828	325466842
325466866	325467039	325467096	325467145	325467163	325467200	325467247	325467257	325467344
325467361	325467635	325467697	325467699	325467744	325467831	325467973	325467984	325468003
325468032	325468332	325468491	325468614	325468691	325468882	325468949	325469137	325469213
325469248	325469293	325469294	325469379	325469539	325469600	325469624	325469686	325470068
325470079	325470091	325470106	325470166	325470203	325470206	325470298	325470301	325470375
325470394	325470395	325470411	325470445	325470800	325470833	325470927	325471007	325471091
325471105	325471134	325471188	325471248	325471304	325471321	325471380	325471457	325471524
325471531	325471846	325472038	325472077	325472191	325472233	325472348	325472437	325472508
325472593	325472601	325472616	325472685	325472690	325472792	325472902	325472910	325472966
325473066	325473205	325473261	325473370	325473718	325473741	325473826	325473923	325473966
325473993	325473994	325474096	325474119	325474175	325474268	325474415	325474735	325474765
325474803	325474877	325474950	325475115	325475229	325475258	325475277	325475287	325475301
325475376	325475411	325475540	325475642	325475698	325475722	325475745	325475789	325475881
325475932	325476051	325476302	325476354	325476871	325476926	325476939	325476940	325477085
325477196	325477256	325477268	325477385	325477422	325477515	325477759	325477762	325477948
325477983	325478019	325478084	325478293	325478438	325478454	325478470	325478523	325478525
325478641	325478653	325478660	325478661	325478711	325478769	325478928	325479144	325479161
325479215	325479271	325479337	325479360	325479478	325479515	325479556	325479787	325480047
325480092	325480174	325480292	325480293	325480316	325480390	325480474	325480481	325480513
325480566	325480751	325480815	325480916	325481164	325481266	325481383	325481509	325481536
325481565	325481801	325481802	325481851	325481895	325481901	325481927	325482001	325482085

Continued on next page

Continued from previous page

325482092	325482116	325482181	325482225	325482254	325482346	325482560	325482606	325482790
325482843	325482940	325482944	325482949	325483023	325483227	325483233	325483241	325483249
325483277	325483382	325483392	325483430	325483501	325483502	325483542	325483643	325483948
325483956	325484502	325484510	325484538	325484738	325484748	325484837	325484858	325484916
325484961	325485057	325485066	325485068	325485149	325485233	325485336	325485345	325485397
325485518	325485780	325485805	325485832	325485844	325485954	325486011	325486017	325486082
325486273	325486302	325486522	325486588	325486621	325486650	325486695	325486799	325486831
325486917	325486972	325487003	325487193	325487284	325487344	325487424	325487473	325487520
325487663	325488050	325488220	325488322	325488324	325488425	325488458	325488489	325488631
325488632	325488644	325488853	325488894	325488934	325488939	325489017	325489031	325489230
325489240	325489271	325489298	325489329	325489386	325489570	325489603	325489677	325489687
325489738	325489773	325489794	325489874	325489895	325489941	325489955	325489968	325489992
325490000	325490072	325490075	325490147	325490211	325490214	325490215	325490310	325490531
325490574	325490631	325490682	325490808	325490836	325490970	325491006	325491010	325491069
325491094	325491109	325491150	325491178	325491219	325491236	325491250	325491290	325491341
325491434	325491551	325491629	325491639	325491672	325491745	325491747	325491798	325491820
325491911	325491962	325492012	325492108	325492182	325492262	325492280	325492311	325492504
325492505	325492513	325492594	325492800	325492816	325493097	325493390	325493814	325494005
325494213	325494239	325494249	325494344	325494363	325494372	325494395	325494407	325494430
325494481	325494494	325494529	325494558	325494595	325494785	325494886	325495000	325495002
325495048	325495074	325495088	325495208	325495211	325495259	325495270	325495342	325495355
325495377	325495454	325495455	325495460	325495795	325495840	325496216	325496274	325496289
325496315	325496334	325496436	325496663	325496735	325496847	325496849	325496880	325496887
325496904	325496907	325497042	325497113	325497223	325497260	325497297	325497395	325497508
325497568	325497594	325497705	325497733	325497815	325497835	325497836	325497985	325498011
325498013	325498032	325498070	325498099	325498104	325498128	325498203	325498212	325498247
325498338	325498341	325498365	325498375	325498418	325498730	325498754	325498772	325498826
325498855	325498858	325498978	325499126	325499175	325499196	325499271	325499517	325499603
325499605	325499630	325499768	325499779	325499922	325500001	325500348	325500460	325500480
325500547	325500767	325500893	325500924	325500941	325501083	325501153	325501319	325501487
325501542	325501549	325501654	325502055	325502082	325502119	325502256	325502263	325502348
325502462	325502480	325502519	325502542	325502652	325502753	325502831	325502875	325502975
325503038	325503084	325503132	325503238	325503303	325503355	325503373	325503497	325503628
325503760	325503842	325503967	325504099	325504122	325504208	325504297	325504384	325504430
325504431	325504509	325504682	325504707	325504838	325504929	325504963	325504985	325505005
325505051	325505136	325505169	325505373	325505395	325505413	325505442	325505449	325505528
325505569	325505675	325505676	325505731	325505785	325505939	325505951	325506018	325506066
325506097	325506139	325506180	325506294	325506350	325506364	325506489	325506506	325506528
325506617	325506639	325506991	325507043	325507064	325507117	325507213	325507232	325507404
325507413	325507548	325507567	325507618	325507650	325507713	325507800	325507819	325507827
325507837	325507863	325507866	325507899	325507927	325507941	325508130	325508738	325508814
325508815	325508874	325509052	325509222	325509227	325509294	325509342	325509427	325509451
325509499	325509540	325509552	325509582	325509948	325510077	325510082	325510164	325510339
325510540	325510623	325510667	325510677	325510833	325510842	325510895	325510919	325510929
325511035	325511144	325511343	325511502	325511518	325511671	325511684	325511763	325511817
325511836	325511872	325512098	325512172	325512265	325512275	325512329	325512389	325512430
325512433	325512449	325512537	325512540	325512869	325512878	325512939	325512948	325512957
325513015	325513059	325513103	325513188	325513202	325513367	325513411	325513637	325513657
325513682	325513749	325513751	325513813	325513815	325513817	325513892	325513964	325514022
325514053	325514071	325514293	325514504	325514659	325514854	325514895	325514929	325514940
325514958	325514986	325515095	325515103	325515124	325515259	325515448	325515508	325515537
325515582	325515598	325515819	325515826	325516044	325516088	325516149	325516333	325516355
325516415	325516540	325516568	325516635	325516747	325516763	325516814	325516846	325516853
325516888	325516905	325516978	325517214	325517276	325517395	325517418	325517461	325517550
325517741	325517744	325517764	325517900	325517937	325517987	325518163	325518308	325518330
325518354	325518376	325518769	325518822	325518910	325518914	325518923	325519321	325519387
325519388	325519424	325519457	325519692	325519736	325519752	325519795	325519870	325519876
325520006	325520013	325520038	325520065	325520115	325520177	325520221	325520274	325520305
325520351	325520355	325520364	325520404	325520465	325520844	325520917	325520969	325520988
325521068	325521087	325521398	325521400	325521424	325521576	325521665	325521722	325521759
325521785	325521804	325521941	325521993	325522284	325522321	325522329	325522378	325522401
325522560	325522630	325522674	325522687	325522742	325522773	325522996	325523008	325523061
325523160	325523387	325523407	325523523	325523716	325523729	325523833	325523836	325523937
325524025	325524056	325524150	325524191	325524224	325524235	325524431	325524497	325524512
325524566	325524567	325524680	325524688	325524695	325524956	325525066	325525267	325525358
325525523	325525547	325525795	325525912	325525995	325526015	325526124	325526226	325526428
325526508	325526566	325526812	325526826	325526877	325527031	325527041	325527113	325527479

Continued on next page

Continued from previous page

325527642	325527649	325527794	325527832	325527955	325528100	325528218	325528299	325528751
325528822	325528837	325528879	325528966	325529070	325529245	325529249	325529273	325529335
325529401	325529509	325529545	325529648	325529762	325529813	325529912	325530004	325530016
325530025	325530178	325530356	325530502	325530509	325530560	325530563	325530712	325530774
325530957	325530982	325531031	325531175	325531318	325531324	325531397	325531468	325531471
325531582	325531641	325531664	325531794	325531802	325531944	325532342	325532471	325532484
325532549	325532550	325532660	325532693	325532749	325532867	325532947	325533222	325533242
325533403	325533404	325533459	325533468	325533538	325533544	325533564	325533783	325533899
325533907	325533933	325534012	325534156	325534276	325534313	325534342	325534675	325534713
325534730	325534836	325534952	325535094	325535136	325535148	325535170	325535198	325535207
325535269	325535298	325535430	325535440	325535972	325536081	325536087	325536127	325536130
325536136	325536314	325536534	325536542	325536560	325536610	325536718	325536751	325536948
325536978	325537261	325537616	325537631	325537650	325537661	325537668	325537775	325537795
325537827	325537964	325537982	325538074	325538109	325538113	325538172	325538262	325538329
325538377	325538486	325538568	325538651	325538739	325538752	325538860	325538956	325539125
325539158	325539231	325539256	325539266	325539384	325539474	325539637	325539681	325539754
325539759	325539820	325539872	325539878	325539975	325540023	325540064	325540093	325540105
325540150	325540164	325540226	325540537	325540653	325540712	325540826	325540832	325540840
325540866	325541009	325541194	325541225	325541275	325541293	325541360	325541442	325541505
325541774	325541799	325541908	325541909	325541977	325542018	325542054	325542080	325542106
325542131	325542156	325542275	325542451	325542468	325542481	325542527	325542564	325542611
325542660	325542694	325542789	325542871	325542995	325543086	325543146	325543214	325543293
325543372	325543406	325543617	325543631	325543660	325543662	325543709	325543732	325543798
325544343	325544389	325544425	325544444	325544595	325544627	325544722	325544788	325544822
325544853	325544872	325544939	325544953	325545006	325545072	325545078	325545266	325545340
325545342	325545344	325545391	325545411	325545421	325545471	325545502	325545567	325545614
325545692	325545841	325545911	325545979	325546374	325546413	325546455	325546523	325546706
325546778	325546796	325546945	325547090	325547301	325547448	325547489	325547518	325547528
325547702	325547705	325547815	325547897	325547910	325548002	325548014	325548088	325548124
325548150	325548364	325548633	325548656	325548671	325548794	325548890	325548935	325548937
325549011	325549026	325549041	325549115	325549158	325549310	325549343	325549437	325549473
325549482	325549644	325549686	325549723	325549739	325549824	325549870	325549882	325549980
325550191	325550320	325550407	325550483	325550617	325550648	325550822	325550862	325550915
325551083	325551173	325551176	325551178	325551341	325551357	325551444	325551506	325551551
325551598	325551617	325551626	325551677	325551708	325551726	325551732	325551919	325551977
325551993	325552189	325552253	325552567	325552617	325552709	325552753	325552786	325552792
325552999	325553125	325553159	325553190	325553385	325553575	325553620	325553703	325553733
325553758	325553854	325554065	325554116	325554151	325554194	325554236	325554424	325554463
325554509	325554853	325554965	325555001	325555048	325555246	325555454	325555587	325555597
325555742	325555922	325556098	325556272	325556312	325556324	325556344	325556395	325556411
325556448	325556565	325556596	325556726	325556829	325556954	325557273	325557313	325557433
325557458	325557632	325557807	325558085	325558246	325558322	325558411	325558455	325558598
325558609	325558620	325558690	325558698	325558770	325558865	325558926	325559067	325559134
325559562	325559583	325559611	325559750	325559857	325560125	325560239	325560338	325560342
325560356	325560514	325560588	325560613	325560654	325560697	325560712	325560805	325560889
325561091	325561220	325561317	325561363	325561417	325561469	325561529	325561689	325561727
325561752	325561993	325562092	325562109	325562112	325562412	325562421	325562451	325562456
325562502	325562554	325562606	325562661	325562667	325562680	325562737	325562750	325562805
325562835	325562897	325563108	325563118	325563144	325563244	325563331	325563587	325563614
325563701	325563750	325563765	325563792	325563798	325563826	325563850	325563852	325563865
325563879	325564031	325564085	325564145	325564250	325564281	325564356	325564400	325564509
325564684	325564724	325564743	325564777	325564780	325564832	325564868	325564947	325565060
325565212	325565491	325565541	325565651	325565683	325565764	325565918	325566003	325566019
325566052	325566067	325566106	325566241	325566344	325566455	325566667	325566738	325566763
325566807	325566827	325566938	325566942	325566952	325567100	325567117	325567233	325567385
325567456	325567529	325567533	325567536	325567682	325567796	325567931	325568224	325568287
325568383	325568419	325568425	325568481	325568629	325568741	325569073	325569089	325569091
325569179	325569185	325569227	325569292	325569308	325569368	325569371	325569401	325569470
325569556	325569610	325569698	325569717	325569812	325569852	325569867	325569957	325570053
325570063	325570104	325570138	325570271	325570358	325570480	325570587	325570733	325570835
325570919	325570981	325571048	325571125	325571153	325571236	325571269	325571352	325571355
325571577	325571603	325571628	325571744	325571789	325572121	325572125	325572193	325572224
325572248	325572287	325572379	325572421	325572574	325572734	325572876	325572950	325572989
325573070	325573125	325573141	325573159	325573252	325573286	325573342	325573422	325573736
325573751	325573778	325573807	325573922	325573973	325574020	325574087	325574159	325574197
325574231	325574357	325574375	325574458	325574531	325574557	325574594	325574669	325574764
325574876	325574995	325575142	325575225	325575229	325575266	325575570	325575585	325575614

Continued on next page

Continued from previous page

325575689	325575707	325575718	325575752	325575844	325575945	325575972	325576216	325576291
325576304	325576352	325576580	325576621	325576672	325576683	325576704	325576826	325576887
325577089	325577326	325577329	325577330	325577510	325577692	325577733	325577764	325577838
325577987	325578005	325578097	325578193	325578215	325578305	325578349	325578358	325578415
325578443	325578558	325578595	325578702	325578711	325578836	325578867	325578938	325578941
325578987	325579067	325579094	325579147	325579182	325579303	325579489	325579539	325579777
325579789	325579836	325579861	325579944	325579952	325580000	325580037	325580257	325580353
325580407	325580431	325580658	325580688	325580779	325580853	325580888	325580898	325580908
325581006	325581055	325581197	325581230	325581315	325581331	325581485	325581710	325581713
325581750	325581943	325582147	325582184	325582245	325582331	325582365	325582413	325582442
325582463	325582470	325582640	325582669	325582909	325582942	325582971	325583016	325583030
325583056	325583245	325583280	325583359	325583360	325583418	325583471	325583493	325583536
325583554	325583769	325583898	325584005	325584112	325584185	325584241	325584294	325584328
325584382	325584482	325584742	325584807	325585004	325585067	325585090	325585099	325585532
325585616	325585883	325585890	325585931	325585946	325585975	325586049	325586053	325586133
325586165	325586185	325586211	325586251	325586343	325586513	325586522	325586526	325586566
325586577	325586595	325586697	325586728	325586737	325587158	325587163	325587210	325587326
325587439	325587571	325587577	325587626	325587700	325587712	325587760	325587814	325587835
325587976	325588011	325588041	325588084	325588177	325588265	325588270	325588329	325588533
325588624	325588995	325589010	325589186	325589219	325589347	325589378	325589608	325589874
325590048	325590069	325590075	325590098	325590302	325590378	325590410	325590459	325590518
325590622	325590666	325590707	325590716	325590809	325590888	325591264	325591363	325591395
325591419	325591482	325591490	325591644	325591721	325591775	325591785	325591824	325591902
325591975	325592038	325592045	325592207	325592210	325592382	325592431	325592441	325592483
325592808	325592917	325593012	325593134	325593312	325593349	325593431	325593660	325593737
325593914	325593916	325593922	325593969	325593995	325594042	325594066	325594090	325594150
325594159	325594224	325594312	325594384	325594450	325594453	325594461	325594528	325594626
325594645	325594678	325594806	325594941	325595181	325595266	325595286	325595596	325595638
325595673	325595732	325595889	325595908	325595954	325596007	325596334	325596336	325596425
325596676	325597036	325597082	325597134	325597186	325597294	325597305	325597368	325597443
325597474	325597813	325597830	325598196	325598206	325598262	325598287	325598291	325598607
325598843	325598861	325598933	325598947	325599206	325599243	325599328	325600125	325600230
325600268	325600477	325600614	325600638	325600642	325600754	325600888	325600892	325601020
325601432	325601569	325601865	325601888	325601921	325601959	325602115	325602319	325602422
325602443	325602460	325602527	325602817	325602942	325603024	325603054	325603188	325603272
325603445	328497258	328497398	328497440	328497462	328497641	328497683	328497698	328497877
328497960	328497979	328498021	328498042	328498221	328498376	328498701	328498722	328498790
328498792	328498793	328498930	328499044	328499533	328499542	328499591	328499608	328499612
328499686	328499722	328499744	328499813	328499909	328499912	328500012	328500056	328500135
328500192	328500578	328500621	328500695	328500711	328500780	328500820	328500847	328500953
328501003	328501152	328501362	328501442	328501477	328501771	328501881	328501940	328501960
328502089	328502322	328502326	328502359	328502366	328502404	328502535	328502600	328502665
328502758	328502930	328502971	328503016	328503052	328503098	328503126	328503151	328503167
328503281	328503406	328503414	328503461	328503537	328503550	328503558	328503575	328503682
328503739	328503844	328503853	328503930	328503931	328503991	328504058	328504164	328504219
328504325	328504529	328504570	328504637	328504680	328504688	328504715	328504728	328504758
328504979	328505087	328505146	328505322	328505356	328505409	328505607	328505638	328505792
328505865	328505981	328505987	328506163	328506255	328506276	328506438	328506457	328506512
328506530	328506543	328506624	328506749	328506816	328506894	328506936	328506996	328507022
328507084	328507160	328507397	328507406	328507463	328507617	328507686	328507741	328507946
328508061	328508089	328508184	328508192	328508300	328508848	328508982	328509005	328509140
328509223	328509231	328509233	328509275	328509279	328509295	328509491	328509493	328509532
328509560	328509635	328509715	328509773	328509780	328509869	328509912	328510032	328510088
328510102	328510215	328510273	328510581	328510625	328510811	328510833	328510865	328510879
328510939	328510988	328511091	328511217	328511255	328511293	328511417	328511442	328511482
328511546	328511591	328511728	328511835	328511932	328511941	328512047	328512290	328512324
328512361	328512620	328512649	328512728	328512918	328512994	328513089	328513110	328513232
328513373	328513425	328513458	328513625	328513671	328513672	328513675	328513953	328513967
328514059	328514346	328514405	328514510	328514694	328514747	328514797	328514800	328514856
328514901	328514951	328514965	328515042	328515052	328515061	328515166	328515230	328515259
328515393	328515497	328515516	328515616	328515693	328515704	328515816	328516208	328516286
328516323	328516401	328516487	328516674	328516716	328516801	328516804	328517091	328517111
328517125	328517207	328517270	328517277	328517502	328517510	328517677	328517751	328517760
328517765	328517809	328517810	328517872	328518161	328518526	328518541	328518728	328518749
328518802	328518957	328518985	328519061	328519089	328519164	328519189	328519242	328519255
328519272	328519407	328519415	328519563	328519583	328519649	328519665	328519720	328519851
328519908	328520138	328520150	328520547	328520608	328520682	328520695	328520889	328520912

Continued on next page

Continued from previous page

328520944	328520975	328521003	328521042	328521177	328521194	328521208	328521254	328521271
328521395	328521571	328521656	328521776	328521831	328521913	328522011	328522180	328522234
328522260	328522405	328522676	328522787	328522961	328523030	328523086	328523181	328523194
328523260	328523308	328523477	328523495	328523599	328523848	328523852	328523892	328524044
328524055	328524120	328524159	328524337	328524407	328524413	328524468	328524529	328524790
328524796	328524818	328524846	328525055	328525090	328525112	328525178	328525195	328525318
328525352	328525447	328525555	328525573	328525727	328525791	328525977	328526061	328526095
328526130	328526139	328526182	328526208	328526221	328526312	328526572	328526694	328526775
328526886	328526972	328526977	328526982	328527049	328527119	328527259	328527291	328527450
328527453	328527564	328527601	328527647	328527732	328527811	328528056	328528209	328528237
328528287	328528296	328528299	328528306	328528367	328528405	328528621	328528674	328528723
328528812	328528969	328529210	328529238	328529277	328529312	328529357	328529609	328529612
328529621	328529782	328529895	328529985	328529993	328529997	328530097	328530123	328530264
328530358	328530380	328530469	328530501	328530665	328530667	328530748	328530959	328530973
328530988	328531019	328531026	328531159	328531175	328531280	328531328	328531426	328531718
328531728	328531955	328531987	328532044	328532132	328532172	328532307	328532374	328532515
328532602	328532662	328532690	328532917	328532954	328532960	328532969	328532980	328533176
328533255	328533361	328533414	328533547	328533555	328533743	328533754	328533851	328534053
328534225	328534266	328534303	328534355	328534468	328534570	328534578	328534669	328534694
328534718	328534757	328534834	328534914	328535131	328535208	328535272	328535283	328535369
328535386	328535509	328535531	328535653	328535719	328535756	328535972	328536015	328536305
328536312	328536359	328536380	328536395	328536400	328536488	328536692	328536716	328536768
328536825	328536883	328536909	328536940	328537057	328537294	328537308	328537310	328537327
328537346	328537362	328537459	328537575	328537709	328537757	328537763	328537853	328537873
328537897	328537930	328537935	328538031	328538070	328538071	328538107	328538121	328538130
328538132	328538143	328538153	328538179	328538234	328538562	328538584	328538676	328538890
328538944	328538973	328539069	328539103	328539119	328539131	328539220	328539324	328539336
328539406	328539428	328539436	328539703	328539728	328539740	328539750	328539819	328539843
328539873	328540058	328540087	328540123	328540292	328540369	328540456	328540530	328540576
328540761	328540769	328540786	328540803	328540807	328541016	328541025	328541186	328541253
328541300	328541324	328541399	328541419	328541450	328541464	328541545	328541578	328541631
328541660	328541690	328541722	328541877	328542018	328542062	328542082	328542135	328542245
328542445	328542535	328542540	328542575	328542806	328542845	328542888	328543241	328543312
328543416	328543417	328543451	328543481	328543566	328543629	328543668	328543673	328543704
328543721	328543804	328543909	328543961	328544015	328544059	328544090	328544147	328544185
328544354	328544399	328544496	328544520	328544527	328544646	328544914	328544916	328545099
328545117	328545223	328545264	328545288	328545339	328545349	328545439	328545626	328545713
328545757	328545764	328545765	328545790	328545954	328546039	328546078	328546174	328546280
328546297	328546486	328546523	328546569	328549100	328549112	328606469	328618184	332768338
332768380	332768420	332768480	332768499	335093262	335093547	335093563	335093586	335184481

H. Electron Position Probabilities

Table H.1: Probabilities of electron lone pairs being placed on various atom types. An atom type, Z^x, is the element Z with x bonded neighbours, and the probabilities $\rho(y)$ are given for y = 1, 2 or 3 lone pairs. Probabilities are calculated across all occurrences of this atom type in the KEGG Drug database[1] and MMFF94 validation suite[2] (tables G.1 and G.2).

Atom type	Count	$\rho(1)$	$\rho(2)$	$\rho(3)$
Br^1	171	1.00000	1.00000	1.00000
C^1	5	1.00000	0.00000	0.00000
C^2	443	0.00000	0.00000	0.00000
C^3	63116	0.00008	0.00000	0.00000
C^4	56975	0.00000	0.00000	0.00000
Cl^1	1448	1.00000	1.00000	1.00000
Cl^4	2	0.00000	0.00000	0.00000
F^1	1791	1.00000	1.00000	1.00000
H^1	152568	0.00000	0.00000	0.00000
N^1	261	1.00000	0.05747	0.00000
N^2	3944	0.99366	0.02181	0.00000
N^3	11602	0.95871	0.00000	0.00000
N^4	224	0.00000	0.00000	0.00000
O^1	12692	1.00000	1.00000	0.09321
O^2	12567	1.00000	0.99960	0.00000
O^3	1	1.00000	0.00000	0.00000
P^2	3	1.00000	0.00000	0.00000
P^3	30	0.96667	0.00000	0.00000
P^4	318	0.00000	0.00000	0.00000
S^1	175	1.00000	1.00000	0.18286
S^2	1319	1.00000	0.99697	0.00000
S^3	71	0.98592	0.00000	0.00000
S^4	773	0.00000	0.00000	0.00000
S^6	1	0.00000	0.00000	0.00000

Table H.2: Probabilities of higher order bonds between pairs of atom types. An atom type, Z^x, is the element Z with x bonded neighbours, and the probabilities $\rho(y)$ are given for bond orders of $y = 2$ (double) or 3 (triple). Probabilities are calculated across all occurrences of this atom type in the KEGG Drug database,[1] and MMFF94 validation suite (tables G.1 and G.2).[2]

Type A	Type B	Count	$\rho(2)$	$\rho(3)$
C^1	N^2	5	1.00000	1.00000
C^2	C^2	97	1.00000	1.00000
C^2	C^3	180	0.06667	0.00000
C^2	N^1	241	1.00000	1.00000
C^2	N^2	10	0.20000	0.00000
C^2	S^1	2	1.00000	0.00000
C^3	C^3	52423	0.47710	0.00000
C^3	N^2	5971	0.58617	0.00000
C^3	N^3	11710	0.01503	0.00000
C^3	O^1	9761	0.94960	0.00000
C^3	O^2	6665	0.00090	0.00000
C^3	P^2	3	1.00000	0.00000
C^3	S^1	137	0.85401	0.00000
C^3	S^2	1333	0.00300	0.00000
C^3	S^3	42	0.04762	0.00000
C^4	O^1	9	0.11111	0.00000
Cl^4	O^1	8	0.37500	0.00000
N^1	N^2	20	1.00000	0.25000
N^2	N^2	301	0.51495	0.00000
N^2	N^3	497	0.03421	0.00000
N^2	N^4	2	0.50000	0.00000
N^2	O^1	23	0.65217	0.00000
N^2	S^3	1	1.00000	0.00000
N^2	S^4	55	0.07273	0.00000
N^3	N^3	149	0.01342	0.00000
N^3	O^1	654	0.47095	0.00000
N^4	O^1	9	0.11111	0.00000
O^1	P^3	3	0.66667	0.00000
O^1	P^4	470	0.63191	0.00000
O^1	S^2	3	1.00000	0.00000
O^1	S^3	73	0.91781	0.00000
O^1	S^4	1679	0.91781	0.00000
P^4	S^1	27	0.77778	0.00000
S^1	S^3	2	1.00000	0.00000
S^1	S^4	1	1.00000	0.00000

Glossary

AMBER Assisted Model Building with Energy Refinement; a family of force fields for molecular dynamic simulations of biomolecules.

bond order the number of chemical bonds between a pair of atoms. Bond orders are restricted to integer values..

CHARMM Chemistry at HARvard using Molecular Mechanics; a family of force fields for molecular dynamic simulations of biomolecules.

CherryPicker An algorithm for parameterising novel large molecules using fragmented smaller molecules..

CIP Cahn–Ingold–Prelog priority rules are a standard process used to completely and unequivocally assign an R or S descriptor to each stereocenter and an E or Z descriptor to each double bond of a molecule..

force field A set of parameters describing how atoms interact with one another within a system.

formal charge the charge assigned to an atom in a molecule assuming that electrons in all chemical bonds are shared equally between atoms. A formal charge is limited to integer values..

GROMOS Groningen Molecular Simulation; a family of united atom force fields for molecular dynamic simulations of biomolecules.

MMFF94 Merck Molecular Force Field; a general purpose force field.

molecular dynamics A simulation technique for computing the equilibrium properties of a classical many-body system..

OPLS Optimised Potentials for Liquid Simulations; a family of force fields for molecular dynamic simulations of biomolecules.