The GROMOS Software for (Bio)Molecular Simulation

GROMOS87

Groningen Molecular Simulation (GROMOS) Library Manual

W.F. van Gunsteren, H.J.C. Berendsen
GROMOS is an acronym of the GROningen MOlecular Simulation computer program package, which has been developed since 1978 for the dynamic modelling of (bio)molecules. This document contains the first manual, which accompanied the major release of GROMOS in 1987. While it remained incomplete (chapters 6 and 8-16 were not finalized), the structure formed the basis for the manuals of the GROMOS96 and GROMOS11 versions.

The GROMOS87 manual can be cited as

Contents

0. Introduction

1. Purpose, scope and limitations
   1.1 Dynamic modelling of molecular systems in chemistry and physics
   1.2 Overview of GROMOS
   1.3 Applications in chemistry and physics
   1.4 Philosophy underlying GROMOS structure
   1.5 GROMOS force fields
   1.6 Computer time required for MD simulations
   1.7 Limitations

2. Modelling of systems and description of the GROMOS force fields
   2.1 Introduction
   2.2 Atom types
   2.3 Covalent bond forces
   2.4 Covalent bond angle forces
   2.5 Improper dihedral angle forces
   2.6 Dihedral angle torsion forces
   2.7 Non-bonded forces
      2.7.1 Excluded neighbours
      2.7.2 Normal van der Waals' interaction
      2.7.3 Third neighbour van der Waals' interaction
      2.7.4 Coulomb interaction
2.7.5 Charge groups
2.7.6 Searching neighbours
  2.7.6.1 Scanning all charge group pairs
  2.7.6.2 Atom-atom neighbour list
  2.7.6.3 Charge group neighbour list
  2.7.6.4 Grid-cell plus linked list technique
2.7.7 Long-range Coulomb interaction
2.7.8 Switching functions
2.8 Special forces
  2.8.1 Introduction
  2.8.2 Position restraining
  2.8.3 Distance restraining
  2.8.4 Virtual and pseudo atoms
  2.8.5 Bond-angle restraining
  2.8.6 Dihedral angle restraining
2.9 Constraints
  2.9.1 Introduction
  2.9.2 Constraints using subroutine SHAKE
    2.9.2.1 Constrained positions
    2.9.2.2 Constrained velocities
    2.9.2.3 Constrained forces
  2.9.3 Bond-length constraints in solute
  2.9.4 Bond-length and bond-angle constraints in solvent
2.10 Treatment of boundaries
  2.10.1 Introduction
  2.10.2 System in vacuo
  2.10.3 Periodic boundary conditions
2.10.3.1 Introduction II-87
2.10.3.2 Periodic rectangular box II-90
2.10.3.3 Periodic monoclinic box II-92
2.10.3.4 Periodic truncated octahedron II-94
2.10.4 Wall region of restrained atoms II-96

2.11 Energy minimization II-98
2.11.1 Introduction II-98
2.11.2 Steepest descents minimization II-100
2.11.3 Conjugate gradients minimization II-102
2.11.4 Steepest descents minimization with constraints (SHAKE) II-106
2.11.5 Conjugate gradients minimization with constraints (SHAKE) II-107

2.12 Molecular Dynamics
2.12.1 Introduction
2.12.2 Temperature scaling
2.12.3 Pressure scaling
2.12.4 Calculation of the virial
2.12.5 MD algorithms
2.12.6 Initialization and equilibration

2.13 Free energy determination
2.13.1 Introduction
2.13.2 Parameterization of the Hamiltonian
2.13.3 Constraints
2.13.4 Choice of state A and state B

2.14 Stochastic Dynamics
2.14.1 Introduction
2.14.2 Leap-frog SD algorithm
2.14.3 Choice of atomic friction coefficient
2.14.4 Initialization and equilibration

3. GROMOS Data Structure

3.1 Introduction

3.2 Title record on GROMOS files

3.3 Molecular Topologies
   3.3.1 Introduction
   3.3.2 Molecular topology, binary form
   3.3.3 Molecular topology, formatted form
   3.3.4 Perturbation of a molecular topology

3.4 Atomic Cartesian coordinates and related quantities
   3.4.1 Storage of a single configuration or time frame
      3.4.1.1 Standard binary form
      3.4.1.2 Standard formatted form
      3.4.1.3 Other non-standard forms
   3.4.2 Storage of trajectories or series of configurations and related quantities
      3.4.2.1 Storage of trajectories by subr. PACK
      3.4.2.2 Storage of trajectories in other non-standard forms
   3.4.2.3 Storage of energies belonging to trajectories

3.5 Residue Topology building blocks
   3.5.1 Introduction
   3.5.2 Separate molecules
   3.5.3 Amino acid residues
   3.5.4 Nucleotides
   3.5.5 Glucose units
   3.5.6 Other linear chain building blocks
3.6 Interaction Function Parameters

3.6.1 Introduction

3.6.2 Bond-stretch, bond-angle bending and torsional interaction parameters

3.6.3 Van der Waals interaction parameters and integer atom codes

3.6.4 Atomic charges and charge group codes

3.6.5 Excluded neighbors

3.7 Other types of data

3.7.1 Atom-atom distance restraints

3.7.2 Internal coordinates

3.7.3 Single quantity trajectories

3.7.4 Solute constraints

3.7.5 Dihedral restraints

Standard GROMOS files

4.1 GROMOS force field files

4.1.1 The 26Cl force field

4.1.2 The 37C2 force field

4.1.3 The 37D2 force field

4.1.4 The 37C4 force field

4.1.5 The 37D4 force field

4.2 GROMOS residue topology building blocks

4.3 GROMOS standard configurations

4.3.1 Water

4.3.2 Amino acids

5. GROMOS Library architecture
5.1 Introduction
5.2 Topology builders
5.3 Coordinate resequencers and reformatters (to GROMOS)
5.4 Coordinate generators
5.5 Minimizers and simulators
5.6 Analysers
5.7 Coordinate reformatters (from GROMOS)

6. GROMOS Technical details
6.1 Array sizes and common blocks
6.2 Error messages
6.3 Machine compatibility
6.4 Special versions
6.5 Nomenclature
6.6 Units
6.7 Charge group codes
6.8 Boundary conditions and periodicity
6.9 Writing and reading files
6.10 Generation of hydrogen atom coordinates
6.11 How to list topology building blocks and force field parameters
6.12 What to do when SHAKE fails
6.13 Removal of center of mass motion
6.14 Performing translational and rotational least squares fit
6.15 Transformation between Cartesian and oblique contravariant coordinates
6.16 Crystallographic B-factors and root mean square fluctuations
6.17 Principal axes transformation
6.18 Definition of hydrogen bonds
6.19 Time correlation functions

7. GROMOS Program descriptions with tutorial examples
   7.1 Introduction and overview of the examples
   7.2 Topology builders
   7.3 Coordinate resequencers and reformatters
   7.4 Coordinate generators
   7.5 Minimizers and simulators
   7.6 Analysers
   7.7 Coordinate reformatters

8. More sophisticated examples

9. Modification of programs or force field by users
   9.1 Introduction
   9.2 Array sizes
   9.3 Data
   9.4 Adding residue topology building blocks
   9.5 Adding or changing atom types and force field parameters
   9.6 Functions

10. Installation of GROMOS
    10.1 Introduction
    10.2 GROMOS on 32-bit machines
    10.3 GROMOS on 60- or 64-bit machines
    10.4 GROMOS on a Cray with vectorized subroutines
10.5 GROMOS on a Cyber 205 with vectorized subroutines

10.6 GROMOS on FPS machines

10.7 GROMOS on a CONVEX with vectorized subroutines

10.8 GROMOS on a Fujitsu VP with vectorized subroutines

11. What GROMOS cannot do

12. Future developments

13. Literature

14. List of occurring symbols

15. Overview of subroutine calls: LINKING of GROMOS

16. Program and subroutine headings
INTRODUCTION

GROMOS is an acronym of the GROningen MoLecular Simulation computer program package, which has been developed for the dynamic modelling of (bio)molecules. More specifically GROMOS has the following basic capabilities:

1. Simulation of proteins or arbitrary molecules using the molecular dynamics (MD) or stochastic dynamics (SD) method.

2. Energy minimisation (EM) of these molecules.

3. Analysis of molecular conformations obtained by experiment (X-ray, 2D-NMR), by model building or by computer simulation.

GROMOS is made available under the following standard conditions:

1. The programs shall be used for scientific purposes only, excluding industrial or commercial purposes.

2. Proper acknowledgement shall be made to the author of the programs in publications resulting from the use of these programs.

3. The programs shall not be made available to users outside the recipient's laboratory, unless written consent is obtained.

Any deviation from these conditions must be negotiated with the copyright owners.
The structure of this GROMOS manual is the following.

Chapter 1 contains a brief overview of the purpose, scope and limitations of dynamic modelling techniques for molecular systems and the GROMOS package and force fields.

Chapters 2 to 6 contain a systematic description of GROMOS. Chapter 2 gives the basic concepts and formulae which are used in GROMOS, together with a description of the GROMOS force fields. Chapter 3 describes the data structure of GROMOS and Chapter 4 lists the standard GROMOS files. Chapter 5 contains a global description of the architecture of GROMOS and lists the basic capabilities of the various programs. Chapter 6 discusses a whole variety of technical details of GROMOS.

Chapters 7 to 10 form a guided introduction to GROMOS. In chapter 7 the programs are described at the hand of tutorial examples. Chapter 8 is reserved for more sophisticated examples. Chapter 9 discusses how GROMOS programs or subroutines or the GROMOS force fields can be modified by users. Chapter 10 contains guidelines for installation of GROMOS.

In chapters 11 and 12 it is briefly mentioned what GROMOS cannot do and what it may be able to do in future.

Chapter 13 contains references to relevant literature. Chapter 14 consists of a list of GROMOS symbols and Chapter 15 yields an overview of subroutine calls in GROMOS which is useful when linking object modules.

Finally, we note that all GROMOS programs and subroutines contain a heading comment which precisely describes their input and output. The GROMOS tape contains a file PROCOM.TXT which contains these program and subroutine headings in alphabetical order. A print-out of this file should be added to this manual as chapter 16.
1. PURPOSE, SCOPE AND LIMITATIONS

1.1 Dynamic modelling of molecular systems in chemistry and physics

During the past thirty years the computer has taken an increasingly prominent position in science. This is due to the rapid increase of computer power. Every six to seven years the ratio of performance to price has increased by a factor of ten. This development has paved the way for simulating in atomic detail a variety of physical processes on a computer. Computer simulation is a powerful tool to predict molecular properties that are inaccessible to experiments once the reliability of the molecular models, force fields and computational procedures has been established by comparison of simulated properties with known experimental ones. This may lead to the design of substances or molecules that possess specific properties useful in practical applications. Here, one may think of applications in drug or vaccine design, in protein engineering or in material science. The common approach to modelling a molecular system on a computer is a static one. For example, quantum calculations yield an equilibrium charge distribution; Molecular Mechanics calculations yield one or a few minimum energy conformations of a molecule; on a graphics device molecules are studied in terms of fixed conformations.

However, a molecular system at room temperature is by no means of static character. A system of interacting atoms traverses multiple minima of the potential energy surface. One would like to know the multidimensional distribution function of all atomic coordinates and its development in time. This knowledge can never be complete. Only parts of configuration space can be searched for relevant low (free) energy conformations. The computer simulation technique of Molecular Dynamics
provides the possibility to scan that part of configuration space that is accessible to the molecular system at the given temperature.

Static modelling techniques are completely inadequate to describe the properties of a system in a number of applications. Examples are the behaviour of liquid water and its influence on the conformation of a solute, and the calculation of quantities like entropy and free energy. The latter determine such properties as the binding strength of small drug molecules to large acceptor molecules, which is crucial in the process of drug design.

Therefore dynamic modelling techniques are a very promising new tool in the field of (bio)molecular chemistry and physics.
1.2 Overview of GROMOS

The GROMOS package has originally been written for simulation of protein molecules. In the course of time it has developed into a general purpose molecular simulation package.

GROMOS contains six types of programs:

1. Programs that build a molecular topology.

PROMT : generates a molecular topology by plain reading; useful for small molecules or when a (formatted) molecular topology has been edited (changed by hand) or when special interaction parameters are to be used.

PROGMT : generates a molecular topology from building blocks like amino acid residue topologies or nucleotide or glycosidic topologies or given special molecule topologies, using the amino acid or nucleotide or glucose unit sequence.

PROMMT : merges two molecular topology files.

PROCMT : allows for changes in a molecular topology file by producing a formatted molecular topology, or removes specific parts (atoms) from a molecular topology file, according to a distance criterion.

2. Programs that transform a given atom coordinate sequence to the atom sequence in the molecular topology, and to the GROMOS coordinate format.

PROBRK : reads Brookhaven tape format.

PROCSC1 : changes atom sequence and format.

PROCSC2 : changes atom sequence and format, transforms from oblique to cartesian coordinates.
3. Programs that generate atom coordinates

PROGCA: generates cartesian coordinates from internal coordinates (bond-lengths, angles and dihedrals).

PROSSC: substitutes amino acid side-chain coordinates.

PROGCH: generates hydrogen atom coordinates (polar hydrogens).

PROGWH: idem, but for waters.

PROCRY: performs crystal symmetry transformations.

PROBOX: puts solute molecule(s) in box with solvent molecules.

PROION: substitutes charged ions for solvent molecules.

4. Programs that perform simulations.

PROEM: energy minimisation (steepest descents, conjugate gradients).

PROMD: molecular dynamics (classical, constant temperature or constant pressure).

PROSD: stochastic dynamics.

5. Programs that analyse configurations or sequences of configurations.

PROAVX: averages solute atom positions.

PROAVQ: averages solute internal coordinates.

PROAJC: averages NMR J-coupling constants.

PRODR: averages atom-atom distance restraints.

PROAVN: averages number of neighbour atoms.

PROAHB: averages hydrogen bonds.

PROMHBD: monitors hydrogen bonds (lifetimes).

PROCOC: averages solute or solvent site occupancies.

PROAVS: averages solvent atom positions.

PROCOX: compares two solute configurations: atom positions.

PROCOQ: compares two solute configurations: internal coordinates.
6. Programs that merge or reduce coordinate files or transform atom coordinates to a special format (interfacing).

PROMCF : merges or reduces coordinate files.

PROPSF : converts to oblique contra-variant fractional coordinates.

PROPDF : converts to master coordinate file format (display).
1.3 Applications in chemistry and physics

Interesting applications of GROMOS86 (the latest version of GROMOS) in chemistry and physics are:
- prediction of the dependence of a molecular conformation on the type of environment (water, apolar solvent, crystal, etc.);
- calculation of relative binding constants by evaluating free energy differences between various molecular complexes;
- prediction of energetic and structural changes caused by modification of amino acids in enzymes or of base pairs in DNA;
- derivation of three-dimensional (3D) molecular structure on the basis of 2D-NMR data by using restrained MD techniques;
- dynamic modelling of molecular complexes by searching configuration space using MD;
- prediction of properties of materials under extreme conditions of temperature and pressure, which may be experimentally inaccessible.
1.4 Philosophy underlying GROMOS structure

The GROMOS package is meant for use in a scientific environment, which may be characterized by a continuously changing flow of users, who all have different application of simulation methods in their field, ranging from glasses and liquid crystals to crystals and solutions of biomolecules. Therefore, GROMOS has been developed on the following principles.
- Transparency of code, so that modification is easy.
- Modular architecture, so that only parts of it can be used in changing combinations and subroutines may be replaced by ones written by the user.
- Independence of the code of the force field that is used.
- Independence of the code of the computer hardware that is available.

GROMOS consists of about 45000 lines of standard FORTRAN (66) code with a few extensions to FTN77. This property makes it run on any machine (CDC-, IBM-, DEC-, VAX-, Amdahl-, Honeywell-, Univac-, FPS-series, Cyber-205, Cray, etc.). GROMOS contains special optimized routines for modern vector-supercomputers (Cray, Cyber-205, Convex, Fujitsu). It contains programs for the analysis of MD trajectories and structures. It does not provide graphics routines, but interfacing to graphics packages is straightforward.

GROMOS is a batch-oriented package. This property is related to the building block philosophy on which its architecture is based. The about 100 independent building blocks (programs, subroutines) can be combined in a great variety of ways. It is up to the user to choose the combination which fits the task he wants to perform. This set-up requires some thinking and understanding from the user, but returns for that a large amount of
flexibility. It often happens that a new combination of building blocks can perform a task which was not thought of when GROMOS was developed.

GROMOS is delivered with 53 examples (command-, input- and output-files) covering the most important applications of all 39 programs.

1.5 GROMOS force fields

The GROMOS package comes with the GROMOS force fields. The general GROMOS force field (C-version) has been developed for application to aqueous or apolar solutions of proteins, nucleotides and sugars. However, a gas phase version (D-version) for simulation of isolated molecules is also available.

The quality of the GROMOS force field should be judged from the literature concerning its application to chemical systems.
Molecular Dynamics (MD) computer simulations are rather computer time demanding. In order to obtain an impression of the computing effort needed to simulate various systems a benchmark has been performed using GROMOS on a variety of computers. Results are given on two molecules, cyclosporin and DNA, in various environments. The numbers in Table 1.6.1 are seconds CPU time required for 10 steps of MD including one reconstruction of the charge group neighbour list. This corresponds to 0.02 picoseconds of simulation. Since the computing effort is mainly determined by the calculation of the non-bonded interaction, 10 steps of energy minimization (EM) or stochastic dynamics (SD) would require about the same computer time as given in Table 1.6.1.
<table>
<thead>
<tr>
<th>Computer</th>
<th>CPA vac</th>
<th>CPA water</th>
<th>CPA vac</th>
<th>DNA vac</th>
<th>DNA vac</th>
<th>DNA vac</th>
<th>DNA vac</th>
</tr>
</thead>
<tbody>
<tr>
<td>VMS4.2 (Frankfurt)</td>
<td>27</td>
<td>91</td>
<td>1646</td>
<td>848</td>
<td>144</td>
<td>3918</td>
<td></td>
</tr>
<tr>
<td>VAX 750 (Groningen)</td>
<td>38</td>
<td>105</td>
<td>1630</td>
<td>920</td>
<td>167</td>
<td>3786</td>
<td></td>
</tr>
<tr>
<td>VAX 780 (Bracknell)</td>
<td>25</td>
<td>68</td>
<td>1010</td>
<td>571</td>
<td>106</td>
<td>2504</td>
<td></td>
</tr>
<tr>
<td>VAX 8300 (Groningen)</td>
<td>24</td>
<td>80</td>
<td>1465</td>
<td>706</td>
<td>117</td>
<td>3518</td>
<td></td>
</tr>
<tr>
<td>VAX 8600 (Groningen)</td>
<td>5.2</td>
<td>16.3</td>
<td>291</td>
<td>169</td>
<td>25</td>
<td>681</td>
<td></td>
</tr>
<tr>
<td>FPS 164 (Bracknell)</td>
<td>9.1</td>
<td>28.3</td>
<td>443</td>
<td>258</td>
<td>45</td>
<td>1109</td>
<td></td>
</tr>
<tr>
<td>FPS 264 (Bracknell)</td>
<td>2.2</td>
<td>7.5</td>
<td>102</td>
<td>60</td>
<td>11</td>
<td>260</td>
<td></td>
</tr>
<tr>
<td>CONVEX-XP (note 1)</td>
<td>4.5</td>
<td>16.4</td>
<td>291</td>
<td>148</td>
<td>25</td>
<td>689</td>
<td></td>
</tr>
<tr>
<td>CONVEX-XP (note 3)</td>
<td>3.0</td>
<td>8.9</td>
<td>92</td>
<td>73</td>
<td>15</td>
<td>277</td>
<td></td>
</tr>
<tr>
<td>IBM3090 (Orsay)</td>
<td>0.94</td>
<td>4.3</td>
<td>46</td>
<td>41</td>
<td>4.9</td>
<td>168</td>
<td></td>
</tr>
<tr>
<td>CRAY-2 (note 3)</td>
<td>0.77</td>
<td>2.3</td>
<td>21</td>
<td>17</td>
<td>3.9</td>
<td>53</td>
<td></td>
</tr>
<tr>
<td>Cyber-205 (note 1)</td>
<td>1.52</td>
<td>6.2</td>
<td>100</td>
<td>47</td>
<td>7.8</td>
<td>244</td>
<td></td>
</tr>
<tr>
<td>Cyber-205 (note 2)</td>
<td>1.47</td>
<td>6.2</td>
<td>105</td>
<td>45</td>
<td>7.5</td>
<td>234</td>
<td></td>
</tr>
<tr>
<td>Cyber-205 (note 3)</td>
<td>0.68</td>
<td>3.8</td>
<td>19.4</td>
<td>6.7</td>
<td>3.5</td>
<td>48</td>
<td></td>
</tr>
<tr>
<td>VP200 (Amsterdam)</td>
<td>0.87</td>
<td>2.7</td>
<td>44.3</td>
<td>26.5</td>
<td>4.5</td>
<td>109</td>
<td></td>
</tr>
<tr>
<td>VP200 (Orsay)</td>
<td>0.43</td>
<td>1.5</td>
<td>11.0</td>
<td>5.8</td>
<td>2.0</td>
<td>31</td>
<td></td>
</tr>
</tbody>
</table>

Note 1: standard GROMOS, no vector option
Note 2: standard GROMOS, vector compiler
Note 3: vectorised version, vector compiler
*: version using additional long-range force, expected 10% slower
The meaning of the abbreviations in the column headings is the following:

**CPA** = cyclosporin A (cyclic undecapeptide) 90 atoms

**CPA vac** = one CPA in vacuum

**CPA solv** = one CPA in periodic box (truncated octahedron) including 594 simple solvent molecules representing carbon tetrachloride.

Box size 5.9 nm

**CPA water** = one CPA in periodic box (truncated octahedron) including 764 water molecules. Box size 3.66 nm

**DNA** = a 12-base pair fragment of deoxyribonucleic acid, 542 atoms and 22 counterions (sodium)

**DNA vac** = one DNA in vacuum

**DNA cryst** = 4 DNA molecules and 1220 water molecules in crystal symmetry.

Periodic cell dimensions: 2.49 x 4.04 x 6.62 nm

**Rcut** = Cut-off range in nm for short-range interactions included in neighbour list, evaluated every step

**Gpair** = number of atom charge groups involved in neighbour list construction

(Note: in all periodic-box simulations coupling to a bath of constant temperature and pressure has been applied)
1.7 Limitations

When applying MD to simulate a particular system, a number of preliminary questions have to be answered and choices related to the level of accuracy must be made. The assumptions and approximations that are made with respect to the molecular model and the computational procedures will determine the accuracy of the results obtained. Clearly there are limitations to the usefulness of application of simulation techniques. These will be briefly discussed below.

1. Since Newton's equations of motion are solved in a MD simulation, a classical description must be appropriate for the phenomena to be studied. Generally when considering a molecular system at room temperature quantum effects will not play a significant role as long as no covalent bonds are broken, etc.

2. With the use of modern computers the length of a MD simulation extends from a few tens of picoseconds up till nanoseconds, depending on the size of the system. This means that the time scale of a process that can be simulated at the atomic level, is limited. For the simulation of activated processes special techniques are available, which require the pathway of the process to be known.

3. Only a limited number of atoms can be simulated, typically up till 10,000-20,000 atoms. The question is how many atoms are essentially involved in the phenomena to be studied. Atomic degrees of freedom that are not essential for an adequate description of the phenomenon being studied, may be removed by applying constraints, or stochastic techniques in combination with potentials of mean force, or the extended wall region boundary condition.
4. Last but not least the interaction function potential or force field that is used, will determine the accuracy of the obtained simulation results. A great variety of molecular models and force fields for molecular systems under various conditions is available. The choice of a particular force field should depend on the system properties one is interested in. Some applications require more refined force fields than others. Moreover, there should be a balance between the level of accuracy or refinement of different parts of a molecular model. Otherwise the computer effort put into a very detailed and accurate part of the calculation may easily be wasted due to the distorting effects of the crude parts of the model.

Although computer simulation is a very powerful technique to study the properties of molecular systems at the atomic level, one should bear in mind the various assumptions and approximations that are made and be aware of the limitations of the method.
2. MODELLING OF SYSTEMS AND DESCRIPTION OF THE GROMOS FORCE FIELD

2.1 Introduction

In this chapter the molecular model, the force field and the computational procedures used in GROMOS are described. The GROMOS force field is described in Chapter 2.2-7. The application of special forces is discussed in Chapter 2.8. The use of constraints, such as bond-length constraints, is discussed in Chapter 2.9. Chapter 2.10 contains a description of the various boundary conditions that can be applied. Chapter 2.11-14 contains a description of various computational procedures, such as energy minimisation (Chapter 2.11), molecular dynamics (Chapter 2.12) and stochastic dynamics (Chapter 2.14). Chapter 2.13 describes a procedure to calculate free energy differences.

The GROMOS force field has the following form:

\[
V(\{\vec{R}_i\}) = V(\vec{R}_1, \vec{R}_2, \ldots, \vec{R}_N) = \sum_{n=1}^{N_b} \frac{1}{2} k_n (\theta_n - \theta_0)^2 + \sum_{n=1}^{N_\theta} \frac{1}{2} k_\theta (\phi_n - \phi_0)^2 + \sum_{n=1}^{N_\xi} \frac{1}{2} k_\xi (\xi_n - \xi_0)^2 + \sum_{n=1}^{N_\chi} k_\chi [1 + \cos(n_\chi, \phi_n, - \delta_n)] + \sum_{n=1}^{N_{\text{at}}} \left[ C_{12}(1, n)/r_{1j}^{12} - C_6(1, n)/r_{1j}^6 + q_i q_j / (4 \pi \varepsilon_0 r_{1j}) \right] S(r_{1j}) + \text{special terms}
\]  

(2.1.1)
The Cartesian position vectors of the $1 = 1, \ldots, N_{\text{at}}$ atoms in the system are denoted by $r_1, r_2, \ldots, r_{N_{\text{at}}}$. The terms represent: covalent bond stretching interactions (Chapter 2.3), bond-angle bending interactions (Chapter 2.4), harmonic so-called improper (out-of-plane, out-of-tetrahedral configuration) dihedral bending interactions (Chapter 2.5), sinusoidal dihedral torsion interactions (Chapter 2.6) and non-bonded van der Waals' and electrostatic (Coulomb) interactions (Chapter 2.7). The special terms, describing position restraining or distance restraining, are discussed in Chapter 2.6.

Currently there exist five versions of the **GROMOS force field**. The **C-versions** are the basic force fields designed for molecules in solution or in crystalline form. The **D-versions** are derived from the C-versions in order to be used for simulating molecules in vacuo. The atomic charges and van der Waals' parameters are changed such that charged atom groups are neutralized while maintaining the hydrogen-bonding capacity of the individual atoms. The GROMOS force field files are described in Chapter 4.1. Below, the parameters of the latest versions 37C8 and 37D8 will be given. The old versions (26C1, 37C2 and 37D2) will not be discussed.
2.2 **Atom types**

A molecular model contains three types of information:

a. Atomic mass $m_i$ of the atom with sequence number $i (-1, \ldots, N_{at})$.

b. Atomic parameters concerning the non-bonded interaction, such as atomic charge $q_i$, and for atom pairs $(i,j)$ the van der Waals' parameters $C_6(i,j)$ and $C_{12}(i,j)$.

c. Parameters concerning the covalent bond related, so-called bonded interactions, such as $K_n$, $\theta_0$, $K_{\theta}$, $\phi_0$, $K_{\phi}$, $\epsilon_n$, $r_n$, and $\delta_n$.

In the first GROMOS force field, denoted by the code 26C1, 26 atom types were defined. The atom types, represented by an integer atom code IAC running from 1 to 26, determined the force field parameters of all the three types a, b and c in the following way.

a. The atomic mass $m_i$ of atom with sequence number $i$ is equal to the mass $m(IAC(i))$ of an atom of type IAC.

b. The van der Waals' parameters $C_6(i,j)$ and $C_{12}(i,j)$ of the pair of atoms with sequence numbers $i$ and $j$ are equal to $C_6(IAC(i), IAC(j))$ and $C_{12}(IAC(i), IAC(j))$, so are a function of the pair of integer atom codes (atom types) IAC($i$) and IAC($j$). The charge $q_i$ is given for each atom, so is a function of the atomic sequence number $i$.

c. The bonded interaction parameters are determined by the integer atom codes IAC (atom types) of part of the atoms involved in the bond, bond angle or (improper) dihedral.
For bond \( n \) between atoms \( i \) and \( j \), we have \( K_{D_n}^{b_0} (IAC(i), IAC(j)) \) and \( b_0_n (IAC(i), IAC(j)). \)

- For bond angle \( n \) between atoms \( i, j \) and \( k \), we have \( K_{\theta_n} (IAC(i), IAC(j), IAC(k)) \) and \( \theta_0_n (IAC(i), IAC(j), IAC(k)). \)

- For improper dihedral \( n \) between atoms \( i, j, k \) and \( l \), the parameters are only determined by the atom types of atoms \( i \) and \( l \), so \( K_{\epsilon_n} (IAC(i), IAC(l)) \) and \( \epsilon_0_n (IAC(i), IAC(l)). \)

- For dihedral \( n \) between atoms \( i, j, k \) and \( l \), the parameters are only determined by the atom types of atoms \( j \) and \( k \), so \( K_{\eta_n} (IAC(j), IAC(k)), \eta_n (IAC(j), IAC(k)) \) and \( \eta_0_n (IAC(j), IAC(k)). \)

Since the atom type defines so many different quantities in this way, it was decided to decouple information concerning mass and non-bonded parameters on the one hand from parameters concerning bonded interactions on the other in the newer GROMOS force fields. However, the old system may still be used (see program PROCMT).

In the current GROMOS force fields, denoted by the codes 37C4 (general version) and 37D4 (vacuum version), the 37 atom types basically determine the atomic mass and van der Waals' parameters. The atom type may also be used to determine part of the bonded interaction parameters.

The 37 atom types are the following
<table>
<thead>
<tr>
<th>IAC</th>
<th>TYPE</th>
<th>MASS</th>
<th>DESCRIPTION</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>O</td>
<td>15.9994</td>
<td>carbonyl oxygen (C=O)</td>
</tr>
<tr>
<td>2</td>
<td>OM</td>
<td>15.9994</td>
<td>carboxyl oxygen (CO⁻)</td>
</tr>
<tr>
<td>3</td>
<td>OA</td>
<td>15.9994</td>
<td>hydroxyl oxygen (OH)</td>
</tr>
<tr>
<td>4</td>
<td>OW</td>
<td>15.9994</td>
<td>water oxygen</td>
</tr>
<tr>
<td>5</td>
<td>N</td>
<td>14.0067</td>
<td>peptide nitrogen (N or NH)</td>
</tr>
<tr>
<td>6</td>
<td>NT</td>
<td>14.0067</td>
<td>terminal nitrogen (NH₂)</td>
</tr>
<tr>
<td>7</td>
<td>NL</td>
<td>14.0067</td>
<td>terminal nitrogen (NH₃)</td>
</tr>
<tr>
<td>8</td>
<td>NR5</td>
<td>14.0067</td>
<td>aromatic N (5⁴-ring, 2 bonds)</td>
</tr>
<tr>
<td>9</td>
<td>NR5*</td>
<td>14.0067</td>
<td>aromatic N (5-ring, 3 bonds)</td>
</tr>
<tr>
<td>10</td>
<td>NP</td>
<td>14.0067</td>
<td>porphyrin nitrogen</td>
</tr>
<tr>
<td>11</td>
<td>C</td>
<td>12.0110</td>
<td>bare carbon (peptide, C=O, C-N)</td>
</tr>
<tr>
<td>12</td>
<td>CH₁</td>
<td>13.0190</td>
<td>aliphatic CH₃-group</td>
</tr>
<tr>
<td>13</td>
<td>CH₂</td>
<td>14.0270</td>
<td>aliphatic CH₂-group</td>
</tr>
<tr>
<td>14</td>
<td>CH₃</td>
<td>15.0350</td>
<td>aliphatic CH₃-group</td>
</tr>
<tr>
<td>15</td>
<td>CR₅₁</td>
<td>13.0190</td>
<td>aromatic CH-group (5-ring)</td>
</tr>
<tr>
<td>16</td>
<td>CR₆¹</td>
<td>13.0190</td>
<td>aromatic CH-group (6⁴-ring)</td>
</tr>
<tr>
<td>17</td>
<td>CB</td>
<td>12.0110</td>
<td>bare carbon (5-, 6-ring)</td>
</tr>
<tr>
<td>18</td>
<td>H</td>
<td>1.0080</td>
<td>hydrogen bonded to nitrogen</td>
</tr>
<tr>
<td>19</td>
<td>HO</td>
<td>1.0080</td>
<td>hydroxyl hydrogen</td>
</tr>
<tr>
<td>20</td>
<td>HW</td>
<td>1.0080</td>
<td>water hydrogen</td>
</tr>
<tr>
<td>21</td>
<td>HS</td>
<td>1.0080</td>
<td>hydrogen bonded to sulfur</td>
</tr>
<tr>
<td>22</td>
<td>S</td>
<td>32.0600</td>
<td>sulfur</td>
</tr>
<tr>
<td>#</td>
<td>Atom</td>
<td>Type</td>
<td>Description</td>
</tr>
<tr>
<td>---</td>
<td>------</td>
<td>-----------------------</td>
<td>--------------------------------------------------</td>
</tr>
<tr>
<td>23</td>
<td>Fe</td>
<td>55.8470</td>
<td>Iron</td>
</tr>
<tr>
<td>24</td>
<td>Zn</td>
<td>65.3700</td>
<td>Zinc</td>
</tr>
<tr>
<td>25</td>
<td>Nα</td>
<td>14.0067</td>
<td>Arg NH (NH2)</td>
</tr>
<tr>
<td>26</td>
<td>Nε</td>
<td>14.0067</td>
<td>Arg NE (NH)</td>
</tr>
<tr>
<td>27</td>
<td>P</td>
<td>30.9738</td>
<td>Phosphor</td>
</tr>
<tr>
<td>28</td>
<td>Os</td>
<td>15.9994</td>
<td>Sugar or ester oxygen</td>
</tr>
<tr>
<td>29</td>
<td>Cs1</td>
<td>13.0190</td>
<td>Sugar CH-group</td>
</tr>
<tr>
<td>30</td>
<td>NR6</td>
<td>14.0067</td>
<td>Aromatic N (6-ring, 2 bonds)</td>
</tr>
<tr>
<td>31</td>
<td>NR6*</td>
<td>14.0067</td>
<td>Aromatic N (6-ring, 3 bonds)</td>
</tr>
<tr>
<td>32</td>
<td>Cs2</td>
<td>14.0270</td>
<td>Sugar CH2-group</td>
</tr>
<tr>
<td>33</td>
<td>Si</td>
<td>28.0800</td>
<td>Silicon</td>
</tr>
<tr>
<td>34</td>
<td>Na</td>
<td>22.9898</td>
<td>Sodium (1+)</td>
</tr>
<tr>
<td>35</td>
<td>Cl</td>
<td>35.4530</td>
<td>Chlorine (1+)</td>
</tr>
<tr>
<td>36</td>
<td>Ca</td>
<td>40.0800</td>
<td>Calcium (2+)</td>
</tr>
<tr>
<td>37</td>
<td>Mg</td>
<td>24.3050</td>
<td>Magnesium (2+)</td>
</tr>
</tbody>
</table>

The atom type information can be found in the residue topology building block files RT***.DAT and in the interaction function parameter files IFP***.DAT. The information is printed in more readable form in the output of program PROGNT, see example files OUTGNT***.LIS.
2.3 Covalent bond forces

The term in the interaction potential (2.1.1) that represents the covalent bond stretching interaction reads:

$$\sum_{n=1}^{N_b} b_n (b_n - b_{0_n}) = \sum_{n=1}^{N_b} \frac{1}{2} K_n [b_n - b_{0_n}]^2$$  \hspace{1cm} (2.3.1)

The summation generally runs over all $n = 1, \ldots, N_b$ covalent bonds in the system. The actual bond-length of the $n$-th bond between atoms with sequence numbers $i$ and $j$ is denoted by $b_n$. It is given by

$$b_n = r_{ij}$$  \hspace{1cm} (2.3.2)

where

$$\hat{r}_{ij} = \hat{r}_i - \hat{r}_j$$  \hspace{1cm} (2.3.3)

and

$$r_{ij} = (\hat{r}_{ij} \cdot \hat{r}_{ij})^{1/2} = |\hat{r}_{ij}|$$  \hspace{1cm} (2.3.4)

The forces on atoms $i$ and $j$ due to the $n$-th term in formula (2.3.1) are

$$F_i = \frac{dV_{b_n}}{db_n} \cdot \frac{3b_n}{\hat{r}_{ij}}$$

$$= - K_n [b_n - b_{0_n}] \cdot (\hat{r}_{ij} / r_{ij})$$  \hspace{1cm} (2.3.5)
\[ F_j = - F_i \] (2.3.5)

The interaction energy and forces are calculated in subroutine COBOND. For reasons of ease of analysis, the list of \( N_b \) covalent bonds is split into two lists, one of bonds involving hydrogen atoms, and one involving the other bonds. These lists are kept in the molecular topology file (Chapter 3.3.2). The first list contains NBONH bonds involving hydrogen atoms. Three items are stored: IBH, JBH[1..NBONH] are the atom sequence numbers of the atoms forming bond \( i \rightarrow j \) as a function of the bond sequence number \( n \), and ICBH[1..NBONH] is the bond-type code, denoting the parameters \( b_{bn} \) and \( b_{bn}^0 \), as a function of the bond sequence number \( n \). The force field parameters \( b_D \) and \( b_0 \) for the various types of covalent bonds are stored in CB[1..NBTY] and BO[1..NBTY] as a function of the bond-type code (ICBH or ICB). They can be found in the interaction function parameter files IFP**.DAT, and are printed in more readable form in the output of program PROGRT, see example files OUTGRT**.LIS. For the GROMOS force fields 37C and 37D they are listed in Table 2.3.1. The list for the bonds involving no hydrogen atoms contains corresponding items denoted by IB, JB, ICB[1..NBON].
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>C - O</td>
<td>1200.0</td>
<td>1.230</td>
</tr>
<tr>
<td>2</td>
<td>C - DM</td>
<td>1000.0</td>
<td>1.250</td>
</tr>
<tr>
<td>3</td>
<td>C - DA</td>
<td>900.0</td>
<td>1.560</td>
</tr>
<tr>
<td>4</td>
<td>C - N</td>
<td>1000.0</td>
<td>1.330</td>
</tr>
<tr>
<td>5</td>
<td>C - NT</td>
<td>900.0</td>
<td>1.330</td>
</tr>
<tr>
<td>6</td>
<td>C - NL</td>
<td>900.0</td>
<td>1.330</td>
</tr>
<tr>
<td>7</td>
<td>CH1 - DA</td>
<td>800.0</td>
<td>1.430</td>
</tr>
<tr>
<td>8</td>
<td>CH1 - N</td>
<td>900.0</td>
<td>1.470</td>
</tr>
<tr>
<td>9</td>
<td>CH1 - NT</td>
<td>900.0</td>
<td>1.470</td>
</tr>
<tr>
<td>10</td>
<td>CH1 - NL</td>
<td>900.0</td>
<td>1.470</td>
</tr>
<tr>
<td>11</td>
<td>CH1 - C</td>
<td>800.0</td>
<td>1.530</td>
</tr>
<tr>
<td>12</td>
<td>CH1 - CH1</td>
<td>800.0</td>
<td>1.530</td>
</tr>
<tr>
<td>13</td>
<td>CH2 - DM</td>
<td>900.0</td>
<td>1.430</td>
</tr>
<tr>
<td>14</td>
<td>CH2 - DA</td>
<td>800.0</td>
<td>1.430</td>
</tr>
<tr>
<td>15</td>
<td>CH2 - N</td>
<td>900.0</td>
<td>1.470</td>
</tr>
<tr>
<td>16</td>
<td>CH2 - NT</td>
<td>900.0</td>
<td>1.470</td>
</tr>
<tr>
<td>17</td>
<td>CH2 - NL</td>
<td>900.0</td>
<td>1.470</td>
</tr>
<tr>
<td>18</td>
<td>CH2 - C</td>
<td>800.0</td>
<td>1.530</td>
</tr>
<tr>
<td>19</td>
<td>CH2 - CH1</td>
<td>800.0</td>
<td>1.530</td>
</tr>
<tr>
<td>20</td>
<td>CH2 - CH2</td>
<td>800.0</td>
<td>1.530</td>
</tr>
<tr>
<td>21</td>
<td>CH3 - N</td>
<td>900.0</td>
<td>1.470</td>
</tr>
<tr>
<td>22</td>
<td>CH3 - C</td>
<td>800.0</td>
<td>1.530</td>
</tr>
<tr>
<td>23</td>
<td>CH3 - CH1</td>
<td>800.0</td>
<td>1.530</td>
</tr>
<tr>
<td>24</td>
<td>CH3 - CH2</td>
<td>800.0</td>
<td>1.530</td>
</tr>
<tr>
<td>25</td>
<td>CR51 - NR5</td>
<td>1000.0</td>
<td>1.330</td>
</tr>
<tr>
<td>26</td>
<td>CR51 - NR5*</td>
<td>1000.0</td>
<td>1.330</td>
</tr>
<tr>
<td>27</td>
<td>CR61 - CH2</td>
<td>1000.0</td>
<td>1.390</td>
</tr>
<tr>
<td>28</td>
<td>CR61 - CR61</td>
<td>1000.0</td>
<td>1.390</td>
</tr>
<tr>
<td>29</td>
<td>CB - O</td>
<td>1200.0</td>
<td>1.230</td>
</tr>
<tr>
<td>30</td>
<td>CB - DA</td>
<td>900.0</td>
<td>1.360</td>
</tr>
<tr>
<td>31</td>
<td>CB - NT</td>
<td>900.0</td>
<td>1.330</td>
</tr>
<tr>
<td>32</td>
<td>CB - NR5</td>
<td>1000.0</td>
<td>1.330</td>
</tr>
<tr>
<td>33</td>
<td>CB - NR5*</td>
<td>1000.0</td>
<td>1.330</td>
</tr>
<tr>
<td>34</td>
<td>CB - NP</td>
<td>1000.0</td>
<td>1.380</td>
</tr>
<tr>
<td>35</td>
<td>CB - C</td>
<td>800.0</td>
<td>1.530</td>
</tr>
<tr>
<td>36</td>
<td>CB - CH2</td>
<td>800.0</td>
<td>1.530</td>
</tr>
<tr>
<td>37</td>
<td>CB - CH3</td>
<td>800.0</td>
<td>1.530</td>
</tr>
<tr>
<td>38</td>
<td>CB - CR51</td>
<td>1000.0</td>
<td>1.330</td>
</tr>
<tr>
<td>39</td>
<td>CB - CR61</td>
<td>1000.0</td>
<td>1.390</td>
</tr>
<tr>
<td>40</td>
<td>CB - CB</td>
<td>1000.0</td>
<td>1.390</td>
</tr>
<tr>
<td>41</td>
<td>H - N</td>
<td>895.0</td>
<td>1.000</td>
</tr>
<tr>
<td>42</td>
<td>H - NT</td>
<td>895.0</td>
<td>1.000</td>
</tr>
<tr>
<td>43</td>
<td>H - NL</td>
<td>895.0</td>
<td>1.000</td>
</tr>
<tr>
<td>44</td>
<td>H - NR5*</td>
<td>895.0</td>
<td>1.000</td>
</tr>
<tr>
<td>45</td>
<td>HO - DA</td>
<td>750.0</td>
<td>1.000</td>
</tr>
<tr>
<td>46</td>
<td>HW - DW</td>
<td>0.0</td>
<td>1.000</td>
</tr>
<tr>
<td>47</td>
<td>S - D</td>
<td>900.0</td>
<td>1.500</td>
</tr>
<tr>
<td>48</td>
<td>S - CH2</td>
<td>900.0</td>
<td>1.830</td>
</tr>
<tr>
<td>49</td>
<td>S - CH3</td>
<td>900.0</td>
<td>1.760</td>
</tr>
<tr>
<td>50</td>
<td>S - HS</td>
<td>750.0</td>
<td>1.330</td>
</tr>
<tr>
<td>#</td>
<td>Bond-Type</td>
<td>CB[i]</td>
<td>BO[i]</td>
</tr>
<tr>
<td>----</td>
<td>-----------</td>
<td>-------</td>
<td>-------</td>
</tr>
<tr>
<td>51</td>
<td>S-S</td>
<td>1000.0</td>
<td>2.040</td>
</tr>
<tr>
<td>52</td>
<td>FE-NR5</td>
<td>1000.0</td>
<td>1.980</td>
</tr>
<tr>
<td>53</td>
<td>FE-NP</td>
<td>1000.0</td>
<td>2.090</td>
</tr>
<tr>
<td>54</td>
<td>NZ-C</td>
<td>900.0</td>
<td>1.340</td>
</tr>
<tr>
<td>55</td>
<td>NE-H</td>
<td>895.0</td>
<td>1.000</td>
</tr>
<tr>
<td>56</td>
<td>NE-C</td>
<td>900.0</td>
<td>1.340</td>
</tr>
<tr>
<td>57</td>
<td>NE-CH2</td>
<td>900.0</td>
<td>1.470</td>
</tr>
<tr>
<td>58</td>
<td>NE-H</td>
<td>895.0</td>
<td>1.000</td>
</tr>
<tr>
<td>59</td>
<td>P-OM</td>
<td>900.0</td>
<td>1.480</td>
</tr>
<tr>
<td>60</td>
<td>P-DA</td>
<td>600.0</td>
<td>1.610</td>
</tr>
<tr>
<td>61</td>
<td>OS-CH2</td>
<td>600.0</td>
<td>1.430</td>
</tr>
<tr>
<td>62</td>
<td>OS-P</td>
<td>600.0</td>
<td>1.610</td>
</tr>
<tr>
<td>63</td>
<td>CS1-DA</td>
<td>600.0</td>
<td>1.480</td>
</tr>
<tr>
<td>64</td>
<td>CS1-NR5*</td>
<td>600.0</td>
<td>1.480</td>
</tr>
<tr>
<td>65</td>
<td>CS1-CH2</td>
<td>600.0</td>
<td>1.530</td>
</tr>
<tr>
<td>66</td>
<td>CS1-OS</td>
<td>600.0</td>
<td>1.435</td>
</tr>
<tr>
<td>67</td>
<td>CS1-CS1</td>
<td>600.0</td>
<td>1.520</td>
</tr>
<tr>
<td>68</td>
<td>NR6-CR61</td>
<td>1000.0</td>
<td>1.320</td>
</tr>
<tr>
<td>69</td>
<td>NR6-CB</td>
<td>1000.0</td>
<td>1.340</td>
</tr>
<tr>
<td>70</td>
<td>NR6*-CH2</td>
<td>800.0</td>
<td>1.480</td>
</tr>
<tr>
<td>71</td>
<td>NR6*-CH3</td>
<td>800.0</td>
<td>1.480</td>
</tr>
<tr>
<td>72</td>
<td>NR6*-CR61</td>
<td>800.0</td>
<td>1.400</td>
</tr>
<tr>
<td>73</td>
<td>NR6*-CB</td>
<td>800.0</td>
<td>1.400</td>
</tr>
<tr>
<td>74</td>
<td>NR6*-H</td>
<td>895.0</td>
<td>1.000</td>
</tr>
<tr>
<td>75</td>
<td>NR6*-CS1</td>
<td>800.0</td>
<td>1.480</td>
</tr>
<tr>
<td>76</td>
<td>CS2-DA</td>
<td>600.0</td>
<td>1.430</td>
</tr>
<tr>
<td>77</td>
<td>CS2-CR61</td>
<td>800.0</td>
<td>1.390</td>
</tr>
<tr>
<td>78</td>
<td>CS2-CB</td>
<td>800.0</td>
<td>1.390</td>
</tr>
<tr>
<td>79</td>
<td>CS2-OS</td>
<td>600.0</td>
<td>1.435</td>
</tr>
<tr>
<td>80</td>
<td>CS2-CS1</td>
<td>600.0</td>
<td>1.520</td>
</tr>
<tr>
<td>81</td>
<td>CS2-NR6*</td>
<td>800.0</td>
<td>1.400</td>
</tr>
<tr>
<td>82</td>
<td>SI-CH1</td>
<td>600.0</td>
<td>1.870</td>
</tr>
<tr>
<td>83</td>
<td>SI-OS</td>
<td>600.0</td>
<td>1.630</td>
</tr>
</tbody>
</table>

CB[i] in kcal mol⁻¹Å⁻², BO[i] in Å, the sequential code is denoted by I.
2.4 Covalent bond angle forces

The term in the interaction potential (2.4.1) that represents the bond-angle bending interaction reads

\[ V_\theta^{(n)}(\theta_n) = \sum_{n=1}^{N_\theta} \frac{1}{2} K_\theta \left[ \theta_n - \theta_0 \right]^2 \]  

(2.4.1)

The summation generally runs over all \( n = 1, \ldots, N_\theta \) bond angles in the system. The actual bond angle value of the \( n \)-th bond angle between atoms with sequence numbers \( i, j \) and \( k \) (1-j-k) is denoted by \( \theta_n \). It is given by

\[ \theta_n = \arccos \left( \frac{\vec{r}_{i,j} \cdot \vec{r}_{k,j}}{|\vec{r}_{i,j}| \cdot |\vec{r}_{k,j}|} \right) \]  

(2.4.2)

The forces on atoms \( i, j \) and \( k \) due to the \( n \)-th term in formula (2.4.1) are

\[ F_i = - \frac{\partial V_\theta^{(n)}}{\partial \theta_n} - \frac{\partial V_\theta^{(n)}}{\partial \cos \theta_n} \left( \frac{\partial \cos \theta_n}{\partial \theta_n} \right) \]  

(2.4.3)

\[ F_k = - \frac{\partial V_\theta^{(n)}}{\partial \theta_n} - \frac{\partial V_\theta^{(n)}}{\partial \cos \theta_n} \left( \frac{\partial \cos \theta_n}{\partial \theta_n} \right) \]  

(2.4.4)
\[
\vec{F}_j = -\vec{F}_1 - \vec{F}_k
\]

The interaction energy and forces are calculated in subroutine ANGLE. For reasons of ease of analysis, the list of \(N_b\) bond angles is split into two lists, one of bond angles involving hydrogen atoms, and one involving the other bond angles. These lists are kept in the molecular topology file (Chapter 3.3.2). The first list contains \(N_{THEH}\) bond angles involving hydrogen atoms. Four items are stored: ITH, JTH, KTH[1..NTHEH] are the atom sequence numbers of the atoms forming bond angle \(i\-j\-k\) as a function of the bond-angle sequence number \(n\), and ICTH[1..NTHEH] is the bond-angle type code, denoting the parameters \(K_\theta^n\) and \(\theta^*_n\), as a function of the bond angle sequence number \(n\). The force field parameters \(K_\theta\) and \(\theta^*_n\) for the various types of bond angles are stored in CT[1..NTTY] and TO[1..NTTY] as a function of the bond-angle type code (ICTH or ICT). They can be found in the interaction parameter files IFP***.DAT, and are printed in more readable form in the output of program PROGMI; see example files OUTGMI***.LIS. For the GROMOS force fields 37C and 37D, they are listed in Table 2.4.1. The list for the bond angles involving no hydrogen atoms contains corresponding items denoted by IT, JT, KT, ICT[1..NTHE].
<table>
<thead>
<tr>
<th>#</th>
<th>ANGLE-TYPE</th>
<th>CT[1]</th>
<th>TO[1]</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>HO - OA - C</td>
<td>95.0</td>
<td>109.5</td>
</tr>
<tr>
<td>2</td>
<td>HO - OA - CH1</td>
<td>95.0</td>
<td>109.5</td>
</tr>
<tr>
<td>3</td>
<td>HO - OA - CH2</td>
<td>95.0</td>
<td>109.5</td>
</tr>
<tr>
<td>4</td>
<td>HO - OA - CB</td>
<td>95.0</td>
<td>109.5</td>
</tr>
<tr>
<td>5</td>
<td>P - OA - HO</td>
<td>95.0</td>
<td>109.5</td>
</tr>
<tr>
<td>6</td>
<td>CS1 - OA - HO</td>
<td>95.0</td>
<td>109.5</td>
</tr>
<tr>
<td>7</td>
<td>CS2 - OA - HO</td>
<td>95.0</td>
<td>109.5</td>
</tr>
<tr>
<td>8</td>
<td>HW - OW - HW</td>
<td>0.0</td>
<td>109.5</td>
</tr>
<tr>
<td>9</td>
<td>CH1 - N - C</td>
<td>120.0</td>
<td>122.0</td>
</tr>
<tr>
<td>10</td>
<td>CH2 - N - C</td>
<td>120.0</td>
<td>122.0</td>
</tr>
<tr>
<td>11</td>
<td>CH2 - N - CH1</td>
<td>120.0</td>
<td>116.0</td>
</tr>
<tr>
<td>12</td>
<td>CH3 - N - C</td>
<td>120.0</td>
<td>117.0</td>
</tr>
<tr>
<td>13</td>
<td>CH3 - N - CH1</td>
<td>120.0</td>
<td>125.0</td>
</tr>
<tr>
<td>14</td>
<td>CH3 - N - CH2</td>
<td>120.0</td>
<td>125.0</td>
</tr>
<tr>
<td>15</td>
<td>H - N - C</td>
<td>70.0</td>
<td>123.0</td>
</tr>
<tr>
<td>16</td>
<td>H - N - CH1</td>
<td>90.0</td>
<td>115.0</td>
</tr>
<tr>
<td>17</td>
<td>H - N - CH2</td>
<td>90.0</td>
<td>115.0</td>
</tr>
<tr>
<td>18</td>
<td>H - N - CH3</td>
<td>90.0</td>
<td>120.0</td>
</tr>
<tr>
<td>19</td>
<td>H - NT - C</td>
<td>70.0</td>
<td>120.0</td>
</tr>
<tr>
<td>20</td>
<td>H - NT - CH1</td>
<td>90.0</td>
<td>120.0</td>
</tr>
<tr>
<td>21</td>
<td>H - NT - CH2</td>
<td>90.0</td>
<td>120.0</td>
</tr>
<tr>
<td>22</td>
<td>H - NT - CB</td>
<td>70.0</td>
<td>120.0</td>
</tr>
<tr>
<td>23</td>
<td>H - NT - H</td>
<td>60.0</td>
<td>120.0</td>
</tr>
<tr>
<td>24</td>
<td>H - NL - C</td>
<td>90.0</td>
<td>109.5</td>
</tr>
<tr>
<td>25</td>
<td>H - NL - CH1</td>
<td>90.0</td>
<td>109.5</td>
</tr>
<tr>
<td>26</td>
<td>H - NL - CH2</td>
<td>90.0</td>
<td>109.5</td>
</tr>
<tr>
<td>27</td>
<td>H - NL - H</td>
<td>80.0</td>
<td>109.5</td>
</tr>
<tr>
<td>28</td>
<td>CR51 - NR5* - CR51</td>
<td>100.0</td>
<td>108.0</td>
</tr>
<tr>
<td>29</td>
<td>CB - NR5* - CR51</td>
<td>100.0</td>
<td>108.0</td>
</tr>
<tr>
<td>30</td>
<td>FE - NR5* - CR51</td>
<td>60.0</td>
<td>125.0</td>
</tr>
<tr>
<td>31</td>
<td>CR51 - NR5* - CR51</td>
<td>100.0</td>
<td>108.0</td>
</tr>
<tr>
<td>32</td>
<td>CB - NR5* - CR51</td>
<td>100.0</td>
<td>108.0</td>
</tr>
<tr>
<td>33</td>
<td>CS1 - NR5* - CR51</td>
<td>100.0</td>
<td>126.0</td>
</tr>
<tr>
<td>34</td>
<td>CS1 - NR5* - CB</td>
<td>100.0</td>
<td>126.0</td>
</tr>
<tr>
<td>35</td>
<td>H - NR5* - CR51</td>
<td>90.0</td>
<td>126.0</td>
</tr>
<tr>
<td>36</td>
<td>H - NR5* - CB</td>
<td>90.0</td>
<td>126.0</td>
</tr>
<tr>
<td>37</td>
<td>CB - NP - CB</td>
<td>100.0</td>
<td>108.0</td>
</tr>
<tr>
<td>38</td>
<td>FE - NP - CB</td>
<td>100.0</td>
<td>126.0</td>
</tr>
<tr>
<td>39</td>
<td>OM - C - OM</td>
<td>120.0</td>
<td>126.0</td>
</tr>
<tr>
<td>40</td>
<td>O - C - O</td>
<td>120.0</td>
<td>124.0</td>
</tr>
<tr>
<td>41</td>
<td>N - C - O</td>
<td>120.0</td>
<td>124.0</td>
</tr>
<tr>
<td>42</td>
<td>NT - C - O</td>
<td>120.0</td>
<td>124.0</td>
</tr>
<tr>
<td>43</td>
<td>NL - C - O</td>
<td>120.0</td>
<td>124.0</td>
</tr>
<tr>
<td>44</td>
<td>CH1 - C - O</td>
<td>120.0</td>
<td>121.0</td>
</tr>
<tr>
<td>45</td>
<td>CH1 - C - OM</td>
<td>120.0</td>
<td>117.0</td>
</tr>
<tr>
<td>46</td>
<td>CH1 - C - O A</td>
<td>120.0</td>
<td>115.0</td>
</tr>
<tr>
<td>(i)</td>
<td>ANGLE-TYPE</td>
<td>(\text{CT}[i])</td>
<td>(\text{TO}[i])</td>
</tr>
<tr>
<td>-----</td>
<td>------------</td>
<td>----------------</td>
<td>----------------</td>
</tr>
<tr>
<td>47</td>
<td>CH1 - C - N</td>
<td>120.0</td>
<td>115.0</td>
</tr>
<tr>
<td>48</td>
<td>CH1 - C - NT</td>
<td>120.0</td>
<td>115.0</td>
</tr>
<tr>
<td>49</td>
<td>CH2 - C - D</td>
<td>120.0</td>
<td>121.0</td>
</tr>
<tr>
<td>50</td>
<td>CH2 - C - DM</td>
<td>120.0</td>
<td>117.0</td>
</tr>
<tr>
<td>51</td>
<td>CH2 - C - DA</td>
<td>120.0</td>
<td>115.0</td>
</tr>
<tr>
<td>52</td>
<td>CH2 - C - N</td>
<td>120.0</td>
<td>115.0</td>
</tr>
<tr>
<td>53</td>
<td>CH2 - C - NT</td>
<td>120.0</td>
<td>115.0</td>
</tr>
<tr>
<td>54</td>
<td>CH2 - C - NL</td>
<td>120.0</td>
<td>115.0</td>
</tr>
<tr>
<td>55</td>
<td>CH3 - C - D</td>
<td>120.0</td>
<td>121.0</td>
</tr>
<tr>
<td>56</td>
<td>CH3 - C - N</td>
<td>120.0</td>
<td>115.0</td>
</tr>
<tr>
<td>57</td>
<td>CB - C - D</td>
<td>120.0</td>
<td>121.0</td>
</tr>
<tr>
<td>58</td>
<td>CB - C - N</td>
<td>120.0</td>
<td>115.0</td>
</tr>
<tr>
<td>59</td>
<td>CB - C - NT</td>
<td>120.0</td>
<td>115.0</td>
</tr>
<tr>
<td>60</td>
<td>N2 - C - N2</td>
<td>120.0</td>
<td>120.0</td>
</tr>
<tr>
<td>61</td>
<td>NE - C - N2</td>
<td>120.0</td>
<td>120.0</td>
</tr>
<tr>
<td>62</td>
<td>C - CH1 - N</td>
<td>110.0</td>
<td>109.5</td>
</tr>
<tr>
<td>63</td>
<td>C - CH1 - NT</td>
<td>110.0</td>
<td>109.5</td>
</tr>
<tr>
<td>64</td>
<td>C - CH1 - NL</td>
<td>110.0</td>
<td>109.5</td>
</tr>
<tr>
<td>65</td>
<td>CH1 - CH1 - DA</td>
<td>110.0</td>
<td>109.5</td>
</tr>
<tr>
<td>66</td>
<td>CH1 - CH1 - N</td>
<td>110.0</td>
<td>109.5</td>
</tr>
<tr>
<td>67</td>
<td>CH1 - CH1 - NT</td>
<td>110.0</td>
<td>109.5</td>
</tr>
<tr>
<td>68</td>
<td>CH1 - CH1 - NL</td>
<td>110.0</td>
<td>109.5</td>
</tr>
<tr>
<td>69</td>
<td>CH1 - CH1 - C</td>
<td>110.0</td>
<td>109.5</td>
</tr>
<tr>
<td>70</td>
<td>CH1 - CH1 - CH1</td>
<td>110.0</td>
<td>111.0</td>
</tr>
<tr>
<td>71</td>
<td>CH2 - CH1 - CH1</td>
<td>110.0</td>
<td>109.5</td>
</tr>
<tr>
<td>72</td>
<td>CH2 - CH1 - N</td>
<td>110.0</td>
<td>109.5</td>
</tr>
<tr>
<td>73</td>
<td>CH2 - CH1 - NT</td>
<td>110.0</td>
<td>109.5</td>
</tr>
<tr>
<td>74</td>
<td>CH2 - CH1 - NL</td>
<td>110.0</td>
<td>109.5</td>
</tr>
<tr>
<td>75</td>
<td>CH2 - CH1 - C</td>
<td>110.0</td>
<td>109.5</td>
</tr>
<tr>
<td>76</td>
<td>CH2 - CH1 - CH1</td>
<td>110.0</td>
<td>111.0</td>
</tr>
<tr>
<td>77</td>
<td>CH2 - CH1 - CH2</td>
<td>110.0</td>
<td>109.5</td>
</tr>
<tr>
<td>78</td>
<td>CH3 - CH1 - DA</td>
<td>110.0</td>
<td>111.0</td>
</tr>
<tr>
<td>79</td>
<td>CH3 - CH1 - N</td>
<td>110.0</td>
<td>109.5</td>
</tr>
<tr>
<td>80</td>
<td>CH3 - CH1 - NT</td>
<td>110.0</td>
<td>109.5</td>
</tr>
<tr>
<td>81</td>
<td>CH3 - CH1 - NL</td>
<td>110.0</td>
<td>109.5</td>
</tr>
<tr>
<td>82</td>
<td>CH3 - CH1 - C</td>
<td>110.0</td>
<td>109.5</td>
</tr>
<tr>
<td>83</td>
<td>CH3 - CH1 - CH1</td>
<td>110.0</td>
<td>111.0</td>
</tr>
<tr>
<td>84</td>
<td>CH3 - CH1 - CH2</td>
<td>110.0</td>
<td>111.0</td>
</tr>
<tr>
<td>85</td>
<td>CH3 - CH1 - CH3</td>
<td>110.0</td>
<td>111.0</td>
</tr>
<tr>
<td>86</td>
<td>SI - CH1 - CH3</td>
<td>110.0</td>
<td>109.5</td>
</tr>
<tr>
<td>87</td>
<td>C - CH2 - N</td>
<td>110.0</td>
<td>109.5</td>
</tr>
<tr>
<td>88</td>
<td>C - CH2 - NT</td>
<td>110.0</td>
<td>109.5</td>
</tr>
<tr>
<td>89</td>
<td>C - CH2 - NL</td>
<td>110.0</td>
<td>109.5</td>
</tr>
<tr>
<td>90</td>
<td>CH1 - CH2 - DA</td>
<td>110.0</td>
<td>109.5</td>
</tr>
<tr>
<td>91</td>
<td>CH1 - CH2 - N</td>
<td>110.0</td>
<td>109.5</td>
</tr>
<tr>
<td>92</td>
<td>CH1 - CH2 - C</td>
<td>110.0</td>
<td>111.0</td>
</tr>
<tr>
<td>93</td>
<td>CH1 - CH2 - CH1</td>
<td>110.0</td>
<td>109.5</td>
</tr>
<tr>
<td>94</td>
<td>CH2 - CH2 - DM</td>
<td>110.0</td>
<td>109.5</td>
</tr>
<tr>
<td>95</td>
<td>CH2 - CH2 - DA</td>
<td>110.0</td>
<td>109.5</td>
</tr>
<tr>
<td>96</td>
<td>CH2 - CH2 - N</td>
<td>110.0</td>
<td>109.5</td>
</tr>
<tr>
<td>97</td>
<td>CH2 - CH2 - NT</td>
<td>110.0</td>
<td>109.5</td>
</tr>
<tr>
<td>98</td>
<td>CH2 - CH2 - NL</td>
<td>110.0</td>
<td>109.5</td>
</tr>
<tr>
<td>99</td>
<td>CH2 - CH2 - C</td>
<td>110.0</td>
<td>111.0</td>
</tr>
<tr>
<td>100</td>
<td>CH2 - CH2 - CH1</td>
<td>110.0</td>
<td>111.0</td>
</tr>
<tr>
<td>101</td>
<td>CH2 - CH2 - CH2</td>
<td>110.0</td>
<td>111.0</td>
</tr>
<tr>
<td>102</td>
<td>CH3 - CH2 - DM</td>
<td>110.0</td>
<td>109.5</td>
</tr>
<tr>
<td>103</td>
<td>CH3 - CH2 - DA</td>
<td>110.0</td>
<td>111.0</td>
</tr>
<tr>
<td>104</td>
<td>CH3 - CH2 - CH2</td>
<td>110.0</td>
<td>111.0</td>
</tr>
<tr>
<td>105</td>
<td>CH3 - CH2 - CH1</td>
<td>110.0</td>
<td>111.0</td>
</tr>
<tr>
<td>106</td>
<td>CB - CH2 - DM</td>
<td>110.0</td>
<td>109.5</td>
</tr>
<tr>
<td>107</td>
<td>CB - CH2 - CH1</td>
<td>110.0</td>
<td>111.0</td>
</tr>
<tr>
<td>I</td>
<td>ANGLE-TYPE</td>
<td>C_{T}[i]</td>
<td>T_{O}[i]</td>
</tr>
<tr>
<td>----</td>
<td>---------------</td>
<td>----------</td>
<td>----------</td>
</tr>
<tr>
<td>106</td>
<td>CB - CH2 - CH2</td>
<td>110.0</td>
<td>111.0</td>
</tr>
<tr>
<td>109</td>
<td>S - CH2 - CH1</td>
<td>110.0</td>
<td>113.0</td>
</tr>
<tr>
<td>110</td>
<td>S - CH2 - CH2</td>
<td>110.0</td>
<td>109.5</td>
</tr>
<tr>
<td>111</td>
<td>NE - CH2 - CH2</td>
<td>110.0</td>
<td>111.0</td>
</tr>
<tr>
<td>112</td>
<td>DS - CH2 - CH1</td>
<td>110.0</td>
<td>111.0</td>
</tr>
<tr>
<td>113</td>
<td>NR6* - CH2 - CH1</td>
<td>110.0</td>
<td>110.0</td>
</tr>
<tr>
<td>114</td>
<td>NR6* - CH2 - CB</td>
<td>110.0</td>
<td>109.5</td>
</tr>
<tr>
<td>115</td>
<td>NR6* - CH2 - CB1</td>
<td>110.0</td>
<td>109.5</td>
</tr>
<tr>
<td>116</td>
<td>NR6* - CR51 - NR6</td>
<td>100.0</td>
<td>108.0</td>
</tr>
<tr>
<td>117</td>
<td>NR5* - CR51 - NR5*</td>
<td>100.0</td>
<td>108.0</td>
</tr>
<tr>
<td>118</td>
<td>CB - CR51 - NR5</td>
<td>100.0</td>
<td>108.0</td>
</tr>
<tr>
<td>119</td>
<td>CB - CR51 - NR5*</td>
<td>100.0</td>
<td>108.0</td>
</tr>
<tr>
<td>120</td>
<td>CR61 - CR61 - CR61</td>
<td>100.0</td>
<td>120.0</td>
</tr>
<tr>
<td>121</td>
<td>CB - CR61 - CH2</td>
<td>100.0</td>
<td>120.0</td>
</tr>
<tr>
<td>122</td>
<td>CB - CR61 - CB</td>
<td>100.0</td>
<td>120.0</td>
</tr>
<tr>
<td>123</td>
<td>CB - CR61 - CB</td>
<td>100.0</td>
<td>120.0</td>
</tr>
<tr>
<td>124</td>
<td>NR6 - CR61 - CB</td>
<td>100.0</td>
<td>120.0</td>
</tr>
<tr>
<td>125</td>
<td>NR6 - CR61 - CR61</td>
<td>100.0</td>
<td>120.0</td>
</tr>
<tr>
<td>126</td>
<td>NR6* - CR61 - CR61</td>
<td>100.0</td>
<td>120.0</td>
</tr>
<tr>
<td>127</td>
<td>NR6* - CR61 - CB</td>
<td>100.0</td>
<td>120.0</td>
</tr>
<tr>
<td>128</td>
<td>CS2 - CR61 - CR61</td>
<td>100.0</td>
<td>120.0</td>
</tr>
<tr>
<td>129</td>
<td>CH2 - CB - NR5</td>
<td>100.0</td>
<td>126.0</td>
</tr>
<tr>
<td>130</td>
<td>CH2 - CB - NR5*</td>
<td>100.0</td>
<td>126.0</td>
</tr>
<tr>
<td>131</td>
<td>CR51 - CB - NR5</td>
<td>100.0</td>
<td>108.0</td>
</tr>
<tr>
<td>132</td>
<td>CR51 - CB - NR5*</td>
<td>100.0</td>
<td>108.0</td>
</tr>
<tr>
<td>133</td>
<td>CR61 - CB - CH2</td>
<td>100.0</td>
<td>126.0</td>
</tr>
<tr>
<td>134</td>
<td>CR61 - CB - OA</td>
<td>100.0</td>
<td>120.0</td>
</tr>
<tr>
<td>135</td>
<td>CR61 - CB - NT</td>
<td>100.0</td>
<td>132.0</td>
</tr>
<tr>
<td>136</td>
<td>CR61 - CB - NR5*</td>
<td>100.0</td>
<td>126.0</td>
</tr>
<tr>
<td>137</td>
<td>CR61 - CB - NP</td>
<td>100.0</td>
<td>120.0</td>
</tr>
<tr>
<td>138</td>
<td>CR61 - CB - C</td>
<td>100.0</td>
<td>120.0</td>
</tr>
<tr>
<td>139</td>
<td>CR61 - CB - CH2</td>
<td>100.0</td>
<td>120.0</td>
</tr>
<tr>
<td>140</td>
<td>CR61 - CB - CH3</td>
<td>100.0</td>
<td>120.0</td>
</tr>
<tr>
<td>141</td>
<td>CR61 - CB - CR61</td>
<td>100.0</td>
<td>120.0</td>
</tr>
<tr>
<td>142</td>
<td>CB - CB - O</td>
<td>100.0</td>
<td>120.0</td>
</tr>
<tr>
<td>143</td>
<td>CB - CB - NT</td>
<td>100.0</td>
<td>120.0</td>
</tr>
<tr>
<td>144</td>
<td>CB - CB - NR5</td>
<td>100.0</td>
<td>108.0</td>
</tr>
<tr>
<td>145</td>
<td>CB - CB - NR5*</td>
<td>100.0</td>
<td>108.0</td>
</tr>
<tr>
<td>146</td>
<td>CB - CB - NP</td>
<td>100.0</td>
<td>108.0</td>
</tr>
<tr>
<td>147</td>
<td>CB - CB - CH2</td>
<td>100.0</td>
<td>126.0</td>
</tr>
<tr>
<td>148</td>
<td>CB - CB - CH3</td>
<td>100.0</td>
<td>120.0</td>
</tr>
<tr>
<td>149</td>
<td>CB - CB - CR51</td>
<td>100.0</td>
<td>108.0</td>
</tr>
<tr>
<td>150</td>
<td>CB - CB - CR61</td>
<td>100.0</td>
<td>120.0</td>
</tr>
<tr>
<td>151</td>
<td>CB - CB - CB</td>
<td>100.0</td>
<td>120.0</td>
</tr>
<tr>
<td>152</td>
<td>NR6 - CB - O</td>
<td>100.0</td>
<td>120.0</td>
</tr>
<tr>
<td>153</td>
<td>NR6 - CB - NT</td>
<td>100.0</td>
<td>120.0</td>
</tr>
<tr>
<td>154</td>
<td>NR6 - CB - NR5*</td>
<td>100.0</td>
<td>132.0</td>
</tr>
<tr>
<td>155</td>
<td>NR6 - CB - CH2</td>
<td>100.0</td>
<td>120.0</td>
</tr>
<tr>
<td>156</td>
<td>NR6 - CB - CR61</td>
<td>100.0</td>
<td>120.0</td>
</tr>
<tr>
<td>157</td>
<td>NR6 - CB - CB</td>
<td>100.0</td>
<td>120.0</td>
</tr>
<tr>
<td>158</td>
<td>NR6 - CB - CR61</td>
<td>100.0</td>
<td>120.0</td>
</tr>
<tr>
<td>159</td>
<td>NR6* - CB - O</td>
<td>100.0</td>
<td>120.0</td>
</tr>
<tr>
<td>160</td>
<td>NR6* - CB - NT</td>
<td>100.0</td>
<td>120.0</td>
</tr>
<tr>
<td>161</td>
<td>NR6* - CB - CR61</td>
<td>100.0</td>
<td>120.0</td>
</tr>
<tr>
<td>162</td>
<td>NR6* - CB - CB</td>
<td>100.0</td>
<td>120.0</td>
</tr>
<tr>
<td>163</td>
<td>NR6* - CB - NR6</td>
<td>100.0</td>
<td>120.0</td>
</tr>
<tr>
<td>164</td>
<td>NR6* - CB - NR6*</td>
<td>100.0</td>
<td>120.0</td>
</tr>
<tr>
<td>165</td>
<td>CS2 - CB - C</td>
<td>100.0</td>
<td>120.0</td>
</tr>
<tr>
<td>166</td>
<td>CS2 - CB - CH2</td>
<td>100.0</td>
<td>120.0</td>
</tr>
<tr>
<td>167</td>
<td>CS2 - CB - CR61</td>
<td>100.0</td>
<td>120.0</td>
</tr>
<tr>
<td>168</td>
<td>CS2 - CB - NR6</td>
<td>100.0</td>
<td>120.0</td>
</tr>
<tr>
<td>I</td>
<td>ANGLE-TYPE</td>
<td>CT(I)</td>
<td>TS(I)</td>
</tr>
<tr>
<td>----</td>
<td>------------</td>
<td>-------</td>
<td>-------</td>
</tr>
<tr>
<td>169</td>
<td>CH3 - S - O</td>
<td>110.0</td>
<td>107.0</td>
</tr>
<tr>
<td>170</td>
<td>CH3 - S - CH2</td>
<td>110.0</td>
<td>100.0</td>
</tr>
<tr>
<td>171</td>
<td>CH3 - S - CH3</td>
<td>110.0</td>
<td>97.0</td>
</tr>
<tr>
<td>172</td>
<td>HS - S - CH2</td>
<td>95.0</td>
<td>96.0</td>
</tr>
<tr>
<td>173</td>
<td>S - S - CH2</td>
<td>110.0</td>
<td>104.0</td>
</tr>
<tr>
<td>174</td>
<td>NP - FE - NR5</td>
<td>10.0</td>
<td>115.0</td>
</tr>
<tr>
<td>175</td>
<td>NP - FE - NP</td>
<td>100.0</td>
<td>90.0</td>
</tr>
<tr>
<td>176</td>
<td>H - NZ - C</td>
<td>70.0</td>
<td>120.0</td>
</tr>
<tr>
<td>177</td>
<td>H - NZ - H</td>
<td>80.0</td>
<td>120.0</td>
</tr>
<tr>
<td>178</td>
<td>CH2 - NE - C</td>
<td>120.0</td>
<td>124.0</td>
</tr>
<tr>
<td>179</td>
<td>H - NE - C</td>
<td>70.0</td>
<td>120.0</td>
</tr>
<tr>
<td>180</td>
<td>H - NE - CH2</td>
<td>90.0</td>
<td>116.0</td>
</tr>
<tr>
<td>181</td>
<td>OM - P - OM</td>
<td>140.0</td>
<td>120.0</td>
</tr>
<tr>
<td>182</td>
<td>DA - P - DA</td>
<td>95.0</td>
<td>109.6</td>
</tr>
<tr>
<td>183</td>
<td>OS - P - OS</td>
<td>95.0</td>
<td>109.6</td>
</tr>
<tr>
<td>184</td>
<td>OS - P - DA</td>
<td>95.0</td>
<td>103.0</td>
</tr>
<tr>
<td>185</td>
<td>OS - P - OS</td>
<td>95.0</td>
<td>103.0</td>
</tr>
<tr>
<td>186</td>
<td>P - OS - CH2</td>
<td>95.0</td>
<td>120.0</td>
</tr>
<tr>
<td>187</td>
<td>P - OS - P</td>
<td>95.0</td>
<td>120.0</td>
</tr>
<tr>
<td>188</td>
<td>CS1 - OS - CS1</td>
<td>80.0</td>
<td>109.5</td>
</tr>
<tr>
<td>189</td>
<td>CS1 - OS - P</td>
<td>95.0</td>
<td>120.0</td>
</tr>
<tr>
<td>190</td>
<td>CS2 - OS - P</td>
<td>95.0</td>
<td>120.0</td>
</tr>
<tr>
<td>191</td>
<td>SI - OS - CS1</td>
<td>95.0</td>
<td>120.0</td>
</tr>
<tr>
<td>192</td>
<td>SI - OS - CS2</td>
<td>95.0</td>
<td>120.0</td>
</tr>
<tr>
<td>193</td>
<td>SI - OS - SI</td>
<td>95.0</td>
<td>155.0</td>
</tr>
<tr>
<td>194</td>
<td>OS - CS1 - NR5*</td>
<td>68.0</td>
<td>109.5</td>
</tr>
<tr>
<td>195</td>
<td>CS1 - CS1 - DA</td>
<td>68.0</td>
<td>109.5</td>
</tr>
<tr>
<td>196</td>
<td>CS1 - CS1 - NR5*</td>
<td>68.0</td>
<td>109.5</td>
</tr>
<tr>
<td>197</td>
<td>CS1 - CS1 - OS</td>
<td>68.0</td>
<td>109.5</td>
</tr>
<tr>
<td>198</td>
<td>CS1 - CS1 - CS1</td>
<td>60.0</td>
<td>109.5</td>
</tr>
<tr>
<td>199</td>
<td>NR6* - CS1 - CH2</td>
<td>60.0</td>
<td>109.5</td>
</tr>
<tr>
<td>200</td>
<td>NR6* - CS1 - CS1</td>
<td>60.0</td>
<td>109.5</td>
</tr>
<tr>
<td>201</td>
<td>NR6* - CS1 - OA</td>
<td>60.0</td>
<td>109.5</td>
</tr>
<tr>
<td>202</td>
<td>CS2 - CS1 - OA</td>
<td>68.0</td>
<td>109.5</td>
</tr>
<tr>
<td>203</td>
<td>CS2 - CS1 - NR5*</td>
<td>68.0</td>
<td>109.5</td>
</tr>
<tr>
<td>204</td>
<td>CS2 - CS1 - CH2</td>
<td>60.0</td>
<td>109.5</td>
</tr>
<tr>
<td>205</td>
<td>CS2 - CS1 - OS</td>
<td>68.0</td>
<td>109.5</td>
</tr>
<tr>
<td>206</td>
<td>CS2 - CS1 - CS1</td>
<td>60.0</td>
<td>109.5</td>
</tr>
<tr>
<td>207</td>
<td>CS2 - CS1 - NR6*</td>
<td>60.0</td>
<td>109.5</td>
</tr>
<tr>
<td>208</td>
<td>CB - NR6 - CR61</td>
<td>100.0</td>
<td>120.0</td>
</tr>
<tr>
<td>209</td>
<td>CB - NR6 - CB</td>
<td>100.0</td>
<td>120.0</td>
</tr>
<tr>
<td>210</td>
<td>CH3 - NR6* - CH2</td>
<td>100.0</td>
<td>120.0</td>
</tr>
<tr>
<td>211</td>
<td>CR61 - NR6* - CR61</td>
<td>100.0</td>
<td>120.0</td>
</tr>
<tr>
<td>212</td>
<td>CB - NR6* - CH2</td>
<td>100.0</td>
<td>120.0</td>
</tr>
<tr>
<td>213</td>
<td>CB - NR6* - CH3</td>
<td>100.0</td>
<td>120.0</td>
</tr>
<tr>
<td>214</td>
<td>CB - NR6* - CR61</td>
<td>100.0</td>
<td>120.0</td>
</tr>
<tr>
<td>215</td>
<td>CB - NR6* - CB</td>
<td>90.0</td>
<td>120.0</td>
</tr>
<tr>
<td>216</td>
<td>H - NR6* - CH2</td>
<td>90.0</td>
<td>120.0</td>
</tr>
<tr>
<td>217</td>
<td>H - NR6* - CB</td>
<td>100.0</td>
<td>120.0</td>
</tr>
<tr>
<td>218</td>
<td>CS1 - NR6* - CR61</td>
<td>100.0</td>
<td>120.0</td>
</tr>
<tr>
<td>219</td>
<td>CS1 - NR6* - CB</td>
<td>90.0</td>
<td>120.0</td>
</tr>
<tr>
<td>220</td>
<td>CS1 - NR6* - H</td>
<td>100.0</td>
<td>120.0</td>
</tr>
<tr>
<td>221</td>
<td>CS2 - NR6* - CB</td>
<td>90.0</td>
<td>120.0</td>
</tr>
<tr>
<td>222</td>
<td>CS2 - NR6* - H</td>
<td>100.0</td>
<td>120.0</td>
</tr>
<tr>
<td>223</td>
<td>CB - CS2 - CR61</td>
<td>68.0</td>
<td>109.5</td>
</tr>
<tr>
<td>224</td>
<td>CS1 - CS2 - OA</td>
<td>68.0</td>
<td>109.5</td>
</tr>
<tr>
<td>225</td>
<td>CS1 - CS2 - OS</td>
<td>68.0</td>
<td>109.5</td>
</tr>
<tr>
<td>I</td>
<td>ANGLE-TYPE</td>
<td>CT[I]</td>
<td>TD[I]</td>
</tr>
<tr>
<td>----</td>
<td>------------------</td>
<td>-------</td>
<td>-------</td>
</tr>
<tr>
<td>226</td>
<td>CS1 - CS2 - CS1</td>
<td>60.0</td>
<td>109.5</td>
</tr>
<tr>
<td>227</td>
<td>NR6* - CS2 - CB</td>
<td>100.0</td>
<td>120.0</td>
</tr>
<tr>
<td>228</td>
<td>NR6* - CS2 - CS1</td>
<td>100.0</td>
<td>120.0</td>
</tr>
<tr>
<td>229</td>
<td>CH1 - SI - CH1</td>
<td>95.0</td>
<td>109.5</td>
</tr>
<tr>
<td>230</td>
<td>OS - SI - CH1</td>
<td>95.0</td>
<td>109.5</td>
</tr>
<tr>
<td>231</td>
<td>OS - SI - OS</td>
<td>95.0</td>
<td>109.5</td>
</tr>
</tbody>
</table>

CT[I] in kcal mol⁻¹rad⁻², TD[I] in degrees, the sequential code is denoted by I.
2.5 Improper dihedral angle forces

The term in the interaction potential (2.1.1) that represents the harmonic, so-called improper (out-of-plane, out-of-tetrahedral configuration) dihedral bending interaction reads

\[
\sum_{n=1}^{N} \xi_n [\xi_n - \xi_0] \quad (2.5.1)
\]

The summation runs over a set of dihedral angles \( n = 1 \ldots N_\xi \), which are selected to keep groups of atoms near a specified spatial configuration. For example, in an amino acid residue the atoms \( \text{CA}, \text{C}, \text{O} \) and \( \text{N} \) are kept near a planar configuration by defining an improper dihedral \( \text{C-CA-N-O} \) with \( \xi_0 = 0^\circ \) (cyclopropane). If the \( \text{CA} \)-atom in an amino acid carries no explicit hydrogen, viz. it is a united atom of type CH\(^2\), the atoms \( \text{CA}, \text{N}, \text{C} \) and \( \text{CB} \) are kept near a tetrahedral configuration by defining an improper dihedral \( \text{CA-N-C-CB} \) (L-amino acid) or \( \text{CA-C-N-CB} \) (D-amino acid) with \( \xi_0 = 35.26^\circ \). A third example is a planar ring configuration, like in a phenylalanine amino acid residue, which is kept close to planarity by defining 6 improper dihedrals, viz. \( \text{CG-CD\(^1\)-CE\(^1\)-CZ}, \text{CD\(^1\)-CE\(^1\)-CZ-CE\(^2\)}, \text{CE\(^1\)-CZ-CE\(^2\)-CD\(^2\)}, \text{CZ-CE\(^2\)-CD\(^2\)-CG}, \text{CE\(^2\)-CD\(^2\)-CG-CD\(^1\)}, \text{CD\(^2\)-CG-CD\(^1\)-CE\(^1\)}\), all with \( \xi_0 = 0^\circ \). The improper dihedral definitions can be found in the residue topology building block files RT***.DAT. The information is printed in more readable form in the output of program PROGMT, see example files OUTGTM***.LIS. The actual improper dihedral angle of the improper dihedral \( n \) defined by atoms with sequence numbers \( i, j, k \) and \( l \) (\( i-j-k-l \)) is denoted by \( \xi_n \). It is given by
\[ \xi_n = \text{sign}(\xi_n) \arccos \left[ \frac{\hat{r}_{ij} \times \hat{r}_{kj} \cdot (\hat{r}_{kj} \times \hat{r}_{kl})}{|\hat{r}_{ij} \times \hat{r}_{kj}| \cdot |\hat{r}_{kj} \times \hat{r}_{kl}|} \right] \]  

(2.5.2)

where

\[ \text{sign}(\xi_n) = \text{sign}[\hat{r}_{kj} \cdot (\hat{r}_{ij} \times \hat{r}_{kj}) \times (\hat{r}_{kj} \times \hat{r}_{kl})] \]  

(2.5.3)

following the IUPAC-IUB convention (Biochemistry 9 (1970) 3471). The forces on atoms \( i, j, k \) and \( l \) due to the \( n \)-th term in formula (2.5.1) are

\[ \hat{F}_i = -\frac{dV}{d\xi_n} \frac{d\xi_n}{d\cos\xi_n} \frac{\partial \cos\xi_n}{\partial \hat{F}_i} \]  

(2.5.4)

\[ \hat{F}_j = -\frac{dV}{d\xi_n} \frac{d\xi_n}{d\cos\xi_n} \frac{\partial \cos\xi_n}{\partial \hat{F}_j} \]  

(2.5.5)

\[ \hat{F}_k = -\frac{dV}{d\xi_n} \frac{d\xi_n}{d\cos\xi_n} \frac{\partial \cos\xi_n}{\partial \hat{F}_k} \]  

(2.5.6)

\[ \hat{F}_l = -\frac{dV}{d\xi_n} \frac{d\xi_n}{d\cos\xi_n} \frac{\partial \cos\xi_n}{\partial \hat{F}_l} \]  

(2.5.7)

with

\[ \frac{dV}{d\xi_n} \frac{d\xi_n}{d\cos\xi_n} = -K_n \left[ \xi_n - \xi_0 \right]/\sin\xi_n \]  

(2.5.8)

and
\[
\frac{\partial \cos \xi_n}{\partial x_{ij}} = \frac{\hat{r}_{ij} - \cos \xi_n}{\sqrt{\hat{r}_{ij}^2 - \cos \xi_n^2}}
\]

(2.5.9)

\[
\frac{\partial \cos \xi_n}{\partial r_{kj}} = \frac{\hat{r}_{kj} - \cos \xi_n}{\sqrt{\hat{r}_{kj}^2 - \cos \xi_n^2}}
\]

(2.5.10)

\[
\frac{\partial \cos \xi_n}{\partial \theta_k} = \frac{\hat{r}_{kj} - \cos \xi_n}{\sqrt{\hat{r}_{kj}^2 - \cos \xi_n^2}}
\]

(2.5.11)

\[
\frac{\partial \cos \xi_n}{\partial \alpha_l} = \frac{\hat{r}_{kj} - \cos \xi_n}{\sqrt{\hat{r}_{kj}^2 - \cos \xi_n^2}}
\]

(2.5.12)

Here we have used the relation \( \xi = \text{sign}(\xi) \cdot \arccos(\cos \xi) \), from which it follows that

\[
\frac{d\xi_n}{d\cos \xi} = \text{sign}(\xi)(1-\cos^2 \xi)^{-1/2} = [-\sin \xi]^{-1/2}
\]

(2.5.13)

The interaction energy and forces are calculated in subroutine DIMANG with switch IDI=2. For reasons of ease of analysis, the list of \( N \) improper dihedral angles is split into two lists, one of improper dihedrals involving
hydrogen atoms, and one involving the other improper dihedrals. These lists are kept in the molecular topology file (Chapter 3.3.2). The first list contains NQHIH improper dihedral angles involving hydrogen atoms. Five items are stored: IQH, JQH, KQH, LQH[1 .. NQHIH] are the atom sequence numbers of the atoms forming improper dihedral i-j-k-l as a function of the improper dihedral sequence number n, and ICQH[1 .. NQHIH] is the improper dihedral type code, denoting the parameters $K_{\xi_n}$ and $\xi_0$, as a function of the improper dihedral sequence number n. The force field parameters $K_{\xi}$ and $\xi_0$ for the various types of improper dihedrals are stored in CQ[1 .. NQTY] and Q0[1 .. NQTY] as a function of the improper dihedral type code (ICQH or ICQ). They can be found in the interaction parameter files IFP***.DAT, and are printed in more readable form in the output of program PROGMOT, see example files OUTGMT***.LIS. For the GROMOS force fields 37C and 37D, they are listed in Table 2.5.1.
<table>
<thead>
<tr>
<th>IMP. DIM. -TYPE</th>
<th>CG [1]</th>
<th>GO [1]</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 NR5* - X - Y - NR5</td>
<td>40.0</td>
<td>0.0</td>
</tr>
<tr>
<td>2 NR5* - X - Y - NR5*</td>
<td>40.0</td>
<td>0.0</td>
</tr>
<tr>
<td>3 C - X - Y - D</td>
<td>40.0</td>
<td>0.0</td>
</tr>
<tr>
<td>4 C - X - Y - OM</td>
<td>40.0</td>
<td>0.0</td>
</tr>
<tr>
<td>5 C - X - Y - NT</td>
<td>40.0</td>
<td>0.0</td>
</tr>
<tr>
<td>6 CH1 - X - Y - DA</td>
<td>80.0</td>
<td>35.26439</td>
</tr>
<tr>
<td>7 CH1 - X - Y - CH1</td>
<td>80.0</td>
<td>35.26439</td>
</tr>
<tr>
<td>8 CH2 - X - Y - N</td>
<td>40.0</td>
<td>0.0</td>
</tr>
<tr>
<td>9 CH2 - X - Y - NT</td>
<td>40.0</td>
<td>0.0</td>
</tr>
<tr>
<td>10 CH2 - X - Y - C</td>
<td>40.0</td>
<td>0.0</td>
</tr>
<tr>
<td>11 CH2 - X - Y - CH1</td>
<td>80.0</td>
<td>35.26439</td>
</tr>
<tr>
<td>12 CH3 - X - Y - CH1</td>
<td>80.0</td>
<td>35.26439</td>
</tr>
<tr>
<td>13 CR51 - X - Y - NR5</td>
<td>40.0</td>
<td>0.0</td>
</tr>
<tr>
<td>14 CR51 - X - Y - NR5*</td>
<td>40.0</td>
<td>0.0</td>
</tr>
<tr>
<td>15 CR51 - X - Y - CR51</td>
<td>40.0</td>
<td>0.0</td>
</tr>
<tr>
<td>16 CR61 - X - Y - CR61</td>
<td>40.0</td>
<td>0.0</td>
</tr>
<tr>
<td>17 CB - X - Y - D</td>
<td>40.0</td>
<td>0.0</td>
</tr>
<tr>
<td>18 CB - X - Y - DA</td>
<td>40.0</td>
<td>0.0</td>
</tr>
<tr>
<td>19 CB - X - Y - NT</td>
<td>40.0</td>
<td>0.0</td>
</tr>
<tr>
<td>20 CB - X - Y - NR5</td>
<td>40.0</td>
<td>0.0</td>
</tr>
<tr>
<td>21 CB - X - Y - NR5*</td>
<td>40.0</td>
<td>0.0</td>
</tr>
<tr>
<td>22 CB - X - Y - NP</td>
<td>40.0</td>
<td>0.0</td>
</tr>
<tr>
<td>23 CB - X - Y - C</td>
<td>40.0</td>
<td>0.0</td>
</tr>
<tr>
<td>24 CB - X - Y - CH2</td>
<td>40.0</td>
<td>0.0</td>
</tr>
<tr>
<td>25 CB - X - Y - CH3</td>
<td>40.0</td>
<td>0.0</td>
</tr>
<tr>
<td>26 CB - X - Y - CR51</td>
<td>40.0</td>
<td>0.0</td>
</tr>
<tr>
<td>27 CB - X - Y - CR61</td>
<td>40.0</td>
<td>0.0</td>
</tr>
<tr>
<td>28 CB - X - Y - CB</td>
<td>40.0</td>
<td>0.0</td>
</tr>
<tr>
<td>29 H - X - Y - N</td>
<td>40.0</td>
<td>0.0</td>
</tr>
<tr>
<td>30 H - X - Y - NT</td>
<td>80.0</td>
<td>35.26439</td>
</tr>
<tr>
<td>31 H - X - Y - NL</td>
<td>40.0</td>
<td>0.0</td>
</tr>
<tr>
<td>32 H - X - Y - NR5*</td>
<td>40.0</td>
<td>0.0</td>
</tr>
<tr>
<td>33 FE - X - Y - NP</td>
<td>40.0</td>
<td>0.0</td>
</tr>
<tr>
<td>34 NZ - X - Y - C</td>
<td>40.0</td>
<td>0.0</td>
</tr>
<tr>
<td>35 NE - X - Y - C</td>
<td>40.0</td>
<td>0.0</td>
</tr>
<tr>
<td>36 NE - X - Y - H</td>
<td>40.0</td>
<td>0.0</td>
</tr>
<tr>
<td>37 CS1 - X - Y - NR5*</td>
<td>40.0</td>
<td>0.0</td>
</tr>
<tr>
<td>38 CS1 - X - Y - CH2</td>
<td>80.0</td>
<td>35.26439</td>
</tr>
<tr>
<td>39 CS1 - X - Y - CB</td>
<td>40.0</td>
<td>0.0</td>
</tr>
<tr>
<td>40 CS1 - X - Y - CS1</td>
<td>80.0</td>
<td>35.26439</td>
</tr>
<tr>
<td>41 NR6 - X - Y - CR61</td>
<td>40.0</td>
<td>0.0</td>
</tr>
<tr>
<td>42 NR6 - X - Y - CB</td>
<td>40.0</td>
<td>0.0</td>
</tr>
<tr>
<td>43 NR6 - X - Y - NR6</td>
<td>40.0</td>
<td>0.0</td>
</tr>
<tr>
<td>44 NR6* - X - Y - CH2</td>
<td>40.0</td>
<td>0.0</td>
</tr>
<tr>
<td>45 NR6* - X - Y - CH3</td>
<td>40.0</td>
<td>0.0</td>
</tr>
<tr>
<td>I</td>
<td>IMP.DIR.-TYPE</td>
<td>CQ[I]</td>
</tr>
<tr>
<td>---</td>
<td>-------------</td>
<td>-------</td>
</tr>
<tr>
<td>46</td>
<td>NR6* - X - Y - CR61</td>
<td>40.0</td>
</tr>
<tr>
<td>47</td>
<td>NR6* - X - Y - CB</td>
<td>40.0</td>
</tr>
<tr>
<td>48</td>
<td>NR6* - X - Y - H</td>
<td>40.0</td>
</tr>
<tr>
<td>49</td>
<td>NR6* - X - Y - CSI</td>
<td>40.0</td>
</tr>
<tr>
<td>50</td>
<td>NR6* - X - Y - NR6*</td>
<td>40.0</td>
</tr>
<tr>
<td>51</td>
<td>NR6* - X - Y - NR6*</td>
<td>40.0</td>
</tr>
<tr>
<td>52</td>
<td>CS2 - X - Y - CB</td>
<td>40.0</td>
</tr>
<tr>
<td>53</td>
<td>CS2 - X - Y - CSI</td>
<td>35.0</td>
</tr>
<tr>
<td>54</td>
<td>CS2 - X - Y - NR6*</td>
<td>40.0</td>
</tr>
<tr>
<td>55</td>
<td>SI - X - Y - CH1</td>
<td>40.0</td>
</tr>
</tbody>
</table>

CQ[I] in kcal mol\(^{-1}\) rad\(^{-2}\), PQ[I] in degrees, the sequential code is denoted by I.
2.6 Dihedral angle torsion forces

The term in the interaction potential (2.1.1) that represents the sinusoidal dihedral torsion interaction reads

$$\sum_{n,n'}^N \sum_{n,n'}^N \phi_n \phi_{n'} \left( V_{\phi} (\phi_n, \phi_{n'}) = k \left[ 1 + \cos(n \phi_n, \phi_{n'}, - \delta_n, - \delta_{n'}) \right] \right)$$

(2.6.1)

The summation runs over a set of dihedral angles $n' = 1 \ldots N_\phi$, which are selected in the following way.

1. For each bond between atoms with sequence number $j$ and $k$ only one quadruple of atoms $i-j-k-l$ is chosen to form dihedral $\phi_{n'}$.
2. Bonds $j-k$ that are part of rigid planar ring structures (e.g., in Phe, Trp, etc.) are not included but treated as improper dihedrals (Chapter 2.5).
3. Torsion angles around bonds $j-k$ where $j$ and $k$ denote carbon atoms in sugar rings are treated in a special manner:
   - one quadruple $i-j-k-l$ is assigned to a dihedral with multiplicity three: $K_{\phi_{n'}} = 1.4$ kcal/mol, $\delta_{n'} = 0.0$ and $n_{n'} = 3$.
   - for each dihedral $i-j-k-l$, where $i$ denotes an oxygen atom and $l$ denotes any atom except an oxygen, a dihedral with multiplicity two is additionally defined: $K_{\phi_{n'}} = 0.1$ kcal/mol, $\delta_{n'} = 0.0$ and $n_{n'} = 2$.
   - for each dihedral $i-j-k-l$, where $i$ and $l$ denote oxygen atoms, a dihedral with multiplicity two is additionally defined: $K_{\phi_{n'}} = 0.5$ kcal/mol, $\delta_{n'} = 0.0$ and $n_{n'} = 2$.
4. Torsion angles around bonds $j-k$, where $j$ denotes a phosphor atom and $k$
\[ \frac{dV}{d\phi_n} = K_{\phi_n} \sin(n_1 \phi_n - n_2)/\sin n_1, \quad \frac{d\phi_n}{d\cos \phi_n} = K_{\phi_n} \sin(n_1 \phi_n - n_2)/\sin n_1. \] 

The interaction energy and forces are calculated in subroutine DIHANG with switch IDI=1. For reasons of ease of analysis, the list of \( N_\phi \) torsional dihedral angles is split into two lists, one of dihedrals involving hydrogen atoms, and one involving the other dihedrals. These lists are kept in the molecular topology file (Chapter 3.3.2). The first list contains NPHIH dihedral angles involving hydrogen atoms. Five items are stored: IPH, JPH, KPH, LPH[1 .. NPHIH] are the atom sequence numbers of the atoms forming dihedral i-j-k-l as a function of the dihedral sequence.
number \( n' \), and ICPH[1 .. NPHI] is the dihedral type code, denoting the parameters \( K_{n'}, n_{n'}, \) and \( \delta_{n'} \), as a function of the dihedral sequence number \( n' \). The force field parameters \( K_{n'}, n_{n'}, \) and \( \delta_{n'} \), for the various types of torsional dihedrals are stored in CP[1 .. NPTY], NP[1 .. NPTY] and PD[1 .. NPTY] as a function of the torsional dihedral type code (ICPH or ICP). They can be found in the interaction parameter files IPF***.DAT, and are printed in more readable form in the output of program PROOMT, see example files OUTFMT***.LIS. For the GRÖMOS force fields 37C and 37D, they are listed in Table 2.6.1. The list for the dihedrals involving no hydrogen atoms contains corresponding items denoted by IP, JP, KP, LP ICP[1 .. NPHI].
<table>
<thead>
<tr>
<th>I</th>
<th>DIHEDRAL TYPE</th>
<th>CP[I]</th>
<th>PD[I]</th>
<th>NP[I]</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>X - C - DA - Y</td>
<td>4.00</td>
<td>180.0</td>
<td>2</td>
</tr>
<tr>
<td>2</td>
<td>X - C - N - Y</td>
<td>8.00</td>
<td>180.0</td>
<td>2</td>
</tr>
<tr>
<td>3</td>
<td>X - C - NT - Y</td>
<td>8.00</td>
<td>180.0</td>
<td>2</td>
</tr>
<tr>
<td>4</td>
<td>X - C - NL - Y</td>
<td>0.90</td>
<td>0.0</td>
<td>2</td>
</tr>
<tr>
<td>5</td>
<td>X - CH1 - DA - Y</td>
<td>0.30</td>
<td>0.0</td>
<td>3</td>
</tr>
<tr>
<td>6</td>
<td>x - CH1 - N - Y</td>
<td>0.24</td>
<td>0.10</td>
<td>180.0</td>
</tr>
<tr>
<td>7</td>
<td>CH1 - NT - Y</td>
<td>0.90</td>
<td>0.0</td>
<td>6</td>
</tr>
<tr>
<td>8</td>
<td>CH1 - NL - Y</td>
<td>0.90</td>
<td>0.0</td>
<td>6</td>
</tr>
<tr>
<td>9</td>
<td>CH1 - C - Y</td>
<td>0.34</td>
<td>0.10</td>
<td>0.0</td>
</tr>
<tr>
<td>10</td>
<td>CH1 - CH1 - Y</td>
<td>1.40</td>
<td>0.0</td>
<td>3</td>
</tr>
<tr>
<td>11</td>
<td>CH2 - DA - Y</td>
<td>0.30</td>
<td>0.0</td>
<td>3</td>
</tr>
<tr>
<td>12</td>
<td>CH2 - N - Y</td>
<td>0.24</td>
<td>0.10</td>
<td>180.0</td>
</tr>
<tr>
<td>13</td>
<td>CH2 - NT - Y</td>
<td>0.90</td>
<td>0.0</td>
<td>6</td>
</tr>
<tr>
<td>14</td>
<td>CH2 - NL - Y</td>
<td>0.90</td>
<td>0.0</td>
<td>6</td>
</tr>
<tr>
<td>15</td>
<td>CH2 - C - Y</td>
<td>0.24</td>
<td>0.10</td>
<td>0.0</td>
</tr>
<tr>
<td>16</td>
<td>CH2 - CH1 - Y</td>
<td>1.40</td>
<td>0.0</td>
<td>3</td>
</tr>
<tr>
<td>17</td>
<td>CH2 - CH2 - Y</td>
<td>1.40</td>
<td>0.0</td>
<td>3</td>
</tr>
<tr>
<td>18</td>
<td>CB - DA - Y</td>
<td>1.70</td>
<td>180.0</td>
<td>2</td>
</tr>
<tr>
<td>19</td>
<td>CB - NT - Y</td>
<td>8.00</td>
<td>180.0</td>
<td>2</td>
</tr>
<tr>
<td>20</td>
<td>CB - NL - Y</td>
<td>0.34</td>
<td>0.10</td>
<td>0.0</td>
</tr>
<tr>
<td>21</td>
<td>CB - C - Y</td>
<td>1.40</td>
<td>180.0</td>
<td>2</td>
</tr>
<tr>
<td>22</td>
<td>CB - CB61 - Y</td>
<td>10.00</td>
<td>180.0</td>
<td>2</td>
</tr>
<tr>
<td>23</td>
<td>S - CH2 - Y</td>
<td>0.70</td>
<td>0.0</td>
<td>3</td>
</tr>
<tr>
<td>24</td>
<td>S - S - Y</td>
<td>4.00</td>
<td>0.0</td>
<td>4</td>
</tr>
<tr>
<td>25</td>
<td>FE - NR5 - Y</td>
<td>0.00</td>
<td>0.0</td>
<td>0</td>
</tr>
<tr>
<td>26</td>
<td>N2 - C - Y</td>
<td>8.00</td>
<td>180.0</td>
<td>2</td>
</tr>
<tr>
<td>27</td>
<td>NE - C - Y</td>
<td>8.00</td>
<td>180.0</td>
<td>2</td>
</tr>
<tr>
<td>28</td>
<td>NE - CH2 - Y</td>
<td>0.24</td>
<td>0.10</td>
<td>180.0</td>
</tr>
<tr>
<td>29</td>
<td>P - DA - Y</td>
<td>0.25</td>
<td>0.0</td>
<td>3</td>
</tr>
<tr>
<td>30</td>
<td>P - DA - Y</td>
<td>0.75</td>
<td>0.0</td>
<td>3</td>
</tr>
<tr>
<td>31</td>
<td>OS - CH2 - Y</td>
<td>0.90</td>
<td>0.0</td>
<td>3</td>
</tr>
<tr>
<td>32</td>
<td>OS - P - Y</td>
<td>0.25</td>
<td>0.0</td>
<td>3</td>
</tr>
<tr>
<td>33</td>
<td>OS - P - Y</td>
<td>0.75</td>
<td>0.0</td>
<td>3</td>
</tr>
<tr>
<td>34</td>
<td>CS1 - DA - Y</td>
<td>0.30</td>
<td>0.0</td>
<td>3</td>
</tr>
<tr>
<td>35</td>
<td>CS1 - NR5 - Y</td>
<td>0.00</td>
<td>0.0</td>
<td>3</td>
</tr>
<tr>
<td>36</td>
<td>CS1 - CH2 - Y</td>
<td>1.40</td>
<td>0.0</td>
<td>3</td>
</tr>
<tr>
<td>37</td>
<td>CS1 - OS - Y</td>
<td>0.90</td>
<td>0.0</td>
<td>3</td>
</tr>
<tr>
<td>38</td>
<td>CS1 - CS1 - Y</td>
<td>1.40</td>
<td>0.0</td>
<td>3</td>
</tr>
<tr>
<td>39</td>
<td>CS1 - CS1 - Y</td>
<td>0.10</td>
<td>0.0</td>
<td>3</td>
</tr>
<tr>
<td>40</td>
<td>CS1 - CS1 - Y</td>
<td>0.50</td>
<td>0.0</td>
<td>3</td>
</tr>
<tr>
<td>41</td>
<td>NR6 - CB - Y</td>
<td>8.00</td>
<td>180.0</td>
<td>2</td>
</tr>
<tr>
<td>42</td>
<td>NR6 - CH2 - Y</td>
<td>0.24</td>
<td>0.10</td>
<td>180.0</td>
</tr>
<tr>
<td>43</td>
<td>NR6 - CB - Y</td>
<td>8.00</td>
<td>180.0</td>
<td>2</td>
</tr>
<tr>
<td>44</td>
<td>NR6 - CS1 - Y</td>
<td>0.00</td>
<td>0.0</td>
<td>2</td>
</tr>
<tr>
<td>45</td>
<td>CS2 - DA - Y</td>
<td>0.30</td>
<td>0.0</td>
<td>3</td>
</tr>
<tr>
<td>46</td>
<td>CS2 - OS - Y</td>
<td>0.90</td>
<td>0.0</td>
<td>3</td>
</tr>
<tr>
<td>47</td>
<td>CS2 - CS1 - Y</td>
<td>1.40</td>
<td>0.0</td>
<td>3</td>
</tr>
<tr>
<td>48</td>
<td>CS2 - CS1 - Y</td>
<td>0.10</td>
<td>0.0</td>
<td>3</td>
</tr>
<tr>
<td>49</td>
<td>CS2 - CS1 - Y</td>
<td>0.50</td>
<td>0.0</td>
<td>2</td>
</tr>
<tr>
<td>50</td>
<td>SI - CH1 - Y</td>
<td>1.40</td>
<td>0.0</td>
<td>3</td>
</tr>
<tr>
<td>51</td>
<td>SI - OS - Y</td>
<td>0.90</td>
<td>0.0</td>
<td>3</td>
</tr>
</tbody>
</table>

CP[I] in kcal mol\(^{-1}\), PD[I] in degrees, the sequential code is denoted by I.
2.7 Non-bonded forces

The term in the interaction potential (2.1.1) that represents the non-bonded interaction, reads

\[ \sum_{i<j}^{N^a} \left[ C_{12}^{a}(i,j)/r_{ij}^{12} - C_6^{a}(i,j)/r_{ij}^6 + q_i q_j/(4\pi\varepsilon_0\varepsilon_n r_{ij}) \right] S(r_{ij}) \]  (2.7.1)

The summation runs in principle over all pairs of atoms with atom sequence number \( i \) and \( j \). However, generally a number of pairs is excluded from the summation. This is discussed in Chapter 2.7.1 (excluded neighbours), Chapter 2.7.5 (charge groups) and Chapter 2.7.6 (searching neighbours). The non-bonded interaction parameters are discussed in Chapter 2.7.2-4. Chapter 2.7.7 contains a scheme for treating long-range (Coulomb) interactions and Chapter 2.7.8 discusses the use of switching functions.

The forces on atoms \( i \) and \( j \) due to the \((i,j)\)-th term of formula (2.7.1) are

\[ \vec{F}_i = -\left[ -12C_{12}^{a}(i,j)/r_{ij}^{12} + 6C_6^{a}(i,j)/r_{ij}^6 - q_i q_j/(4\pi\varepsilon_0\varepsilon_n r_{ij}) \right] \left[ \vec{r}_{ij}/r_{ij}^3 \right] S(r_{ij}) \]

and

\[ \vec{F}_j = -\vec{F}_i \]  (2.7.2)
where the switching function $S(r_{ij})$ and its derivative are defined by
formulae (2.7.8.2-3).

The non-bonded interaction energy and forces are calculated in a number
of different subroutines:

- NBPAL and NONBAL, which make use of the concept of an atom pair list.
- NBPL and NONBML, which make use of the concept of a charge group or
molecular pair list. For these two routines vectorized versions are
- NONBB, which makes use of the concept of a spatial grid to search
neighbours.
- NONBP, which finds atom pairs by scanning all possible pairs of charge
groups in the system.

2.7.1 Excluded neighbours

Atoms $i$ and $j$ that are covalently bound, are called first neighbour
atoms and if they are each covalently bound to one common neighbour atom,
they are called second neighbours. Due to the short distance $r_{ij}$ between
first or second neighbours, the non-bonded interaction (2.7.1) between such
neighbour atoms $i$ and $j$ will be very large. Therefore first and second
neighbours are excluded from the summation in formula (2.7.1).

Lists of excluded atoms are kept in the molecular topology file
(Chapter 3.3.2). For the NRP atoms in the "solute" part of the molecular
topology INE[1 .. NRP] denotes how many excluded neighbours $j$ belong to each
atom $i$ = '1 .. NRP'. KNE[1 .. NRP] denotes where the atom sequence numbers $j$
of the excluded neighbours of a "solute" atom can be found in array JSNE[1 ..
]. The sequence numbers $j$ of atoms that are excluded from the non-bonded
interaction with the atom with sequence number \( i \), are positioned at positions \( \text{KNE}[i]+1, ..., \text{KNE}[i]+\text{INE}[i] \) in JSNE. It is assumed that \( \langle i, j \rangle \). In the solvent part of the molecular topology all NRAM atoms that form a solvent molecule are excluded neighbours of each other.

For the GROMOS force fields the excluded atom information can be found in the residue topology building block files RT***.DAT, and is printed in more readable form in the output of program PROGTM, see example files OUTGTM***.LIS.

2.7.2 Normal Lennard Jones interaction

The non-bonded interaction van der Waals' parameters \( C_{12}(i, j) \) and \( C_{6}(i, j) \) in formula (2.7.1) depend on the atom type or more specifically the integer atom codes \( I = \text{IAC}[i] \) and \( J = \text{IAC}[j] \) of the atoms with atom sequence numbers \( i \) and \( j \). Lists of integer atom codes are kept in the molecular topology file (Chapter 3.3.2). For the NRP atoms of the "solute" part of the molecular topology the integer atom codes are stored in IAC[1..NRP]. The integer atom codes of the NRAM solvent atoms are stored in IACS[1..NRAM]. The van der Waals' parameters are kept in the molecular topology file (Chapter 3.3.2). For the NRATT atom types, \( C_{12}[1..NRATT\cdot(NRATT+1)]/2 \) contains the coefficient \( C_{12} \) in (2.7.1) as a function of the occurring pair codes; the sequence of atom pairs with integer atom codes ranging from 1 to NRATT is:

\[ 1-1, 1-2, 2-2, ..., 1-NRATT, 2-NRATT, ..., NRATT-NRATT. \]

The coefficients \( C_{6} \) in (2.7.1) are kept likewise in \( C_{6}(1..NRATT\cdot(NRATT+1))/2 \). In this way it is possible to change the van der Waals' interaction between each pair of atom types independently.

Originally, the GROMOS van der Waals parameters for an atom pair with
integer atom codes $I$ and $J$ were derived from single atom van der Waals parameters using the relations

$$C_6^{1/2}(I,J) = C_6^{1/2}(I) \cdot C_6^{1/2}(J)$$  \hspace{1cm} (2.7.2.1)$$

and

$$C_{12}^{1/2}(I,J) = C_{12}^{1/2}(I) \cdot C_{12}^{1/2}(J)$$  \hspace{1cm} (2.7.2.2)$$

For the GROMOS force fields 37C and 37D, the single atom van der Waals parameters $C_6^{1/2}(I)$ and $C_{12}^{1/2}(I)$ are given in the third and fourth column of Table 2.7.2.1 as a function of integer atom code or atom type.

GROMOS force fields do not contain a special term in the interaction function (2.1.1) that mimicks hydrogen bonding. The hydrogen bonding capacity of molecules is the result of a balance between Coulomb and van der Waals' attraction and repulsion. In order to mimic correctly the hydrogen bonding properties of polar atoms, their van der Waals' repulsion had to be increased over the value resulting from the use of $C_{12}^{1/2}$ from the fourth column in Tables 2.7.2.1-2 [MD 84.1]. The fifth column in these tables contains the $C_{12}^{1/2}(I)$ values to be used between polar atoms. For correct modelling of hydrogen bonds between atoms that are part of a charged moiety, like the OM atom in a COO⁻ group and a NL atom in an NH₃⁺ group, the repulsive part of the van der Waals' interaction had to be increased even more. The sixth column in Tables 2.7.2.1-2 contains the $C_{12}^{1/2}(I)$ values to be used between oppositely charged atoms. In Table 2.7.2.3 it is denoted which values for $C_{12}^{1/2}(I)$ and $C_{12}^{1/2}(J)$ are to be used in formula (2.7.2.2) for obtaining $C_{12}(I,J)$. 
For the GROMOS force fields, the normal van der Waals' interaction parameters can be found in the interaction function parameter file IPP***.DAT, and are printed in more readable form in the output of program PROGTX behavior file OUTGTX***.LIS.
### Table 2.7.2.1

GROMOS 37C4 normal van der Waals' parameters

<table>
<thead>
<tr>
<th>IAC Type</th>
<th>$C_6^{1/2}$[kcal mol$^{-1}$ Å$^{-1.5}$]$^{1/2}$</th>
<th>$C_{12-}^{1/2}$[kcal mol$^{-1}$ Å$^{-12}$]$^{1/2}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>O</td>
<td>23.25</td>
</tr>
<tr>
<td>2</td>
<td>OM</td>
<td>23.25</td>
</tr>
<tr>
<td>3</td>
<td>OA</td>
<td>23.25</td>
</tr>
<tr>
<td>4</td>
<td>CW</td>
<td>25.01</td>
</tr>
<tr>
<td>5</td>
<td>N</td>
<td>24.13</td>
</tr>
<tr>
<td>6</td>
<td>NT</td>
<td>24.13</td>
</tr>
<tr>
<td>7</td>
<td>NL</td>
<td>24.13</td>
</tr>
<tr>
<td>8</td>
<td>NR5</td>
<td>24.13</td>
</tr>
<tr>
<td>9</td>
<td>NR5*</td>
<td>24.13</td>
</tr>
<tr>
<td>10</td>
<td>NP</td>
<td>24.13</td>
</tr>
<tr>
<td>11</td>
<td>C</td>
<td>23.65</td>
</tr>
<tr>
<td>12</td>
<td>CH1</td>
<td>54.65</td>
</tr>
<tr>
<td>13</td>
<td>CH2</td>
<td>46.53</td>
</tr>
<tr>
<td>14</td>
<td>CH3</td>
<td>46.06</td>
</tr>
<tr>
<td>15</td>
<td>CH51</td>
<td>36.30</td>
</tr>
<tr>
<td>16</td>
<td>CH61</td>
<td>36.30</td>
</tr>
<tr>
<td>17</td>
<td>CB</td>
<td>23.65</td>
</tr>
<tr>
<td>18</td>
<td>H</td>
<td>0.0</td>
</tr>
<tr>
<td>19</td>
<td>HO</td>
<td>0.0</td>
</tr>
<tr>
<td>20</td>
<td>HW</td>
<td>0.0</td>
</tr>
<tr>
<td>21</td>
<td>HS</td>
<td>0.0</td>
</tr>
<tr>
<td>22</td>
<td>S</td>
<td>48.85</td>
</tr>
<tr>
<td>23</td>
<td>FE</td>
<td>0.0</td>
</tr>
<tr>
<td></td>
<td>Element</td>
<td></td>
</tr>
<tr>
<td>---</td>
<td>---------</td>
<td>---</td>
</tr>
<tr>
<td>24</td>
<td>ZN</td>
<td>0.0</td>
</tr>
<tr>
<td>25</td>
<td>NZ</td>
<td>24.13</td>
</tr>
<tr>
<td>26</td>
<td>NE</td>
<td>24.13</td>
</tr>
<tr>
<td>27</td>
<td>P</td>
<td>59.35</td>
</tr>
<tr>
<td>28</td>
<td>OS</td>
<td>23.25</td>
</tr>
<tr>
<td>29</td>
<td>CS1</td>
<td>54.65</td>
</tr>
<tr>
<td>30</td>
<td>NR6</td>
<td>24.13</td>
</tr>
<tr>
<td>31</td>
<td>NR6*</td>
<td>24.13</td>
</tr>
<tr>
<td>32</td>
<td>CS2</td>
<td>46.63</td>
</tr>
<tr>
<td>33</td>
<td>SI</td>
<td>59.35</td>
</tr>
<tr>
<td>34</td>
<td>NA</td>
<td>0.914.15</td>
</tr>
<tr>
<td>35</td>
<td>CL</td>
<td>57.44</td>
</tr>
<tr>
<td>36</td>
<td>CA</td>
<td>0.915.50</td>
</tr>
<tr>
<td>37</td>
<td>MG</td>
<td>3.95</td>
</tr>
</tbody>
</table>
Table 2.7.2.2

GROMOS 37D4 (vacuo) normal van der Waals parameters

<table>
<thead>
<tr>
<th>IAC</th>
<th>TYPE</th>
<th>$C_6^{1/2}$ (IAC)[kcal mol$^{-1}$Å$^{-6/2}$]</th>
<th>$C_{12}^{1/2}$ (IAC)[kcal mol$^{-1}$Å$^{-12/2}$]</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>OM</td>
<td>23.25</td>
<td>421.0</td>
</tr>
<tr>
<td>7</td>
<td>NL</td>
<td>24.13</td>
<td>636.0</td>
</tr>
</tbody>
</table>

The van der Waals' parameters of the 37D4 force field which is to be used for in vacuo calculations, are identical to those of the 37C4 force field which is to be used for calculations including solvent, except of the repulsive parameters of the OM and NL atoms of which the charge has been reduced when deriving the 37D4 force field from the 37C4 one.
| integer | atom code | atom | type | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 24 | 25 | 26 | 27 | 28 | 30 | 31 | 32 | 33 | 34 | 35 | 36 | 37 |
|---------|-----------|------|------|---|---|---|---|---|---|---|---|---|----|----|----|----|----|----|----|----|----|----|----|----|----|----|----|
| 1       | J         | O    | OM   | OA | OW | N  | NT | NL | NR | NRS | NR5 | ZN  | NZ  | NE  | P   | OS  | NR6 | NR6 | ST  | NA  | CL  | CA  | MG |
| 2       |           | O    | OM   | OA | OW | N  | NT | NL | NR | NRS | NR5 | ZN  | NZ  | NE  | P   | OS  | NR6 | NR6 | ST  | NA  | CL  | CA  | MG |
| 3       |           | OA   |      |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |
| 4       |           | OW   |      |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |
| 5       |           | N    |      |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |
| 6       |           | NT   |      |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |
| 7       |           | NL   |      |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |
| 8       |           | NR5  |      |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |
| 9       |           | NR5* |      |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |
| 24      |           | ZN   |      |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |
| 25      |           | NZ   |      |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |
| 26      |           | NE   |      |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |
| 27      |           | P    |      |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |
| 28      |           | OS   |      |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |
| 30      |           | NR6  |      |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |
| 31      |           | NR6* |      |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |
| 32      |           | SI   |      |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |
| 33      |           | NA   |      |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |
| 34      |           | CL   |      |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |
| 35      |           | CA   |      |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |
| 36      |           | MG   |      |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |    |

For an atom pair with integer atom codes I and J, the $c_{12}^{1/2}[I]$ value is taken from the fourth column in Table 2.7.2.1-2 if the matrix element [I,J] equals 1; it is taken from the fifth column of Table 2.7.2.1-2 if it is equal to 2 and from the sixth column if it is equal to 3. Similarly, the $c_{12}^{1/2}[J]$ value is selected using the matrix element [J,I].
2.7.3 Third neighbour van der Waals' interaction

The van der Waals' parameters for the united atoms (CH1, CH2, CH3, CR51, CR61, CS1, CS2) are considerably larger than the ones that are generally used, but reproduce the correct crystal densities of small molecules when tested by applying MD at constant pressure [MD 85.5]. These van der Waals' parameters also avoid the contraction of a protein in vacuo. However, when applied to united atoms that are separated by three covalent bonds, so-called third neighbours, they induce too large a repulsion in gauche conformations.

In order to avoid this effect, the smaller van der Waals' parameters of [MD 82.8] that are given in Table 2.7.3.1 are used for united atoms when obtaining $C_6(I,J)$ and $C_{12}(I,J)$ from (2.7.2,1-2) for third neighbour atoms. The van der Waals parameters for third neighbour or 1-4 interactions are kept in the molecular topology file (Chapter 3.3.2). For the NRATT atom types, CS12[1..NRATT(NRATT+1)/2] contains the coefficient $C_{12}$ for third neighbours in (2.7.1) as a function of the occurring pair codes; the sequence of atom pairs with integer atom codes ranging from 1 to NRATT is:

1-1, 1-2, 2-2, ..., 1-NRATT, 2-NRATT, ..., NRATT-NRATT.

The coefficients $C_6$ for third neighbours in (2.7.1) are kept likewise in CS6[1..NRATT(NRATT+1)/2].

Lists of third neighbour atoms are kept in the molecular topology file (Chapter 3.3.2). For the NRP atoms in the "solute" part of the molecular topology INE14[1..NRP] denotes how many third neighbours j belong to each atom 1..NRP. KNE14[1..NRP] denotes where the atom sequence numbers j of the third neighbours of a "solute" atom can be found in array JSNE14[1..]. The sequence numbers j of atoms that are a third neighbour of the atom with sequence number 1, are positioned at positions KNE14[1]+1, ...
KNE14[1]-INE14[1] in JSNE14. It is assumed that $i<j$. In the solvent part of the molecular topology all NRAM atoms that form a solvent molecule, are excluded neighbour atoms of each other, so will have neither normal nor third neighbour van der Waals' interaction.

The list of third neighbours can be derived from the lists of covalent bonds occurring in the "solute". This is done in program PROGMT, see example files OUTGTM***.LIS.

For the GROMOS 37C and 37D force fields the third neighbour van der Waals' interaction parameters can be found in the interaction function parameter files IPP37**.DAT, and are printed in more readable form in the output of program PROGMT, see example files OUTGTM***.LIS.

Table 2.7.3.1

<table>
<thead>
<tr>
<th>IAC</th>
<th>TYPE</th>
<th>$C_6^{1/2}(IAC)$ [kcal mol$^{-1}$ A$^{-6}$]</th>
<th>$C_12^{1/2}(IAC)$ [kcal mol$^{-1}$ A$^{-12}$]</th>
</tr>
</thead>
<tbody>
<tr>
<td>12</td>
<td>CH1</td>
<td>26.38</td>
<td>945.0</td>
</tr>
<tr>
<td>13</td>
<td>CH2</td>
<td>33.60</td>
<td>1304.0</td>
</tr>
<tr>
<td>14</td>
<td>CH3</td>
<td>40.47</td>
<td>1698.0</td>
</tr>
<tr>
<td>15</td>
<td>CH51</td>
<td>36.35</td>
<td>1411.0</td>
</tr>
<tr>
<td>16</td>
<td>CH51</td>
<td>36.35</td>
<td>1411.0</td>
</tr>
<tr>
<td>29</td>
<td>CS1</td>
<td>26.38</td>
<td>945.0</td>
</tr>
<tr>
<td>32</td>
<td>CS2</td>
<td>33.60</td>
<td>1304.0</td>
</tr>
</tbody>
</table>
2.7.4 Coulomb interaction

The Coulomb interaction in (2.7.1) is inversely proportional to distance \( r_{1j} \) between atoms i and j. Due to this \( 1/r_{1j} \) distance dependence, it is long ranged. The size of the Coulomb interaction at a typical cut-off distance of 0.8-1.0 nm is even for partial atomic charges of size 0.1-0.4 e non-negligible. This would mean that non-bonded Coulomb interactions may only be neglected (cut-off) beyond long distances, like 2.0-3.0 nm. Since the number of non-bonded neighbours grows proportionally to the third power of the cut-off radius, application of a large non-bonded interaction cut-off radius dramatically increases the required computing time.

This situation may be avoided by the application of the following two concepts, which are described below.

- Defining charge groups makes the Coulomb a dipolar interaction, which has a \( 1/r^3 \) distance dependence (Chapter 2.7.5).

- Neglecting the high frequency components of the long-range part of the Coulomb interaction allows for a considerable reduction of the required computing time (Chapter 2.7.7).

In the GROMOS force field the relative dielectric permittivity \( \varepsilon_r = 1 \). The value of \( (4\pi\varepsilon_0)^{-1} \) equals 332.0636 kcal mol\(^{-1}\)e\(^{-2}\)Å\(^{-1}\).

Lists of atomic charges are kept in the molecular topology file (Chapter 3.3.2). For the NRP atoms in the "solute" part of the molecular topology atomic charges (times sqrt(\(4\pi\varepsilon_0)^{-1}\)) are stored in CG[1..NRP]. The atomic charges of the NRAM solvent atoms are stored in CGS[1..NRAM].
2.7.5 Charge groups

When the (partial) atomic charges of a group of atoms add up to exactly zero, the leading term of the electric interaction between two such groups of atoms is of dipolar \((1/r^3)\) character. The sum of the \(1/r\) monopole contributions of the various atom pairs to the group-group interaction will be zero. Therefore, the range of the electric interaction can be considerably reduced when atoms are assembled in so-called charge groups, which have a zero net charge, and for which the electric interaction with other (groups of) atoms is either calculated for all atoms of the charge group or for none.

The GROMOS force fields make use of this concept of charge groups. The atoms that belong to a charge group are chosen such that their partial atomic charges add up to zero. For groups of atoms with a total charge of \(+e\) or \(-e\), like the sidechain atoms of Arg or Asp, the partial atomic charges of the charge group may add up to \(+e\) or \(-e\). In the GROMOS non-bonded interaction subroutines listed in Chapter 2.7, the non-bonded interaction is only calculated between charge groups. When a cut-off radius is used, the distance between two charge groups must be defined. The position of a charge group is defined differently for a charge group belonging to the "solute" part of the molecular topology and one in the "solvent" part of the molecular topology.

- The position of a "solute" charge group is taken to be its center of geometry:

\[
\mathbf{R}_{cg} = \frac{1}{N_{cg}} \sum_{i=1}^{N_{cg}} \mathbf{r}_i
\]

(2.7.5.1)
where the number of atoms belonging to the charge group is denoted by $N_{cg}$.

The position of a "solvent" charge group is taken to be the position of the first atom of a solvent molecule. A "solvent" molecule may only contain one charge group.

Since each solvent molecule consists of one charge group, the "solvent" part of the molecular topology file does not need to contain information on "solvent" charge groups. In the "solute" part of the molecular topology file the charge group information is kept in the following way (Chapter 3.3.2). The number of charge groups in a solute is denoted by $NCAG$. The array INC[1..NCAG] contains a charge group pointer list; INC[I] is the atom sequence number of the last atom of the I-th charge group, where it is assumed that the atoms belonging to one charge group have sequential atom sequence numbers. The latter requirement is a less elegant restriction to choosing the atom sequence when defining residue topology building blocks (Chapter 3.5) or molecular topologies (Chapter 3.3.3).

The atomic charges and charge group definitions for the GROMOS force fields are given in the residue topology building block files R7***.DAT. This information is printed in more readable form in the output of program PROGMT, see example files OUTG7***.LIS. In the residue topology building blocks the last atom of a charge group is denoted by a charge group code 1, whereas the other members of a charge group have charge group code 0. The atomic charges and charge group definitions for amino acid residues, water and nucleotides of the 37C4 and 37D4 GROMOS force fields are listed in Table 2.7.5.1-2.

The atomic charges and charge group definitions for amino acid residues, water and nucleotides of the 37C4 and 37D4 GROMOS force fields are listed in Tables 2.7.5.1-2.
<table>
<thead>
<tr>
<th>Atom Name</th>
<th>Charge in e</th>
<th>Occurring In</th>
</tr>
</thead>
<tbody>
<tr>
<td>N</td>
<td>-0.28</td>
<td>All residues</td>
</tr>
<tr>
<td>H</td>
<td>0.28</td>
<td>All residues</td>
</tr>
<tr>
<td>C</td>
<td>0.38</td>
<td></td>
</tr>
<tr>
<td>O</td>
<td>0.38</td>
<td></td>
</tr>
<tr>
<td>CD</td>
<td>0.09 (.0)</td>
<td></td>
</tr>
<tr>
<td>NE</td>
<td>0.11 (.24)</td>
<td>Arg</td>
</tr>
<tr>
<td>HE</td>
<td>0.24 (.24)</td>
<td></td>
</tr>
<tr>
<td>CZ</td>
<td>0.34 (.0)</td>
<td></td>
</tr>
<tr>
<td>NH1/2</td>
<td>-0.25 (-.48)</td>
<td></td>
</tr>
<tr>
<td>HH11/12/21/22</td>
<td>0.24 (.24)</td>
<td></td>
</tr>
<tr>
<td>C8, CD</td>
<td>0.38</td>
<td>Asn, Gln</td>
</tr>
<tr>
<td>OD1, OE1</td>
<td>0.38</td>
<td></td>
</tr>
<tr>
<td>ND2, NE2, NZ</td>
<td>-0.83</td>
<td>Asn, Gln, Lys</td>
</tr>
<tr>
<td>HD21/22, HE21,22, HZ1/2</td>
<td>0.415</td>
<td></td>
</tr>
<tr>
<td>CG, CD</td>
<td>-0.27 (+.72)</td>
<td>Asp, Gliu</td>
</tr>
<tr>
<td>OD1/2, OE1/2</td>
<td>-.635 (-.36)</td>
<td></td>
</tr>
<tr>
<td>CG, CD</td>
<td>0.53</td>
<td></td>
</tr>
<tr>
<td>OD1, OE1</td>
<td>0.38</td>
<td>Asph, Gliuh</td>
</tr>
<tr>
<td>OD2, OE2</td>
<td>-0.548</td>
<td></td>
</tr>
<tr>
<td>HD2, HE2</td>
<td>0.398</td>
<td></td>
</tr>
<tr>
<td>CB</td>
<td>-.1 (+.2)</td>
<td>Cys</td>
</tr>
<tr>
<td>SG</td>
<td>-.4 (+.2)</td>
<td></td>
</tr>
<tr>
<td>SG</td>
<td>4.64</td>
<td></td>
</tr>
<tr>
<td>-----</td>
<td>------</td>
<td></td>
</tr>
<tr>
<td>HG</td>
<td>0.64</td>
<td></td>
</tr>
<tr>
<td>HD1</td>
<td>0.19</td>
<td></td>
</tr>
<tr>
<td>CD2</td>
<td>0.13</td>
<td></td>
</tr>
<tr>
<td>CE1</td>
<td>0.26</td>
<td></td>
</tr>
<tr>
<td>NE2</td>
<td>-0.58</td>
<td></td>
</tr>
<tr>
<td>CG</td>
<td>0.13</td>
<td></td>
</tr>
<tr>
<td>ND1</td>
<td>-0.58</td>
<td></td>
</tr>
<tr>
<td>CD2</td>
<td>0.0</td>
<td></td>
</tr>
<tr>
<td>CE1</td>
<td>0.26</td>
<td></td>
</tr>
<tr>
<td>NE2</td>
<td>0.0</td>
<td></td>
</tr>
<tr>
<td>HE2</td>
<td>0.19</td>
<td></td>
</tr>
<tr>
<td>CG</td>
<td>-0.05(,0)</td>
<td></td>
</tr>
<tr>
<td>ND1</td>
<td>0.38(-0.3)</td>
<td></td>
</tr>
<tr>
<td>HD1</td>
<td>0.30(+0.3)</td>
<td></td>
</tr>
<tr>
<td>CD2</td>
<td>0.0 (0)</td>
<td></td>
</tr>
<tr>
<td>CE1</td>
<td>-0.24(0)</td>
<td></td>
</tr>
<tr>
<td>NE2</td>
<td>0.31(-0.3)</td>
<td></td>
</tr>
<tr>
<td>HE2</td>
<td>0.30(+0.3)</td>
<td></td>
</tr>
<tr>
<td>CG, CB, CB, CZ</td>
<td>0.15</td>
<td></td>
</tr>
<tr>
<td>OD1, OG, OG1, OH</td>
<td>4.548</td>
<td></td>
</tr>
<tr>
<td>HD1, HG, HG1, HH</td>
<td>0.398</td>
<td></td>
</tr>
<tr>
<td>CE</td>
<td>0.127(0)</td>
<td></td>
</tr>
<tr>
<td>NZ</td>
<td>0.129(-0.744)</td>
<td></td>
</tr>
<tr>
<td>HZ1/2/3</td>
<td>0.248(+0.248)</td>
<td></td>
</tr>
</tbody>
</table>

Cys, His, Hyp, Ser, Thr, Tyr, Lys
CG
CD1
CD2
NE1
HE1
OW
HW1/2

\[ a_{14} \]
\[ 0 \]
\[ 0 \]
\[ -0.05 \]
\[ 0.19 \]
\[ 0.82 \]
\[ 0.41 \]

The charges for the 37D4 force field are given between parentheses. The atoms that are not listed have zero partial charge and form single atom charge groups.
Table 2.7.5.2

GROMOS 37C4 (37D4) atomic charges and charge group definitions

<table>
<thead>
<tr>
<th>atom name</th>
<th>charge in e</th>
<th>occurring in</th>
</tr>
</thead>
<tbody>
<tr>
<td>O3/5*</td>
<td>-.36(-.36)</td>
<td>all nucleotides</td>
</tr>
<tr>
<td>P</td>
<td>.99(1.44)</td>
<td>all nucleotides</td>
</tr>
<tr>
<td>O1/2P</td>
<td>-.635(-.36)</td>
<td>all nucleotides</td>
</tr>
<tr>
<td>C4*</td>
<td>.15</td>
<td>all nucleotides</td>
</tr>
<tr>
<td>O4*</td>
<td>-.36</td>
<td>all nucleotides</td>
</tr>
<tr>
<td>C1*</td>
<td>.20</td>
<td>all nucleotides</td>
</tr>
<tr>
<td>N9,N9,N1,N1</td>
<td>-.20</td>
<td>dAde,dGua,dCyt,dThy</td>
</tr>
<tr>
<td>C4,C4,C6,C6</td>
<td>.20</td>
<td>dAde,dGua</td>
</tr>
<tr>
<td>N3,N3</td>
<td>-.36</td>
<td>dAde,dCyt</td>
</tr>
<tr>
<td>C2,C2</td>
<td>.36</td>
<td></td>
</tr>
<tr>
<td>N1,N3</td>
<td>-.36</td>
<td></td>
</tr>
<tr>
<td>C6,C4</td>
<td>.36</td>
<td></td>
</tr>
<tr>
<td>N6,N2,N4</td>
<td>-.83</td>
<td>dAde,dGua,dCyt</td>
</tr>
<tr>
<td>H61/62,H21/22,H41/42</td>
<td>.415</td>
<td></td>
</tr>
<tr>
<td>N7,N7</td>
<td>-.36</td>
<td>dAde,dGua</td>
</tr>
<tr>
<td>C8,C8</td>
<td>.36</td>
<td></td>
</tr>
<tr>
<td>N1,N3</td>
<td>-.28</td>
<td>dGua,dThy</td>
</tr>
<tr>
<td>H1,H3</td>
<td>.28</td>
<td></td>
</tr>
<tr>
<td>C6,C2,C2</td>
<td>.38</td>
<td>dGua,dCyt,dThy</td>
</tr>
<tr>
<td>O6,O2,O2</td>
<td>.438</td>
<td></td>
</tr>
<tr>
<td>C4</td>
<td>.38</td>
<td>dThy</td>
</tr>
<tr>
<td>O4</td>
<td>.438</td>
<td></td>
</tr>
</tbody>
</table>
The charges for the 37D4 force field are given between parentheses. The atoms that are not listed have zero partial charge and form single atom charge groups.

2.7.6 Searching neighbours

The bulk of the computer time required by a simulation time step is used for calculating the non-bonded interactions, that is, for finding the nearest neighbour atoms and subsequently evaluating the van der Waals and Coulomb interaction terms (2.7.1) for the obtained atom pairs. Therefore, various schemes for performing this task as efficiently as possible are available [MD 54.3]. They are discussed below.

Since the non-bonded interaction between atoms decreases with the distance between them, only interactions between atoms closer to each other than a certain cut-off distance $R_c$ are generally taken into account in simulations. In Chapter 2.7.5 the concept of atom charge groups was introduced in order to reduce the errors due to the application of a cut-off radius $R_c$. Therefore, in the GROMOS non-bonded interaction subroutines the cut-off radius $R_{cG}$, denoted by RCUTP, is used to select nearest neighbour charge groups. That is, the cut-off criterion $R_{cG}$ (RCUTP) is applied to the distance between the positions of the charge groups as defined in Chapter 2.7.5. We note that for periodic systems, the value of $R_{cG}$ must be chosen to be smaller than half the value of the smallest edge of the periodic box in order to avoid the interaction of an atom with both another atom and its periodic image.

Once the nearest neighbour charge groups are selected, it is still possible, but not advised, to apply an atom-atom cut-off radius $R_{cG}$, denoted by RCUT2, before the non-bonded interaction is calculated. However, application of an atom-atom cut-off destroys the gains in accuracy of using
charge groups. Therefore, we generally use $R_c^{\text{min}} = 10$ nm.

2.7.6.1 Scanning all charge group pairs

The simplest way to find the neighbouring charge groups of a charge group, that is, the charge groups that lie within $R_c^{\text{CG}}$, is to scan all possible charge group pairs in the system. For a system consisting of $N_{CG}$ charge groups, the number of pairs amounts $1/2 N_{CG}^2$, which makes the computer time required for finding the neighbours in this way proportional to $N_{CG}^2$. Once the neighbours have been found, the time required for calculating the non-bonded interaction is proportional to $N_{CG}$. We note that non-bonded interactions within a charge group may need to be calculated, when the charge group contains many atoms.

Subroutine NONBP calculates the non-bonded interaction by scanning all charge group pairs in the system and using the charge group cut-off criterion $R_c^{\text{CG}}$. Its use is only recommended when neighbour list techniques as discussed in Chapter 2.7.6.2-3, cannot be applied, due to physical reasons (diffusion of particles) or storage capacity reasons (storage of long neighbour list).

2.7.6.2 Atom-atom neighbour list

The larger the system, the larger is the fraction of time spent on finding the charge group neighbours compared to the fraction spent on calculating the non-bonded interaction from the list of charge group neighbours. Therefore, a considerable reduction of the required computing time can be obtained by storing the neighbour list and updating it only
every n-th (=NSNB) simulation step, where \( n \) is typically of the order of 10. The neighbour list can be stored in two forms.

1. As an atom-atom pair list, which is derived from the charge group pair list (which was selected using \( R_{CG}^{c} \)) by listing all the atom pairs between the selected charge groups.

2. As a charge group pair list containing the pairs of charge groups which were selected using \( R_{CG}^{c} \).

The atom pair list will in general be much longer than the charge group pair list, and so require more storage capacity. But, calculating the non-bonded interaction from an atom pair list will be faster than from a charge group pair list. Using the latter the atom pairs have to be derived from the charge group pairs.

Subroutine NBPAL generates an atom pair list, using the charge group cut-off criterion \( R_{CG}^{c} \). Subroutine NONHAL calculates the non-bonded interaction using this atom pair list.

2.7.6.3 Charge group neighbour list

The advantage of using a charge group pair list compared to an atom-atom pair list as discussed in Chapter 2.7.6.2, lies in the fact that a charge group pair list requires much less storage capacity, while requiring slightly more computing time.

Subroutine NBPMI generates a charge group pair list, using the charge group cut-off criterion \( R_{CG}^{c} \). Subroutine NONPMI calculates the non-bonded interaction using this charge group pair list.

Since these two subroutines are generally used when simulating molecular systems, various special vectorized versions of them are available.
For a **Cray** computer:

- **NBPMLG** and **NONBMLG**: general version
- **NBPMLC** and **NONBMLD**: no periodicity (NTB=0) version
- **NBPMLD**: no periodicity (NTB=0) version, which uses ASSEMBLY routines

For a **Cyber 205** computer:

- **NBPMLW** and **NONBMLW**: general version
- **NBPMLW** and **NONBMLW**: general version (alternative)

For a **Convex** computer:

- **NBPMLX** and **NONBMLX**: general version

For a **Fujitsu VP** computer:

- **NBPMLP** and **NONBMLP**: general version

We note that the standard GROMOS versions **NBPML** and **NONBML** can also be used on these machines, which is useful for test purposes.

### 2.7.6.4 Grid-cell plus linked-list technique

Searching neighbours may also be done by using grid techniques. In such a method a grid (or mesh) consisting of cells covers the space occupied by the molecular system. Neighbours are found by only searching neighbouring grid cells, which makes the required computing time proportional to \( N_{cg} \), but at the expense of considerable overhead due to the grid-cell bookkeeping. The name linked-list technique originates from the way the bookkeeping of which charge groups belong to which grid-cell is
done. The length $h$ of a grid-cell should be chosen such that the cube-corner effect is avoided and the speed is optimized [MD 84,3].

Subroutine NONBB calculates the non-bonded interaction by finding the charge group pairs by the grid-cell plus linked-list technique, using the charge group cut-off criterion $R_{0g}^{cg}$. For very large systems it may be more efficient than the application of the atom or charge group pair list techniques of Chapter 2.7.5.2-3. The parameter CELM is used to determine the length $h$ of a grid-cell.

2.7.7 Long-range Coulomb interaction

In order to evaluate the electrostatic interaction in (2.7.1) with sufficient accuracy, a long cut-off radius $R_{0g}^{cg}$ has to be used; for molecular systems a value of at least 1.5 nm seems necessary. But such a range is very expensive if pair interactions are evaluated; the number of neighbour atoms within 1.5 nm will exceed 300. Therefore, in GROMOS the Coulomb interaction can be evaluated using a twin-range method [MD 85,13].

The Coulomb interactions in (2.7.1) are evaluated at every simulation step using the atom or charge group pair list that is generated with a short range cut-off radius $R_c$ (=RCUTP). The longer range Coulomb interactions, that is, those between charge groups at a distance longer than $R_c$ and smaller than $R_{01}$ (=RCUTL), are evaluated less frequently, viz. only at every $n$-th (=NSNB) simulation step when also the pair list is updated. They are kept unchanged between these updates. In this way the long-range Coulombic forces can be approximately taken into account, without increasing the computing effort significantly, at the expense of neglecting the fluctuation of the forces beyond $R_c$ during $n$ simulation steps. Mathematically, this
method of computing long-range interactions is the first-order approximation of the electric field due to charges in the region between \( R_0 \) and \( R_{cl} \).

Subroutines \texttt{NEPAL} and \texttt{NBPM} (and its vectorized versions) yield the possibility of treating the long-range Coulomb interaction in this way. We note that for periodic systems the value of \( R_{cl} \) \((-RCUTL)\) must be chosen to be smaller than half the value of the smallest edge of the periodic box in order to avoid the interaction of an atom with both another atom and its periodic image. The long-range interaction, which is calculated for charge group pairs at distances between \( R_0 \) \((-RCUTP)\) and \( R_{cl} \) \((-RCUTL)\), is evaluated by using only the plain Coulomb term in (2.7.1). A switching function is not applied and no excluded neighbour lists are used. So, \( R_0 \) must not be chosen too small.

2.7.8 Switching functions

The function \( S(r) \) in (2.7.1) is added to the non-bonded interaction in order to allow for a smooth behaviour of the interaction function when a cut-off radius \( R_0^a \) is used. Its form is chosen such that

\[
\begin{align*}
\text{a: } S(r) &= 1 \text{ and } \frac{dS(r)}{dr} = 0 \text{ at } r = R_0^a \text{ (=RSWI2)} \\
\text{b: } S(r) &= 0 \text{ and } \frac{dS(r)}{dr} = 0 \text{ at } r = R_0^a \text{ (=RCUT2)}
\end{align*}
\]

(2.7.8.1)

From these conditions one finds

\[
S(r) = \frac{[R_0^a - r]^2[R_0^a + 2r - 3R_s^a]/[R_0^a - R_s^a]^2}{[R_0^a - r]^2[R_0^a + 2r - 3R_s^a]/[R_0^a - R_s^a]^2}
\]

(2.7.8.2)

and
Subroutines NONBE, NONBAL, NONBML (and its vectorized versions) and NONBE offer the possibility to use the given switching function $S(r)$ of (2.7.8.2). However, application of atom-atom distance switching functions is not recommended, as discussed in Chapter 2.7.6. We generally use $R_s^a = R_c^a = 0$ or equivalently $R_SWI2 = R_CUT2 = 10$ nm.
2.8 Special forces

2.8.1 Introduction

The interaction potential $V(r_i)$ as given in formula (2.1.1) may contain special terms, which are only used under special circumstances. Here, two of such special terms will be discussed. Position restraining of atoms to given spatial reference positions by a harmonic force will be described in Chapter 2.8.2. Another type of restraining, atom-atom distance restraining by a special interaction term acting on the distance between specified atom pairs, will be described in Chapter 2.8.3.

When the interaction potential ($r_i$) refers to non-atomic sites as centers of interaction, it is still possible to decompose the force on a non-atomic site into forces on those atoms, of which the atomic positions are used to define the position of the non-atomic site. The use of virtual and pseudo atoms is discussed in Chapter 2.8.4.

The possibility to restrain bond-angles or dihedral angles is discussed in Chapters 2.8.5 and 2.8.6.
2.8.2 Position restraining

When simulating a molecular system it may be desirable to fix specific atoms or parts of the system. In molecular dynamics it is not advisable to immobilize atoms completely because this may reduce the flexibility of the system such that transitions and motions that are normally displayed by the system, are completely inhibited by the rigidity of the atoms. Therefore, a better way to keep specific atoms approximately to given reference positions is to restrain the motion of those atoms around these positions by applying a harmonic force, which still leaves room for flexibility and mobility.

The form of the special term in $V(\{r_i\})$ (2.1.1) that performs atom position restraining reads

$$\sum_{n=1}^{N^{pr}} \frac{1}{2} k_n^{pr} (\mathbf{r}_n - \mathbf{r}_n^0)^2$$

(2.8.2.1)

The summation runs over a set of $N^{pr}$ specified atoms. The fixed reference positions are denoted by $\mathbf{r}_n^0$. The harmonic oscillator force constants $k_n^{pr}$ can be chosen

a) to be equal for all specified atoms, or
b) to be inversely proportional to the atomic B-factor.

The actual position of the n-th restrained atom is denoted by $\mathbf{r}_n$. The force on atom $n$ due to the n-th term in formula (2.8.2.1) is

$$\mathbf{F}_n = -k_n^{pr} (\mathbf{r}_n - \mathbf{r}_n^0)$$

(2.8.2.2)

The interaction energy and forces are calculated in subroutine RESTX.

The atom sequence numbers of the restrained atoms are given in JRC[1..NRC].
These are used to select atom reference coordinates from XCG[1..3*NR] and force constants from CXC[1..NR]. Position restraining can be applied to any atom of solute or solvent.

Applications of atom position restraining are the following.

1. When a solute is placed in a box with solvent molecules, the solute-solvent atomic contacts may be very unfavourable. When performing energy minimization or molecular dynamics starting from such a configuration the solute conformation may be distorted by the bad non-bonded contacts with the solvent molecules. By applying atom position restraining to the solute atoms the unfavourable contacts can be relaxed without changing the solute conformation.

2. When only a part of a molecular system is simulated in vacuo without applying periodic boundary conditions, it is necessary to restrain the atoms in the wall region in order to avoid distortion of the system due to the vacuo boundary condition, see Chapter 2.10.4.

2.8.3 Distance restraining

When simulating a molecular system it may be desirable to restrain the distance between specific atoms to a specified value or to only a minimum value or maximum value. This can be performed by using as a special term in the interaction function \( V(\{r_i\}) \) a harmonic oscillator or one half of a harmonic oscillator (for lower or upper bound). However, when the actual distance \( r_{nn'} \) between atoms \( n \) and \( n' \) is very different from the reference distance \( r_{nn'}^0 \), the energy and force due to a harmonic function will be huge. Therefore, the special interaction term for atom-atom distance restraining is chosen to become linear beyond a specified deviation \( \Delta r^h = r^1 - r^0 \) of \( r_{nn'} \) from \( r_{nn'}^0 \), see Fig. 2.8.3.1.
Figure 2.8.3.1

POTENTIAL ENERGY TERM FOR DISTANCE CONSTRAINTS

- Repulsive
- Attractive

**ENERGY**
- Linear
- Harmonic
- Zero

**FORCE**
- Constant
- Linear
- Zero

---

Figure 2.8.3.1
The form of the special term in $V(r_{ij})$ (2.1.1) that performs atom-atom distance restraining reads

$$
\sum_{\mathbf{d}} \sum_{n=1}^{N} V_{dc}(r_{nn'}, r_m^0) (2.8.3.1)
$$

where the $n$-th atom-atom distance restraint involves atoms denoted by $n$ and $n'$. An attractive distance restraint $n$ with length $r_m^0$ between atoms $n$ and $n'$ is represented by

$$
V_{dc}(r_{nn'}, r_m^0) = 0 \quad 0 < r_{nn'} < r_m^0
= \frac{1}{2} K_m^{dc} (r_{nn'} - r_m^0)^2 \quad r_m^0 < r_{nn'} < r_m^0 + \Delta r^h
= K_m^{dc} [r_{nn'} - r_m^0 - 1/2\Delta r^h] \Delta r^h \quad r_m^0 + \Delta r^h < r_{nn'} \quad (2.8.3.2)
$$

The actual distance between atoms $n$ and $n'$ is denoted by $r_{nn'}$, and

$$
r_m^1 = r_m^0 + \Delta r^h \quad \text{is the distance at which } V_{dc} \text{ changes from a quadratic to a linear function of } r_{nn'}^1.
$$

The forces on atoms $n$ and $n'$ due to $V_{dc}$ in (2.8.3.2) are

$$
\vec{F}_n = 0 \quad 0 < r_{nn'} < r_m^0
= -K_m^{dc} [r_{nn'} - r_m^0] \frac{\partial V_{dc}}{\partial r_{nn'}} \quad r_m^0 < r_{nn'} < r_m^0 + \Delta r^h
= -K_m^{dc} \Delta r^h \frac{\partial V_{dc}}{\partial r_{nn'}} \quad r_m^0 + \Delta r^h < r_{nn'} \quad (2.8.3.3)
$$

and

$$
\vec{F}_{n'} = -\vec{F}_n \quad (2.8.3.4)
$$
A repulsive distance restraint $n$ with length $r^0_n$ between atoms $n$ and $n'$ is represented by

$$V_{dc}(r_{nn}', r_n^0) = -k_m^{dc}[r_{nn}', -r_n^0 + 1/2\Delta r^h] \Delta r^h$$

$$= 1/2 \ k_m^{dc}[r_{nn}', -r_n^0]^2$$

$$= 0$$

$$0 < r_{nn}' < r_n^0 - \Delta r^h$$

$$r_n^0 - \Delta r^h < r_{nn}' < r_n$$

$$r_n^0 < r_{nn}'$$

(2.8.3.5)

where $r_n^0 - r_n^0 - \Delta r^h$ is the distance at which $V_{dc}$ changes from a linear to a quadratic function of $r_{nn}'$.

The forces on atoms $n$ and $n'$ due to $V_{dc}$ in (2.8.3.5) are

$$\vec{F}_n = k_m^{dc} \Delta r^h[\vec{r}_{nn}' / r_{nn}']$$

$$= -k_m^{dc}[r_{nn}', -r_n^0][\vec{r}_{nn}' / r_{nn}']$$

$$= 0$$

$$0 < r_{nn}' < r_n^0 - \Delta r^h$$

$$r_n^0 - \Delta r^h < r_{nn}' < r_n$$

$$r_n^0 < r_{nn}'$$

(2.8.3.6)

and

$$\vec{F}_{n'} = -\vec{F}_n$$

(2.8.3.7)

The interaction energy and forces are calculated in subroutine DISRE.

The specification of the atoms $n$ and $n'$ is discussed in Chapter 2.8.4. The force constants $k_m^{dc}$ are denoted by CB[1..NB]. They can be chosen:

- to be equal for all specified distance restraints, or

- to be proportional to a distance restraint weight factor WO[1..NB].

The reference distances $r_n^0$ are denoted by BO[1..NB] = RO[1..NDR], where NB = NDR. Attractive restraining is selected when BO = RO > 0, whereas repulsive restraining is selected by changing the sign of BO = RO (< 0). The distance
$A^h$ is denoted by $DBO = DRO$. Distance restraining cannot be applied to atoms of solvent molecules (Chapter 3.3.1).

Applications of atom-atom distance restraining are the following.

- When a molecular structure is to be obtained that satisfies a set of given distance constraints, atom-atom distance restraining can be used during EM or MD simulation to force the molecular conformation in the desired direction.

- When a part of a molecule is required to keep its form during a simulation, atom-atom distance restraining can be used to fix relative atom positions of a group of atoms without restraining the mobility of the group.
2.8.4 Virtual and pseudo atoms

In the GROMOS force fields hydrogen atoms that are attached to carbon atoms are not explicitly treated, but are incorporated in the latter forming united atoms (Chapter 2.2). However, a distance restraint which is derived from 24-dimensional proton nuclear magnetic resonance experiments, may refer to such a hydrogen atom, which is called a virtual atom. In that case the distance restraint interaction (2.8.3.1-7) refers to a non-atomic site as a center of interaction. A proton-proton distance restraint may also refer to non-atomic sites when a stereo-specific assignment of a resonance to a proton cannot be obtained (e.g. C$^\beta$ protons in proteins or methyl groups in the amino acid residues Leu and Val) or when dynamic effects such as rotation of methyl group hydrogens and flipping of aromatic rings occur. In these cases the distance restraint must refer to a pseudo atom and a correction term $\Delta r^p$ must be added to the restraint distance $r^0_{nn'}$:

$$r^0_{nn'}(\text{pseudo}) = r^0_{nn'}(\text{real}) + \Delta r^p_n + \Delta r^p_{n'}$$  \hspace{1cm} (2.8.4.1)

The value of $\Delta r^p_n$ depends on the geometry of construction of the pseudo atom site and will be given below in Table 2.8.4.1. So, virtual and pseudo sites or atoms are massless points, whose positions are rigorously related to the positions of the masses in the molecule.

When using pseudo or virtual sites, two additional steps are added to the calculation of the forces on the real atoms.
Virtual and pseudo hydrogen atoms and distance constraint correction terms.

<table>
<thead>
<tr>
<th>group</th>
<th>configuration</th>
<th>atom type</th>
<th>correction term $\Delta r^{PS}$</th>
<th>code ICBI on distance constraint or ICBI</th>
<th>$r^e$ (nm)</th>
<th>in subr. DISRE</th>
</tr>
</thead>
<tbody>
<tr>
<td>CH₁ (aliphatic)</td>
<td>virtual</td>
<td></td>
<td>.0</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>CH₁ (aromatic)</td>
<td>virtual</td>
<td></td>
<td>.0</td>
<td>2</td>
<td></td>
<td></td>
</tr>
<tr>
<td>CH₂ (stereospecific)</td>
<td>virtual</td>
<td></td>
<td>.0</td>
<td>4</td>
<td></td>
<td></td>
</tr>
<tr>
<td>CH₂ (non-stereospecific)</td>
<td>pseudo</td>
<td></td>
<td>.09</td>
<td></td>
<td>3</td>
<td></td>
</tr>
<tr>
<td>CH₃</td>
<td>pseudo</td>
<td></td>
<td>.10</td>
<td></td>
<td>5</td>
<td></td>
</tr>
<tr>
<td>CH₃ (non-stereospecific, Val, Leu)</td>
<td>pseudo</td>
<td></td>
<td>.22</td>
<td></td>
<td>6</td>
<td></td>
</tr>
<tr>
<td>CH₁ (aromatic at C₆ or C₇ in Phe and Tyr)</td>
<td>pseudo</td>
<td></td>
<td>.21</td>
<td></td>
<td>0</td>
<td></td>
</tr>
</tbody>
</table>
1. Before the interaction term involving virtual or pseudo sites like
(2.8.3.1, 2, 5) can be evaluated, the virtual or pseudo site \( n \) must be
constructed using the positions of those atoms \((i,j,k,l)\) that define
the virtual or pseudo atom \( n \).

2. The force \( F_n \) acting on the virtual or pseudo site \( n \) must be redistributed
over the atoms \((i,j,k,l)\) that define the virtual or pseudo atom \( n \). The
contribution of \( F_n \) to the force an atom \( i \) is e.g.

\[
\frac{F_i}{F_n} = \frac{\partial x_n}{\partial x_i} + \frac{\partial y_n}{\partial y_i} + \frac{\partial z_n}{\partial z_i}
\]  

(2.8.4.2)

or in matrix notation

\[
\begin{pmatrix}
F_{ix} \\
F_{iy} \\
F_{iz}
\end{pmatrix} = \begin{pmatrix}
\partial x_n/\partial x_i & \partial y_n/\partial x_i & \partial z_n/\partial x_i \\
\partial x_n/\partial y_i & \partial y_n/\partial y_i & \partial z_n/\partial y_i \\
\partial x_n/\partial z_i & \partial y_n/\partial z_i & \partial z_n/\partial z_i
\end{pmatrix} \begin{pmatrix}
F_{nx} \\
F_{ny} \\
F_{nz}
\end{pmatrix}
\]  

(2.8.4.3)

Corresponding formulae hold for atoms \( j, k \) and \( l \). The matrices of
partial derivatives \( \partial x_n/\partial \hat{r}_i \), etc. can easily be derived from the
definition of \( \hat{r}_n \) in terms of \( \hat{r}_i, \hat{r}_j, \hat{r}_k \) and \( \hat{r}_l \). These definitions
have been kept as simple as possible in order to avoid too complex
derivatives.

The virtual and pseudo atoms that can be used when the distance
restraining potential energy term (2.8.3.1, 2, 5) is applied, are displayed
in Table 2.8.4.1. (See also [MD 85.6]) Three types of virtual atoms are
distinguished:

A. CHI-group (aliphatic)
The position vector of the hydrogen $H_n$ is given by

$$\vec{r}_n = \vec{r}_1 + d\vec{s}/s$$  \hspace{1cm} (2.8.4.4)

with

$$\vec{s} = 3\vec{r}_1 - \vec{r}_j - \vec{r}_k - \vec{r}_l$$  \hspace{1cm} (2.8.4.5)

and

$$s = [s_x^2 + s_y^2 + s_z^2]^{1/2}$$  \hspace{1cm} (2.8.4.6)

For the carbon-hydrogen distance (DISH) we chose $d = 0.1$ nm. The partial derivatives are

$$\partial r_n / \partial r_1 = \begin{pmatrix}
[1 + 3d(s_x^2 - s_y^2)/s^3] & -3ds_y/s \frac{3}{s} & -3ds_z/s \frac{3}{s} \\
-3ds_x/s \frac{3}{s} & [1 + 3d(s_y^2 - s_z^2)/s^3] & -3ds_z/s \frac{3}{s} \\
-3ds_x/s \frac{3}{s} & -3ds_y/s \frac{3}{s} & [1 + 3d(s_z^2 - s_y^2)/s^3]
\end{pmatrix}$$  \hspace{1cm} (2.8.4.7)

and

$$\partial r_n / \partial r_j = \begin{pmatrix}
-d(s_x^2 - s_y^2)/s^3 & ds_y/s \frac{3}{s} & ds_z/s \frac{3}{s} \\
-3ds_x/s \frac{3}{s} & -d(s_y^2 - s_z^2)/s^3 & ds_z/s \frac{2}{s} \\
ds_x/s \frac{3}{s} & ds_y/s \frac{3}{s} & -d(s_z^2 - s_x^2)/s^3
\end{pmatrix}$$  \hspace{1cm} (2.8.4.8)

and $\partial r_n / \partial r_k$ and $\partial r_n / \partial r_l$ are identical to (2.8.4.8)
B. CH1-group (aromatic)

The position vector of the hydrogen $H_n$ is given by

$$\mathbf{r}_n = \mathbf{r}_1 + ds/s \quad (2.8.4.9)$$

with

$$\hat{s} = 2\mathbf{r}_1 - \mathbf{r}_j - \mathbf{r}_k \quad (2.8.4.10)$$

and $s$ defined by (2.8.4.6). Again, the carbon-hydrogen distance $d = 0.1$ nm. The partial derivatives are

$$\frac{\partial \mathbf{r}_n}{\partial \mathbf{r}_1} = \begin{pmatrix} [1+2d(s^2-s_x^2)/s^3] & -2ds_y s_x / s^3 & -2ds_z s_x / s^3 \\ -2ds_x s_y / s^3 & [1+2d(s^2-s_y^2)/s^3] & -2ds_z s_y / s^3 \\ -2ds_x s_z / s^3 & -2ds_y s_z / s^3 & [1+2d(s^2-s_z^2)/s^3] \end{pmatrix} \quad (2.8.4.11)$$

and $\frac{\partial \mathbf{r}_n}{\partial \mathbf{r}_j}$ and $\frac{\partial \mathbf{r}_n}{\partial \mathbf{r}_k}$ are given by (2.8.4.8).

C. CH2-group (two virtual protons)

The position vector of the hydrogen $H_n^1$ is given by

$$\mathbf{r}_{n^1} = \mathbf{r}_1 + a\hat{s}/s + \hat{s}\hat{s}/s \quad (2.8.4.12)$$

with $\hat{s}$ and $s$ defined by (2.8.4.10) and (2.8.4.6)

and
\[ \mathbf{\dot{w}} = (\mathbf{r}_i - \mathbf{r}_j) \times (\mathbf{r}_i - \mathbf{r}_k) \]  
(2.8.4.13)

with

\[ w = (w_x^2 + w_y^2 + \frac{1}{2}w_z^2)^{1/2} \]  
(2.8.4.14)

The values of \( \alpha \) and \( \beta \) are derived from

\[ \alpha = d \cos(\theta/2) \]
\[ \beta = d \sin(\theta/2) \]  
(2.8.4.15)

where \( \theta \) is the tetrahedral angle and the carbon-hydrogen distance \( d = 0.1 \) nm. The partial derivatives are \( \left( n = n^1 \right) \):

\[ \frac{\partial \mathbf{\dot{r}}}{\partial \mathbf{r}_i} = \mathbf{A} + \mathbf{B} + \mathbf{C} \]  
(2.8.4.16)

with

\[ A = \begin{pmatrix} \frac{1 + 2 \alpha (s^2 - a_x^2)}{s^3} & -2\alpha a_x a_y / s^3 & -2\alpha a_x a_z / s^3 \\ -2\alpha a_y a_x / s^3 & \frac{1 + 2 \alpha (s^2 - a_y^2)}{s^3} & -2\alpha a_y a_z / s^3 \\ -2\alpha a_z a_x / s^3 & -2\alpha a_z a_y / s^3 & \frac{1 + 2 \alpha (s^2 - a_z^2)}{s^3} \end{pmatrix} \]  
(2.8.4.17)

and

\[ B = \begin{pmatrix} -\beta w_x a_x / \omega^3 & -\beta w_y a_x / \omega^3 & -\beta w_z a_x / \omega^3 \\ -\beta w_x a_y / \omega^3 & -\beta w_y a_y / \omega^3 & -\beta w_z a_y / \omega^3 \\ -\beta w_x a_z / \omega^3 & -\beta w_y a_z / \omega^3 & -\beta w_z a_z / \omega^3 \end{pmatrix} \]  
(2.8.4.18)
and

\[
C = \begin{pmatrix}
0 & \frac{8b_z}{w} & -\frac{8b_y}{w} \\
-\frac{8b_z}{w} & 0 & \frac{8b_x}{w} \\
\frac{8b_y}{w} & -\frac{8b_x}{w} & 0
\end{pmatrix}
\] (2.8.4.19)

where \( \vec{a} = \vec{w} \times \vec{r}_{k} \) and \( \vec{b} = \vec{r}_{j} \), and

\[
\frac{\partial^2}{\partial \vec{r}_{n}^2} = D + E + F
\] (2.8.4.20)

where

\[
D = \begin{pmatrix}
-\alpha(s^2 - s_x^2)/s_z^3 & \alpha s_x/s_z^3 & \alpha s_x^2/s_z^3 \\
\alpha s_x/s_z^3 & -\alpha(s^2 - s_y^2)/s_z^3 & \alpha s_y/s_z^3 \\
\alpha s_x^2/s_z^3 & \alpha s_y/s_z^3 & \alpha(s^2 - s_z^2)/s_z^3
\end{pmatrix}
\] (2.8.4.21)

and \( \vec{E} \) is identical to \( \vec{B} \) in (2.8.4.18) but with \( \vec{b} = \vec{w} \times \vec{r}_{1k} \) and \( \vec{F} \) is identical to \( \vec{C} \) in (2.8.4.19) but with \( \vec{b} = \vec{r}_{1k} \), and

\[
\frac{\partial^2}{\partial \vec{r}_{n}^2} = D + G + H
\] (2.8.4.22)

where \( \vec{G} \) is identical to \( \vec{B} \) in (2.8.4.18) but with \( \vec{b} = \vec{w} \times \vec{r}_{j} \) and \( \vec{H} \) is identical to \( \vec{C} \) in (2.8.4.19) but with \( \vec{b} = \vec{r}_{j} \).

The position vector \( \vec{r}_{n_2} \) of the second virtual proton \( n_2 \) can be obtained from the same formulae by interchanging \( j \) and \( k \).

Four types of pseudo atoms are distinguished.
D. CH2-groups (one pseudo site)

When a distance restraint refers to one proton of a CH2-group of which no stereospecific assignment is known, the restraint is referred to a pseudo site between the two protons. The position vector of the pseudo atom $n$ is defined by

$$\mathbf{r}_n = \mathbf{r}_1 + \alpha \mathbf{s} / s$$  \hspace{1cm} (2.8.4.23)

where $\alpha$ is given by (2.8.4.15) and $s$ is defined by (2.8.4.10). The partial derivatives are

$$\partial \mathbf{r}_n / \partial \mathbf{r}_1 = \Lambda$$  \hspace{1cm} (2.8.4.24)

and

$$\partial \mathbf{r}_n / \partial \mathbf{r}_j = \partial \mathbf{r}_n / \partial \mathbf{r}_k = \Delta$$  \hspace{1cm} (2.8.4.25)

E. CH3-group (one pseudo site)

For a single methyl group and for a methyl group of a diastereotopic pair for which the stereospecific assignment is known, the pseudo atom $n$ is defined to be in the middle of the three hydrogens:

$$\mathbf{r}_n = \mathbf{r}_1 + \gamma \mathbf{s} / s$$  \hspace{1cm} (2.8.4.26)

with
\[ Y = d \cos(\pi - \theta) \quad (2.8.4.27) \]

and

\[ \mathbf{a} = \mathbf{r}_i - \mathbf{r}_j \quad (2.8.4.28) \]

Here \( d = 0.1 \text{ nm} \) and \( \theta \) is tetrahedral.

The partial derivatives are

\[ \frac{\partial^3}{\partial \mathbf{r}_i} = \begin{pmatrix}
[1 + \gamma(s^2_x - s^2_y)/s^3] & -\gamma s_x y / s^2 & -\gamma s_x z / s^2 \\
-\gamma s_y x / s^2 & [1 + \gamma(s^2_y - s^2_z)/s^3] & -\gamma s_y z / s^2 \\
-\gamma s_z x / s^2 & -\gamma s_z y / s^2 & [1 + \gamma(s^2_z - s^2_x)/s^3]
\end{pmatrix} \quad (2.8.4.29) \]

and \( \frac{\partial^3}{\partial \mathbf{r}_j} \) is identical to \( \mathcal{D} \) in (2.8.4.21) but with \( a \) replaced by \( Y \).

**F. Two CH3-groups: (one pseudo site)**

When a restraint involves a methyl group of a stereotopic pair for which no stereospecific assignments are known, the restraint is referred to a pseudo atom \( n \) at the geometric mean position of the 6 hydrogens of the pair:

\[ \mathbf{r}_n = \mathbf{r}_1 + \delta \mathbf{a}/s \quad (2.8.4.30) \]

with

\[ \delta = -\cos(\theta/2)[d_{co} + \gamma] \quad (2.8.4.31) \]
where \( \hat{\theta} \) is defined by (2.8.4.10), the carbon-carbon distance (DISC) is chosen \( d_{cc} = 0.153 \) nm, \( \hat{\theta} \) is tetrahedral and \( \gamma \) is given by (2.8.4.27).

The partial derivative \( \partial \Phi / \partial \hat{\theta} \) is identical to \( A \) in (2.8.4.17) but with \( \alpha \) replaced by \( \hat{\theta} \), \( \partial \Phi / \partial \hat{\beta} \) and \( \partial \Phi / \partial \hat{\theta} \) are identical to \( B \) in (2.8.4.21) but with \( \alpha \) replaced by \( \hat{\theta} \).

G. Two CH1-groups on a flipping ring (one pseudo site)

For the \( \delta \) and \( \epsilon \) protons of tyrosine and phenylalanine rings that are displaying fast 180° ring flips, the restraint should be referred to a pseudo site in the middle between the two protons. For the two \( C_\delta \) protons this site is close to the \( C_\gamma \) atom and for the two \( C_\epsilon \) protons it is close to the \( C_\delta \) atom. Therefore, these atoms are taken as pseudo sites in this case.

The construction of virtual and pseudo atom sites and the redistribution of the forces on these sites over the atoms that define a site, is done in subroutine DISRE. The various atom types are selected using the code shown in Table 2.8.4.1. For the first atom \( n_1 \) of the pairs these codes are listed in \( \text{ICB1}[1..NB] = \text{ICDR1}[1..NDR] \) and the atom sequence numbers of the atoms \( I, J, K \) and \( L \) that define atom \( n_1 \) are given in \( \text{IB1}, \text{JB1}, \text{KB1}, \text{LB1}[1..NB] = \text{IDR1}, \text{JDR1}, \text{KDR1}, \text{LDR1}[1..NDR] \). For the second atom \( n_2 \) of the pairs the corresponding arrays are denoted by \( \text{ICB2} \), etc. The carbon-hydrogen distance is denoted by DISH and the carbon-carbon distance by DISC.

We note that the correction terms \( \Delta \rho_n^{ps} \) which are listed in Table 2.8.4.1 are not automatically incorporated into the specified restraint.
distance $r^0$ by applying (2.8.4.1). The user must incorporate the correction terms $\Delta r_n^{\text{ps}}$ into $B_0[1..\text{NB}] = R_0[1..\text{NDR}]$.

### 2.8.5 Bond-angle restraining

One may wish to restrain a bond-angle to a specified value. For this case of restraining no special subroutines have been made in GROMOS. The bond-angle $\theta(i-j-k)$ between atoms $i$, $j$ and $k$ can be restrained by applying distance restraints (Chapter 2.8.3) to the three distances $i-j$, $j-k$ and $i-k$. An alternative is to add to the molecular topology of a molecular system the bond angle $\theta(i-j-k)$ and to choose appropriate values for $K_\theta$ and $\theta_0$ (Chapter 2.4).

### 2.8.6 Dihedral angle restraining

Dihedral angles can be restrained by specifying the atom sequence numbers $i$, $j$, $k$ and $l$ of the atoms forming dihedral angle $\phi(i-j-k-l)$ together with the dihedral interaction parameters $K_\phi$, $\delta$ and $n$ (Chapter 2.6). The form of the term in $V(\{r_i\})$ (2.1.1) that performs dihedral restraining reads

$$V = \sum_{n=1}^{N_{\text{dih}}} K_n^{\text{dih}} \left[ 1 + \cos(n_n^{\phi} - \delta_n) \right]$$

just like (2.6.1). This interaction energy and the forces are calculated in subroutine DIHANG with switch IDI=1. The dihedrals and parameters are to be specified in a dihedral restraints file (Chapter 3.7.5). The atom sequence numbers of the atoms $i$, $j$, $k$ and $l$ are stored in arrays IFLR, JFLR, KFLR,
LPLR[1..NDLR] and the corresponding interaction parameters $K_{DLR}$, $\delta$ and $\eta$ in arrays CPLR, PDLR, NPLR[1..NDLR]. The dihedral restraining term (2.8.6.1) is finally multiplied by a normalization constant CDLR.

Dihedral restraining may be applied when experimental information on spin-spin J-coupling constants is available, or when one would like to force a molecular conformation into another one.
2.9 CONSTRAINTS

2.9.1 Introduction

The aim of applying constraints in dynamical simulations is to save computer time. The length of the time step $\Delta t$ in a MD or SD simulation is limited by the highest frequency $v_{\text{max}}$ occurring in the molecular system of interest:

$$\Delta t \ll v_{\text{max}}^{-1} \quad (2.9.1.1)$$

By constraining the degrees of freedom with the highest frequencies the time step $\Delta t$ can generally be lengthened, which reduces the computer time required for a simulation of a specific length. For molecular systems one may think of reducing the computational effort by constraining bond-lengths or additionally bond-angles.

The application of constrained dynamics makes sense only when:

1. the frequencies of the frozen degrees of freedom are (considerably) higher than those of the remaining ones, thereby allowing a (considerable) increase of $\Delta t$, and when
2. the frozen degrees of freedom are only weakly coupled to the remaining ones, viz. when the motion of the molecules is not affected by application of the constraints, and when
3. metric tensor effects due to constraining the molecules to a hypersurface in configuration space, play a minor role.
For biomolecules the effect of constraining bond lengths and bond angles has been evaluated [MD 82.8]. It turns out that the application of bond-length constraints saves about a factor of 2 in computer time when hydrogen atoms are explicitly treated and a factor of 3 when the united-atom model is used [MD 77.1]. Metric tensor corrections play no role when only bond-length constraints are applied [MD 80.2]. The use of bond-angle constraints is not allowed in flexible (viz. with rotational internal degrees of freedom) molecules, since it affects the dynamics of such molecules considerably [MD 82.8]. Moreover, metric tensor effects are non-negligible in this case [MD 80.2]. The bond-angle degrees of freedom appear to be coupled to the other molecular degrees of freedom, such as the torsional angle ones.

However, for completely rigid molecules, that is, without internal degrees of freedom, metric tensor effects play no role, and the application of bond-length and bond-angle constraints is common practice.

For application of constraints CROMOS uses the SHAKE-method, which will be briefly described in Chapter 2.9.2. The application of bond-length constraints to a solute molecule is described in Chapter 2.9.3, and the use of constraints in rigid solvent molecules is discussed in Chapter 2.9.4. A review of the use of constraints can be found in [MD 84.6].
2.9.2 Constraints using subroutine SHAKE

The molecular constraints that are considered, are only a function of the atomic coordinates \( \{\vec{r}_i\} \). They are of the form

\[
c_k(\vec{r}_1, \ldots, \vec{r}_{N_{at}}) = 0 \quad k = 1, \ldots, N_c
\]  

for the case of \( N_c \) constraints in a system of \( N_{at} \) atoms. In order to be more specific the constraint equations (2.9.2.1) are put in the form of distance constraints between atoms \( k_1 \) and \( k_2 \):

\[
r_{k_1,k_2}^2 - d_{k_1,k_2} = 0
\]  

(2.9.2.2)

where \( r_{k_1,k_2} \) is defined like in (2.3.3-4) and the \( k \)-th constraint involves atoms \( k_1 \) and \( k_2 \).

When applying constraints in MD or SD, the \( 3N_{at} \) equations of motion have to be integrated, while satisfying the \( N_c \) constraints. This can be accomplished by applying Lagrange's method of undetermined multipliers. A zero term is added to the potential energy in the equations of motion:

\[
m_i \frac{d^2 \vec{r}_i(t)}{dt^2} = -\nabla V(\{\vec{r}_i(t)\}) + \sum_{k=1}^{N_c} \lambda_k(t) \frac{\partial}{\partial \vec{r}_i(t)} c_k
\]  

(2.9.2.3)

It allows to satisfy the constraints at all times by solving for \( \lambda_k(t) \), which are the time-dependent multipliers.

The physical interpretation of (2.9.2.3) becomes clear by rewriting it in terms of forces:
\[ m_1 \frac{d^2 \vec{r}_1}{dt^2} = \vec{F}_1 + \vec{G}_1 \]  \hspace{1cm} (2.9.2.4) \\

with

\[ \vec{F}_1 = -\vec{\nabla}_1 V (\{ \vec{r}_i \}) \]  \hspace{1cm} (2.9.2.5) \\

and

\[ \vec{G}_1 = - \sum_{k=1}^{N_0} \lambda_k \frac{\partial}{\partial \vec{r}_1} \varphi_k \]  \hspace{1cm} (2.9.2.6) \\

The total unconstrained force on atom 1, derived from an interaction function potential (eq. (2.1.1)), is denoted by \( \vec{F}_1 \) and \( \vec{G}_1 \) is the constraint force, which compensates the components of \( \vec{F}_1 \) that act along the directions of the constraints. In the leap-frog MD algorithm, which is used in GROMOS, solving for \( \lambda_k \) is done as follows. In the leap-frog algorithm (Chapter 2.12.5) including constraint forces the following steps are made:

\[ \vec{v}_1(t + \Delta t/2) = \vec{v}_1(t - \Delta t/2) + m_1 \left[ \vec{F}_1(t) + \vec{G}_1(t) \right] \Delta t \]  \hspace{1cm} (2.9.2.7) \\

and

\[ \vec{r}_1(t + \Delta t) = \vec{r}_1(t) + \vec{v}_1(t + \Delta t/2) \Delta t \]  \hspace{1cm} (2.9.2.8) \\

Separating the contributions of \( \vec{F}_1 \) and \( \vec{G}_1 \) one finds

\[ \vec{F}_1(t + \Delta t) = \vec{F}_1(t) + \delta \vec{F}_1 \]  \hspace{1cm} (2.9.2.9)
with

$$\delta \dot{R}_1 = m_1^{-1} \delta_1(t)(\Delta t)^2$$

(2.9.2.10)

where \(\delta_1^1\) are the coordinates after a (normal MD) integration step \(\Delta t\), disregarding all constraints, and \(\delta \bar{R}_1\) are the corrections to be made as a result of the constraints. Using (2.9.2.6) the constraint condition becomes

$$\delta \bar{r}_1 = -m_1^{-1} (\Delta t)^2 \sum_{k=1}^{N_c} \lambda_k \delta_{1k} \bar{r}_1^k (\bar{r}_1^1, \ldots, \bar{r}_1^{N_{at}})$$

(2.9.2.11)

If the constraints are defined in terms of distance constraints, as in (2.9.2.2), \(\delta \bar{r}_1\) becomes

$$\delta \bar{r}_1 = -2m_1^{-1} (\Delta t)^2 \sum_{(k_1, k_2)}^{N_c} \lambda_k \delta_{1j} \bar{r}_1^j (\bar{r}_1^1, \ldots, \bar{r}_1^{N_{at}})$$

(2.9.2.12)

where the summation extends only over constraints involving atom 1. This implies that corrections due to the distance constraint between atoms 1 and j must be applied in the direction of vector \(\bar{r}_1^j\). Corrections to \(\delta \bar{r}_1^j\) and \(\bar{r}_1^j\) are in opposite direction and weighted by the inverse mass of atoms 1 and j.

Consider for simplicity the two atom case shown in Fig. 2.9.2.1. There is one multiplier \(\lambda\). The correction for atom 1 is

$$\delta \bar{r}_1 = -2(\Delta t)^2 m_1^{-1} \lambda \bar{r}_{12} = +2m_1^{-1} \bar{r}_{12}$$

(2.9.2.13)

and for atom 2
\[ \delta \mathbf{r}_{12} = \frac{g}{2} (\Delta t)^2 \delta \mathbf{r}_{12} + \mathbf{m}_{12} \mathbf{r}_{12} \mathbf{r}_{12} = -g \mathbf{m}_{12} \mathbf{r}_{12} \]  

Here, \( g \) is an unknown parameter to be solved by applying the constraint condition \((2.9.2.2)\) to \( \mathbf{r}_{12} (t + \Delta t) \) of \((2.9.2.9)\):

\[ ([\mathbf{r}_{12}^2 + g \mathbf{m}_{12}^{-1} \mathbf{r}_{12}^2 = (\mathbf{r}_{12}^2 + g \mathbf{m}_{12}^{-1} \mathbf{r}_{12}^2)]^2 + \mathbf{c}_{12}^2 = 0 \]  

or

\[ g^2 (\mathbf{m}_{12}^{-1} + \mathbf{m}_{12}^{-1}) \mathbf{r}_{12}^2 + 2g (\mathbf{m}_{12}^{-1} + \mathbf{m}_{12}^{-1}) \mathbf{r}_{12}^2 + \mathbf{r}_{12}^2 + \mathbf{c}_{12}^2 = 0 \]  

This is a quadratic equation for \( g \), which can be solved exactly.

---

**Figure 2.9.2.1**
In the general case there are \( N_c \) equations of type (2.9.2.16), with \( N_c \)
unknown parameters \( g_i \), each equation involving several \( g_i \)'s. Thus at each MD
step a set of \( N_o \) quadratic equations is to be solved. This is done by the
procedure or subroutine SHAKE in the following way. Each constraint
equation of type (2.9.2.16) is solved to first order (neglecting the \( g^2 \)
term), treating all \( N_o \) constraints in succession, and iterating this
procedure until all constraints are satisfied to within a specified
geometric tolerance (TOL).

The application of SHAKE will be denoted by

\[
\text{SHAKE} \left( \dot{\mathbf{r}}(t), \ddot{\mathbf{r}}(t + \Delta t), \dddot{\mathbf{r}}(t + \Delta t) \right)
\]

(2.9.2.17)

This means that the positions \( \dddot{\mathbf{r}}(t + \Delta t) \) that result from the non-constraint
time step, will be reset to give the constrained positions \( \dddot{\mathbf{r}}(t + \Delta t) \). The
direction of the correction vectors:

\[
\delta \mathbf{r} = \dddot{\mathbf{r}}(t + \Delta t) - \dddot{\mathbf{r}}(t + \Delta t)
\]

(2.9.2.16)

is determined by the reference positions \( \dddot{\mathbf{r}}(t) \); that is, for each individual
distance constraint involving atoms \( i \) and \( j \), the correction vector is
parallel to the vector \( \delta \mathbf{r}_{ij}(t) \) in the reference configuration.

When using SHAKE in combination with the GROMOS force field (2.1.1) the
general tolerance (TOL) should be chosen such that the noise in the
simulation due to SHAKE is much smaller than that due to other sources, like
the application of a non-bonded interaction cut-off radius, etc. When
applying energy minimization (EM), we choose \( \text{TOL} = 10^{-3} \), whereas in MD a
value \( \text{TOL} = 10^{-4} \) is used, based on the observation that MD is more sensitive
to the accumulation of errors when moving through configuration space.
2.9.2.1 Constrained positions

When a given molecular configuration \( \mathbf{r}_0 \) does not satisfy a set of constraints, SHAKE can be used to obtain a constrained configuration \( \mathbf{r} \), viz. that satisfies the constraints. This is done by using \( \mathbf{r}_0 \) as reference positions when using SHAKE with \( \mathbf{r}' = \mathbf{r}_0 \) as initial configuration:

\[
\text{SHAKE} \quad (\mathbf{r}_0, \mathbf{r}', \mathbf{r})
\]  (2.9.2.1.1)

2.9.2.2 Constrained velocities

Since the forces \( \mathbf{F} \) from (2.9.2.5) will generally contain components along the constraint directions, this will also be true for the velocities obtained from (2.9.2.7). When applying constraints these components have to be removed, that is, the velocities must be shaken. In the leap-frog algorithm the constrained velocities are simply determined by inverting (2.9.2.8) and using constrained positions at times \( t \) and \( t + \Delta t \):

\[
\mathbf{v}_i(t + \Delta t/2) = \left[ \mathbf{r}_i(t + \Delta t) - \mathbf{r}_i(t) \right] / \Delta t
\]  (2.9.2.2.1)

If unconstrained positions \( \mathbf{r}(t) \) and velocities \( \mathbf{v}(t) \) are given at the same time \( t \), the positions are made to satisfy the constraints by applying SHAKE as specified in Chapter 2.9.2.1. Using constrained positions \( \mathbf{r}(t) \) the velocities are constrained in three steps:

a. Compute

\[
\mathbf{r}'(t + \Delta t) = \mathbf{r}(t) + \mathbf{v}(t) \Delta t
\]  (2.9.2.2.2)
b. Perform

\[ \text{SHAKE}(\ddot{\mathbf{r}}(t), \dot{\mathbf{r}}(t + \Delta t), \mathbf{r}(t + \Delta t)) \]  \hspace{1cm} (2.9.2.2.3)

\[ \dot{\mathbf{v}}(t) = \frac{[\ddot{\mathbf{r}}(t + \Delta t) - \ddot{\mathbf{r}}(t)]}{\Delta t} \]  \hspace{1cm} (2.9.2.2.4)

2.9.2.3 Constrained forces

From (2.9.2.10) and (2.9.2.18) it is clear that the constraint forces \( \ddot{\mathbf{q}}_1(t) \) can be derived by saving the unconstrained configuration \( \ddot{\mathbf{r}}_1(t) \) before application of SHAKE and using

\[ \ddot{\mathbf{q}}_1(t) = \left[ \mathbf{r}_1(t + \Delta t) - \mathbf{r}_1(t) \right] / (\Delta t)^2 \]  \hspace{1cm} (2.9.2.3.1)

We note that the constrained forces, that is, the forces that contain no components along the constraints, are given by \( \ddot{\mathbf{r}}_1 + \ddot{\mathbf{q}}_1 \), since the procedure SHAKE yields a \( \ddot{\mathbf{q}}_1 \) which compensates the components of \( \ddot{\mathbf{r}}_1 \) along the directions of the constraints.
2.9.3 Bond-length constraints in solute

The "solute" part of a molecular topology contains two lists of covalent bonds, one of bonds involving hydrogen atoms, and one involving the other bonds. The first list contains NBONH bonds involving hydrogen atoms. As discussed in Chapter 2.3, three items are stored: IBH, JBH[1..NBONH] are the atom sequence numbers of the atoms forming bond 1-2 as a function of the bond sequence number n, and ICBH[1..NBONH] is the bond-type code, denoting the harmonic force constant \( K_n \) and ideal bond length \( b_n \), as a function of the bond sequence number n. These quantities are stored in CB[1..NBTY] and BO[1..NBTY] as a function of the bond-type code (ICBH or ICB), see Table 2.3.1. The list for the bonds involving no hydrogen atoms contains corresponding items denoted by IB, JB, ICB[1..NBON].

The two lists of covalent bonds can be used to specify the constraints that are imposed on the molecular system, by the subroutine SHAKE. There are three alternatives, which are listed in Table 2.9.3.1. The switch NTC controls which lists of atom pairs are to be used for constraining by subroutine SHAKE. The switch NTF is used to skip the bond interaction terms in the interaction function (2.1.1) when the bond-lengths are constrained. The last column contains the longest time step \( \Delta t \) that may be used in combination with the use of constraints.
### Table 2.9.3.1

**Application of solute bond-length constraints**

<table>
<thead>
<tr>
<th>constraints (SHAKE)</th>
<th>forces (FORCE)</th>
<th>time step</th>
</tr>
</thead>
<tbody>
<tr>
<td>NTC constraint list constraints</td>
<td>NTF forces</td>
<td>$\Delta t$ (fs)</td>
</tr>
<tr>
<td>1 - none</td>
<td>1 all</td>
<td>0.5</td>
</tr>
<tr>
<td>2 IBH-JBH bonds (H's)</td>
<td>2 no bond (H's)</td>
<td>1.0</td>
</tr>
<tr>
<td>3 plus IB-JB bonds (all)</td>
<td>3 no bond (all)</td>
<td>2.0</td>
</tr>
</tbody>
</table>

When applying SHAKE to solute constraints, it is called with NTCC = 0, that is, the constraint lengths are taken from B0[1..NBTY] using the bond-type codes selected by ICBH[1..NBONH] (NTC = 2,3) and by ICB[1..NBON] (NTC = 3).

In older versions of GROMOS it was possible to specify a constraint list (ICOG, JCOG, CONP[1..]) independently of the lists of bond-lengths in the molecular topology. This option has not been maintained.

In a MD simulation at a given temperature $T$, the total kinetic energy of the solute molecules $E_{\text{kin}}$ (solute) will depend on the number of degrees of freedom of the solutes

$$E_{\text{kin}}(\text{solute}) = \sum_{i=1}^{N_{\text{sol}}} \frac{1}{2} m_i \frac{\mathbf{v}_i}{1} = \frac{1}{2} N_{\text{df}}(\text{solute}) kT \quad (2.9.3.1)$$

where $k$ is Boltzmann's constant ($k = 8.31441 \times 10^{-3}$ kJ mol$^{-1}$K$^{-1}$) and the number of degrees of freedom of the solutes is denoted by
Here the number of solute atoms is \( N_{\text{at}} \) (solute) and the number of solute constraints is \( N_{\text{c}} \) (solute). The latter is dependent on the value of switch NTC. In a MD simulation the size of the solute kinetic energy \( E_{\text{kin}} \) (solute) and the size and direction of the atomic velocities \( \vec{v}_i \) will depend on the value of NTC, that is, whether solute constraints are applied or not. This means that if the value of NTC is changed in the course of a MD simulation the velocities have to be rescaled in order to let them correspond to the unchanged temperature \( T \).

For example, a MD job with NTC=3 has produced a final configuration with velocities \( \vec{v}_i \) (NTC=3). If one would continue the simulation with NTC=1 so, if one would assume \( \vec{v}_i \) (NTC=1) = \( \vec{v}_i \) (NTC=3), the temperature \( T \) (NTC=1) would become approximately \( 2/3 \) \( T \) (NTC=3). This can be understood by taking (2.9.3.1) for NTC=1 and NTC=3 and using the equality of the velocities:

\[
\frac{N_{\text{df}} \text{(solute; NTC=3)}}{N_{\text{df}} \text{(solute; NTC=1)}} = \frac{T \text{(NTC=1)}}{T \text{(NTC=3)}}
\]

(2.9.3.3)

Since in a linear molecule the number of covalent bonds is approximately equal to the number of atoms, one has \( T \) (NTC=1) = \( 2/3 \) \( T \) (NTC=3).

In order to avoid this effect, the velocities should be rescaled upon changing NTC from NTC to NTC' by a factor (see input of program PROMID):

\[
\text{abs(HEAT)} = \text{abs}(\sqrt{\frac{N_{\text{df}} \text{(solute; NTC')}}{N_{\text{df}} \text{(solute; NTC)}}})^{1/2}
\]

(2.9.3.4)

The input parameter HEAT is taken negative in order to select only the solute velocities for rescaling, leaving the solvent velocities unchanged.
In the example given above, one would have HEAT = $-\sqrt{3}/2$ for the change from NTC=3 (all bond lengths constrained) to NTC=1 (no constraints), and HEAT = $-\sqrt{2}/3$ for the backwards change from NTC=1 to NTC=3. When changing NTC (and dt) it is advised to use NTCM=1 (Chapter 2.10.2) and INIT=1 (see input of program PROMD).

2.9.4 Bond-length and bond-angle constraints in solvent

In Chapter 3.3.1 it is discussed that solvent molecules of which the topological properties are contained in the "solvent" part of a molecular topology file, are subject to a number of restrictions. One of these is that a solvent molecule is assumed to be rigid. Its internal structure is maintained by the application of distance constraint forces between its atoms using the subroutine SHAKE. At each simulation step SHAKE is called (irrespective of the value of NTC) with NTCC = 1, that is, the square of the constraint lengths are taken from CONS[1..NCONS] and the atom sequence numbers of the atoms forming the distance constraints are taken from ICONS, JCONS[1..NCONS], where NCONS is the number of distance constraints within a solvent molecule. These data are kept in the solvent part of the molecular topology.

For example, the simple point charge (SPC) model of liquid water requires three distance constraints (NCONS = 3). The oxygen atom has atom sequence number one, the hydrogens two and three. The constraint data are (in nanometers):

\[
\begin{align*}
\text{ICONS}[1] &= 1 & \text{JCONS}[1] &= 2 & \text{CONS}[1] &= 0.1\,\text{nm}^2 \\
\text{ICONS}[2] &= 1 & \text{JCONS}[2] &= 3 & \text{CONS}[2] &= 0.1\,\text{nm}^2 \\
\text{ICONS}[3] &= 2 & \text{JCONS}[3] &= 3 & \text{CONS}[3] &= 0.1633\,\text{nm}^2
\end{align*}
\]
2.10 Treatment of boundaries

2.10.1 Introduction

When simulating a system of finite size, some thought must be given on the way the boundary of the system will be treated. The simplest choice is the vacuum boundary condition which is discussed in Chapter 2.10.2. When simulating a liquid or a solution rather than a molecule in the gas phase, it is common practice to minimize edge or wall effects by the application of periodic boundary conditions, which are discussed in Chapter 2.10.3. For a macromolecule in solution the application of periodic boundary conditions may be too expensive. In that case the number of atoms that are simulated, can be limited by simulating only part of the system. Edge or wall effects can be minimized by restraining the motion of the atoms in the outer shell or wall region of the system. This is discussed in Chapter 2.10.4.

2.10.2 System in vacuo

Simulating a molecular system in vacuo, that is, without any wall or boundary, corresponds to the gas phase at pressure zero. When the vacuum boundary is used for a molecule in solution, properties of atoms near or at the surface of the system will be distorted [MD 82.9]. The vacuum boundary condition may also distort the shape of a (non-spherical) molecule, since it generally tends to minimize the surface area. Besides, the shielding effect of a solvent with high dielectric permittivity, like water, on the electric interaction between charges or dipoles in a molecule is missed in vacuo. Therefore, simulation of a charged extended molecule like DNA in vacuo is a
precareous undertaking. The best results in vacuo are obtained for relatively large globular macromolecules.

The vacuum boundary condition is selected using the switch NTB = 0.

When the molecular system contains groups of atoms with a total net charge not equal to zero, it is advised to use the GROMOS 37D4 force field in which charged groups are neutralized in order to compensate the non-shielding vacuum around the molecule. (Chapter 1.5; 2.2; 4.1).

When simulating a system in vacuo, the translational momentum and the angular momentum are conserved quantities. Therefore, it is common practice to stop the translational motion of the center of mass and the rotational motion around the center of mass at the start of such a simulation. This is done by using NTCM = 1 at the start of a simulation. In vacuo these motions will remain absent (zero), so in continuation jobs the center of mass motion need not be stopped again (NTCM = 0 and NSCM = number of steps after which center of mass motion is stopped again = 100000 = infinite).

The temperature $T$ of a system is calculated using the relation

$$\sum_{i=1}^{N_{at}} \frac{1}{2} m_i v_i^2 = \frac{1}{2} N_{df} k T \ \ \ \ (2.10.2.1)$$

where $k$ is Boltzmann's constant ($k = 8.31441 \times 10^{-2} \text{kJ mol}^{-1} \text{K}^{-1}$) and the number of degrees of freedom in the system is denoted by

$$N_{df} = 3 N_{at} - N_c + \text{NDFMIN} \ \ \ \ (2.10.2.2)$$

Here the number of atoms is $N_{at}$ and the number of constraints is $N_c$. The value of NDFMIN will depend on the boundary condition and on whether the center of mass motion has been stopped.
When the overall translational and rotational motion has been stopped
and the system is in vacuo, six degrees of freedom have to be subtracted
from the total in (2.10.2.2), so NdFMN = 6, in order to obtain the correct
kinetic energy per degree of freedom.

2.10.3 Periodic boundary conditions

2.10.3.1 Introduction

The classical way to minimize edge effects in a finite system is to use
periodic boundary conditions. The atoms of the system that is to be
simulated are put into a cubic, or more general into any periodically space
filling shaped box, which is surrounded by 26 identical translated images of
itself (Fig. 2.10.3.1.1).

![Periodic Boundary Conditions](image)

Figure 2.10.3.1.1
When calculating the forces on the black atom in the central box, all interactions with atoms in the central box or images in the surrounding boxes that lie within the spherical cut-off radius $R_c$ are taken into account. When an atom leaves the central box at one side, it enters it with identical velocity at the opposite side at the translated image position.

Application of periodic boundary conditions means that in fact a crystal is simulated. For a molecule in solution the periodicity is an artifact of the computation, so the effects of periodicity on the forces on the atoms should not be significant. This means that an atom should not simultaneously interact with another atom and its image. Therefore, in GROMOS only the interaction between nearest images is evaluated.

Therefore, consequently, the box size $R_{\text{box}}$ must exceed twice the cut-off radius $R_c$:

$$R_{\text{box}} > 2 R_c$$

(2.10.3.1.1)

This condition can be met by choosing the system large enough; e.g. in a crystal the computational box may contain more than one unit cell. Possible distorting effects of the periodic boundary condition may be traced by simulating systems of different size.

Applying periodic boundary conditions implies that when an atom leaves the central box through one of its walls, it enters at the opposite image position with the same velocity. However, in GROMOS this periodic translation is not performed for single atoms, but for all atoms of a charge group. Subroutine SHIAQ translates solute charge group atoms and solvent molecules, applying periodic boundary conditions, such that the first atom of a solute charge group or of a solvent molecule lies within the central periodic box. The position of this central box is specified by the $x^n$, $y^n$, $z^n$. 
z-coordinates XMIN[1..3], with respect to which the central box occupies the positive quadrant. The x-, y-, z-edges of the periodic box are denoted by BOX[1..3].

For an arbitrary molecular configuration the atoms of a charge group may lie far apart in the central box, close to opposite walls, while their nearest images are close to each other. In that case the atoms of a charge group must first be gathered before SHIAG can be applied. Subroutine CONCG gathers the atoms forming a solute charge group by applying periodic boundary conditions, such that the atoms of a charge group lie within $R_{box}/2$ of each other. It is always assumed that the atoms of a solvent molecule lie within $R_{box}/2$ of each other, viz. that the solvent atom coordinates are generated without mixing different periodic images in one solvent molecule.

When a solute molecule consists of a chain of covalently bound atoms, this chain may be cut into different segments by the periodic boundaries; following the chain one may leave the central box through one wall and enter it at the opposite wall at the image position. This means that when the various contributions to the potential energy in (2.1.1) are computed, nearest images of atoms involved in bonds, bond angles, dihedrals, etc. have to be used. This is done in the corresponding subroutines (COBOND, ANGLE, DIHANG, SHAKE, DISRE, RESTX, the non-bonded interaction ones, etc.).

In some situations it is required that the connectivity of the covalently bound chain of a solute molecule, which is broken by the periodic boundary condition, is restored. For example, when a picture of the solute conformation is to be made, or when average atomic positions are to be determined, or when the center of mass of a solute molecule is to be determined. Subroutine CONAT gathers the atoms of a solute molecule by applying periodic boundary conditions, such that each atom lies within
of the solute can be restored.

For further technical details on the application of periodic boundary conditions we refer to Chapter 6.8. Periodic boundary conditions are switched on when NTB ≠ 0. If NTB > 0, the periodic box is rectangular (Chapter 2.10.3.2) or monoclinic (Chapter 2.10.3.3) depending on the value of the angle \( \beta \) (BETA) between the x- and z-axes (edges) of the periodic box. If NTB < 0, the periodic box is a truncated octahedron (Chapter 2.10.3.4).

When simulating an infinite system by applying periodic boundary conditions, it is advised to use the GROMOS 37C4 force field, since no distorting vacuum boundary effects are present when the cut-off radius is chosen sufficiently large.

When simulating a system using periodic boundary conditions, the translational momentum is a conserved quantity, but the angular momentum of the system is not conserved. As in vacuo, it is common practice to stop the translational motion of the center of mass of the system and also the rotational motion around the center of mass at the start of a simulation. However, the rotational degree of freedom will pick up energy when the system evolves, and the translational energy will remain zero. Therefore, when the temperature of the system is calculated using \( (2.10.2.1-2) \) three degrees of freedom have to be subtracted from the total in \( (2.10.2.2) \), so NDPMIN = 3, in order to obtain the correct kinetic energy per degree of freedom.

The application of non-bonded neighbour search techniques in periodic systems is discussed in [MD 84.3].

2.10.3.2 Periodic rectangular box
The simplest case of a periodic box is a rectangular box \((\text{NTB} \geq 0, \text{BETA} = 90.0)\). The lengths of the edges in the \(x\)-, \(y\)-, and \(z\)-directions are denoted by \(a\), \(b\) and \(c\) \((\text{BOX}[1..3])\). The atom or charge group \(i\) can be kept in the computational box that lies in the positive quadrant with respect to an origin at \(\hat{r}_0\), by applying:

\[
\begin{align*}
    x_i &= x_i - \text{NINT}\left(\frac{x_i - x_0 - a/2}{a}\right)a \\
    y_i &= y_i - \text{NINT}\left(\frac{y_i - y_0 - b/2}{b}\right)b \\
    z_i &= z_i - \text{NINT}\left(\frac{z_i - z_0 - c/2}{c}\right)c
\end{align*}
\]  

(2.10.3.2.1)

where the function \(\text{NINT}(x)\) delivers the integer number that is nearest to \(x\).

For two atoms or charge groups \(i\) and \(j\) the vector

\[
\hat{r}_{ij} = \hat{r}_i - \hat{r}_j
\]

(2.10.3.2.2)

connecting nearest images is obtained from

\[
\begin{align*}
    x_{ij} &= x_{ij} - \text{NINT}\left(\frac{x_{ij}}{a}\right)a \\
    y_{ij} &= y_{ij} - \text{NINT}\left(\frac{y_{ij}}{b}\right)b \\
    z_{ij} &= z_{ij} - \text{NINT}\left(\frac{z_{ij}}{c}\right)c
\end{align*}
\]  

(2.10.3.2.3)

For a rectangular computational box the requirement \((2.10.3.1.1)\) becomes

\[
R_0 < 1/2 \text{ MIN}(a, b, c)
\]  

(2.10.3.2.4)
that is, the cutoff radius (RCUTP or RCUTL) must be smaller than half the smallest edge of the box. The function MIN(x, y, ...) is delivered with the smallest of its arguments x, y, ...

2.10.3.3 Periodic monoclinic box

When simulating a molecular system in the crystalline state, the periodicity is determined by the crystal form. The periodicity requirements (2.10.3.2.1, 3) now apply to contravariant coordinates, which need not be identical with the Cartesian ones. In the monoclinic case (NTB > 0, BETA ≠ 90.0), the crystal axes $\hat{a}$, $\hat{b}$ and $\hat{c}$ are chosen as follows. We assume $\hat{b}$ to be orthogonal to $\hat{a}$ and $\hat{c}$, and $\hat{a} \cdot \hat{c} = \cos \beta$ ($\beta = $ BETA, 2nd setting). The Cartesian x axis is chosen along the crystal $\hat{a}$ axis, the Cartesian y axis along $\hat{b}$, and the Cartesian z axis along $\hat{a} \times \hat{b}$. Then the transformation from Cartesian components (x, y, z) to oblique contravariant components (x', y', z') reads (Fig. 2.10.3.3.1):

![Figure 2.10.3.3.1](image-url)
\[ x' = x - z \cot \beta \]
\[ y' = y \]
\[ z' = z / \sin \beta \]  \hspace{1cm} (2.10.3.3.1)

and the inverse transformation is

\[ x = x' + z' \cos \beta \]
\[ y = y' \]
\[ z = z' \sin \beta \]  \hspace{1cm} (2.10.3.3.2)

So, the square of the length of a vector \( \hat{r} \) is given by

\[ r^2 = x^2 + y^2 + z^2 \]  \hspace{1cm} (2.10.3.3.3)
\[ = (x')^2 + (y')^2 + (z')^2 + 2x'z' \cos \beta \]  \hspace{1cm} (2.10.3.3.4)

The transformations (2.10.3.3.1-2) can be performed using subroutine TRACO.

The atom or charge group \( i \) can be kept in the monoclinic computational box that lies in the positive quadrant with respect to an origin at \( \hat{r}_0 \), by evaluating \( \hat{r}_{i0} = \hat{r}_i + \hat{r}_0 \), applying the transformation (2.10.3.3.1) to \( \hat{r}_{i0} \) and \( \hat{r}_0 \), performing the periodicity check (2.10.3.2.1) and transforming the resulting vector back to Cartesian components through (2.10.3.3.2). When calculating the vector \( \hat{r}_{ij} \) connecting nearest images of atoms or charge groups \( i \) and \( j \), one evaluates \( \hat{r}_{ij} \) from (2.10.3.2.2), applies the transformation (2.10.3.3.1), applies the periodicity check (2.10.3.2.3), and transforms back to Cartesian components through (2.10.3.3.2). If only \( r_{ij}^2 \) is required, the last step is replaced by calculating \( r_{ij}^2 \) through (2.10.3.3.4).
For a monoclinic computational box the requirement (2.10.3.1.1) becomes

\[ R_0 < \frac{1}{2} \text{MIN}(b,(a^2 + c^2 - ac \cos \beta)^{1/2},(a^2 + c^2 + ac \cos \beta)^{1/2}) \] (2.10.3.3.5)

that is, the cut-off radius (RCUTP or RCUTL) must be smaller than half the smallest distance between an atom or charge group and its periodic image.

2.10.3.4 Periodic truncated octahedron

When simulating a spherical solute use of a more spherically shaped computational box in stead of a rectangular one may considerably reduce the number of solvent molecules that is needed to fill the remaining empty space in the box. A more spherically shaped space filling periodic box is a truncated octahedron, shown in Fig. 2.10.3.4.1. The distance between the square planes is \( a \), and between the six-sided planes it is \( a\sqrt{3}/2 \).

Figure 2.10.3.4.1
Truncated octahedron periodic boundary conditions are applied when NTB < 0 (a = BOX[1] - BOX[2] - BOX[3], BETA = 90.0). The atom or charge group i can be kept in the computational box that lies in the positive quadrant with respect to an origin at \( \mathbf{r}_0 \), by applying the rectangular periodicity check (2.10.3.2.1) with \( a = b = c \) followed by

\[
\begin{align*}
\text{if } & |x_i - x_0 - a/2| + |y_i - y_0 - a/2| + |z_i - z_0 - a/2| > 3a/4 \\
\text{then } & x_i = x_i - \text{SIGN}(a/2, x_i) \\
& y_i = y_i - \text{SIGN}(a/2, y_i) \\
& z_i = z_i - \text{SIGN}(a/2, z_i) \\
\end{align*}
\]

(2.10.3.4.1)

where the function \( \text{SIGN}(x,y) \) delivers \( x \) multiplied by the sign of \( y \). The vector \( \mathbf{r}_{ij} \) connecting nearest images of atoms or charge groups i and j is obtained by applying the rectangular periodicity check (2.10.3.2.3) with \( a = b = c \) followed by

\[
\begin{align*}
\text{if } & |x_{ij} - a/2| + |y_{ij} - a/2| + |z_{ij} - a/2| > 3a/4 \\
\text{then } & x_{ij} = x_{ij} - \text{SIGN}(a/2, x_{ij}) \\
& y_{ij} = y_{ij} - \text{SIGN}(a/2, y_{ij}) \\
& z_{ij} = z_{ij} - \text{SIGN}(a/2, z_{ij}) \\
\end{align*}
\]

(2.10.3.4.2)

If only \( r_{ij}^2 \) is required, the last step (after applying (2.10.3.2.3)) is replaced by calculating:

\[
\begin{align*}
r_{ij}^2 &= x_{ij}^2 + y_{ij}^2 + z_{ij}^2 + a.\text{MIN}(0, 3a/4 - |x_{ij}| - |y_{ij}| - |z_{ij}|) \\
\end{align*}
\]

(2.10.3.4.3)

For a truncated octahedron the requirement (2.10.3.1.1) becomes
\[ R_c < \frac{a\sqrt{3}l}{4} \] (2.10.3.4.4)

that is, the cut-off radius (R\text{CUTP} or R\text{CUTL}) must be smaller than half the distance between opposite planes that are defining the truncated octahedron.

2.10.4 Wall region of restrained atoms

When simulating crystals or solutions of large molecules, the application of periodic boundary conditions may become very expensive. In that case the number of atoms in the simulation can be limited by simulating only part of the molecular system. For example, a spherical region around a specific atom or point in the system is retained, while all atoms lying beyond a radius \( R_2 \) are removed from the system. Edge effects, due to the presence of vacuum beyond \( R_2 \), can be minimized by restraining the motion of the atoms in the outer shell, viz. between radii \( R_1 \) and \( R_2 \). This shell is called the (extended) wall region (see Fig. 2.10.4.1). The atoms in this wall region can be kept near given stationary reference positions by the technique of (harmonic) position restraining, which has been discussed in Chapter 2.8.2. Atoms in the inner region (within \( R_1 \)) are simulated without any restraints.

When applying the extended wall region boundary conditions, the molecular topology (Chapter 3.3) can be reduced to one containing only atoms that lie with a distance \( R_2 \) (R\text{CUT2}) from a given point, by applying program PROCOMT. The reduced system is not restricted to be of spherical shape. During a simulation the vacuum boundary condition is involved (NTB = 0) and position restraining is applied (subr. RESTX) to the atoms of a given list. Again, the restrained region is not restricted to be a spherical shell.
In order to avoid distorting effects of the vacuum beyond $R_2$ on the atomic motion within $R_1$, one should choose these radii such that

$$R_o < R_2 - R_1 \quad (2.10.4.1)$$

that is, the non-bonded cut-off radius ($RCUTP$) should be smaller than the thickness of the shell of restrained atoms.
2.11 Energy minimization

2.11.1 Introduction

Energy minimization (EM) with an empirical energy function like (2.1.1) is a widely used tool for obtaining low-energy configurations of a molecular system. Various function minimization methods can be used, which can be classified as follows:

1. Direct search methods, requiring only function values. They converge slowly and are therefore not considered here.

2. Gradient methods, requiring function and derivative values. These methods fall into three subclasses:
   a. The steepest descents method (SD) performs well far from a minimum, but converges slowly near a minimum, or when searching in a long, thin, curving valley. It is a robust method, which is easy to implement.
   b. The conjugate gradients method (CG) (R. Fletcher and C.M. Reeves, Comput. J., 7 (1964) 149), which searches along directions corresponding to the local quadratic approximation to the function, usually converges superlinearly. Because it is the most rapidly converging minimizer that does not require manipulation and storage of matrices of dimension equal to the number of degrees of freedom, it appears to be most appropriate for very large systems, like macromolecules.
   c. The variable metric or quasi-Newton methods, which use various approximations to the inverse of the Hessian matrix (matrix of second partial derivatives), are also quadratically convergent, but they
require storage space for the inverse Hessian and time for its manipulation. Hence, they are less suited for application to large systems.

3. Second-order methods, requiring function, derivative and Hessian matrix. These methods are not well suited for application to large systems for the same reason as mentioned in 2c.

For references to various methods we refer to [MD 80.3]. The method of steepest descents is discussed in Chapter 2.11.2, the conjugent gradients technique in Chapter 2.11.3. In Chapters 2.11.4 and 2.11.5 it is described how the SHAKE-method for constraining bond length and/or bond angles can be incorporated in the SD and CG energy minimization algorithms.

SD or CG energy minimization is performed by subroutine RUNEM. The code closely follows the computational scheme as it is described in [MD80.3].

When applying EM algorithms one searches for a minimum energy configuration of a system by moving (approximately) along the gradient of the potential energy through configuration space:

\[
\Delta r_i = -\nabla_i V(r_1, r_2, \ldots, r_{N_{at}}) \quad (2.11.1.1)
\]

Since in this way one basically moves only downhill over the energy hypersurface, EM yields only a local minimum energy configuration, which is generally not far from the initial one. Using formulae like (2.11.1.1) crossing of energy barriers is almost impossible. A more efficient way to find low energy configurations is to apply molecular dynamics (MD). The available kinetic energy may be used to pass over energy barriers which are
not much higher than kT (k = Boltzmann's constant and T = absolute
temperature). It has been shown [MD84.4] that MD at elevated temperatures
can be used to generate a variety of configurations. Therefore, MD searches
a larger part of configuration space for an energy minimum and generally
ends up in a lower energy minimum than ordinary an energy minimizer does
[MD87.1]. However, the application of MD starting from a highly strained,
very high potential energy configuration is not recommended, since the
immediate conversion of potential energy into kinetic energy will raise the
temperature to unacceptably high values. In that case, one should first
perform a number of EM steps in order to reduce the high potential energy of
the system. When the high potential energy is due to close nonbonded
contacts or stretched bond lengths or bent bond angles in a molecular
system, ten to fifty EM steps generally suffice to reduce the potential
energy to values which are normal at room temperature.

In correspondence with [MD80.3] the following notation will be used in
this Chapter. A molecular configuration is represented by the 3N_{\text{at}}-
dimensional vector |x\rangle composed of the 3N_{\text{at}} Cartesian components of the
atomic position vectors \( r_1, r_2, \ldots, r_{N_{\text{at}}}. \) The configuration at the n-th
minimization step is denoted by |x_n\rangle. The scalar product of two vectors |f\rangle
and |g\rangle is defined by

\[ <f|g> = \sum_{i=1}^{N_{\text{at}}} f_i^* g_i \]

where \( f_i \) and \( g_i \) are the 3-dimensional Cartesian vectors belonging to atom \( i. \)
2.11.2 Steepest descents minimization

Energy minimization by the steepest descents (SD) algorithm is simple. The computational scheme for the \((n+1)\)-th minimization step reads:

1. Derive the forces \(\mathbf{f}_n = \mathbf{f}(\mathbf{x}_n)\) from the interaction function \(V(\mathbf{x})\) (see eq. (2.1.1)) using the configuration \(\mathbf{x}_n\).

2. Compute the next configuration from

\[
\mathbf{x}_{n+1} = \mathbf{x}_n + \Delta \mathbf{x} \langle \mathbf{f}_n | \mathbf{f}_n \rangle^{-1/2} \mathbf{f}_n
\]

(2.11.2.1)

where the step-size is denoted by \(\Delta \mathbf{x}\).

SD energy minimization is selected in subroutine RUNEM by setting switch NTEM = 1. The initial step-size \(\Delta \mathbf{x}\) is to be specified: DX0. As long as the potential energy decreases, the step-size is increased by 20\%. If the energy increases, it is halved. The growth of the step-size can be limited by specifying a maximum value: DXM. The energy minimization is terminated when the number of EM steps reaches the value NSTLIM or when the potential energy change between two subsequent steps is less than the value DELE. The final configuration is stored (see program PROEM).

At every NTPR-th minimization step the following items are printed by subroutine RUNEM:

- **STEP**: minimization step number \(n\)
- **STEP-SIZE**: actual value of \(\Delta \mathbf{x}\)
- **RMS-F-FREE**: \(\langle \mathbf{f}_n | \mathbf{f}_n \rangle^{1/2}\)
- **E-POT-TOT**: total potential energy of the molecular system
$E_{\text{BOND-H}}$ - total energy of bond-stretching terms involving H-atoms (solute)

$E_{\text{BOND}}$ - total energy of bond-stretching terms involving non-hydrogen atoms (solute)

$E_{\text{ANGLE-H}}$ - total energy of bond angle bending terms involving H-atoms (solute)

$E_{\text{IM-DIH-H}}$ - total energy of improper (harmonic) dihedral angle terms involving H-atoms (solute)

$E_{\text{IM-DIH}}$ - total energy of improper (harmonic) dihedral angle terms involving non-hydrogen atoms (solute)

$E_{\text{DIH-H}}$ - total energy of (sinusoidal) dihedral angle terms involving H-atoms (solute)

$E_{\text{DIHEDRAL}}$ - total energy of (sinusoidal) dihedral angle terms involving non-hydrogen atoms (solute)

$E_{\text{EL-G1-G1, ...}, E_{\text{EL-G4-G4}}}$ - total electric energy between atoms belonging to four different groups ($G1, ..., G4$) of which the fourth group contains all solvent molecules; the atom sequence number of the last atom of the first and second group are to be specified in NRE[1] and NRE[2], see subroutine FORCE and program PROEM (0 ≤ NRE[1] < NRE[2] ≤ NRP = number of atoms per solute); the remaining solute atoms belong to group three; the ordering of the energies is the following: G1-G1, G1-G2, G2-G2, G1-G3, G2-G3, G3-G3, G1-G4, G2-G4, G3-G4, G4-G4.

$E_{\text{LJ-G1-G1, ...}, E_{\text{LJ-G4-G4}}}$ - idem, but for the total Lennard-Jones energy.

2.11.3 Conjugate gradients minimization
The conjugent gradient (CG) algorithm can be summarized as follows. It is started by deriving the forces or negative gradients, $|f_0\rangle = f(|x_0\rangle)$, from the interaction function $V(|x\rangle)$ (see eq. (2.1.1)) at the initial positions $|x_0\rangle$, and by taking the first search directions $|p_0\rangle$ along the negative gradients, that is, $|p_0\rangle = -|f_0\rangle$. The computational scheme for the $(n+1)$-th minimization step reads:

1. Find the minimum of the interaction function $V(|x\rangle)$ on the line through $|x_n\rangle$ in the direction $|p_n\rangle$; that is, determine $s_{\min}$ such that

$$V_n(s_{\min}) = V(|x_n\rangle + s_{\min}|p_n\rangle)$$

find its minimum, or equivalently, such that

$$g_n(s_{\min}) = \langle p_n | f(|x_n\rangle + s_{\min}|p_n\rangle) = 0$$

The determination of $s_{\min}$ is done in two stages: the first establishes bounds $a$ and $b$ on $s_{\min}$ ($a < s_{\min} < b$) and the second interpolates its value in the interval $(a,b)$.

a. As a first guess for the bounds, take $a = 0$ and $b = \Delta x \langle p_n | p_n \rangle^{-1/2}$.

The step-size $\Delta x$ should be chosen such that no iterations (number = NITI) are required when establishing the bounds $a$ and $b$. We know

$$|f_n(a)\rangle = |f(|x_n\rangle + a|p_n\rangle)\rangle$$

so $g_n(a)$ can easily be obtained. The same quantities have to be examined at $b$, so we compute
\[ |x_n(b)\rangle = |x_n\rangle + b|p_n\rangle \quad (2.11.3.4) \]

and evaluate \( V_n(b) \) and the forces \( |f_n(b)\rangle \) at \( |x_n(b)\rangle \) from the interaction function, and compute \( g_n(b) \). The values \( a \) and \( b \) are accepted as bounds on \( s_{\text{min}} \) if

\[ g_n(b) < 0 \text{ or } V_n(b) \geq V_n(a) \quad (2.11.3.5) \]

Otherwise \( s_{\text{min}} \) lies beyond \( b \), so \( b \) is too small and so is \( \Delta x \) and the process of looking for bounds on \( s_{\text{min}} \) is to be repeated (repetition count = NITI) with \( a \) taken equal to \( b \) and \( b \) doubled.

b. Use a cubic interpolation formula in order to find \( s_{\text{min}} \) in the interval \((a,b)\):

\[ s_{\text{min}} = b - \frac{[w - z - g_n(b)][b-a][g_n(a) - g_n(b) + 2w]^{-1}}{w} \quad (2.11.3.6) \]

with

\[ w = [z^2 - g_n(a)g_n(b)]^{1/2} \quad (2.11.3.7) \]

and

\[ z = 3[V_n(a) - V_n(b)][b-a]^{-1} - g_n(a) - g_n(b) \quad (2.11.3.8) \]

the new positions are:

\[ |x_{n+1}\rangle = |x_n\rangle + s_{\text{min}}|p_n\rangle \quad (2.11.3.9) \]
2. Derive the new energy $V_{n+1} = V(|x_{n+1}>)$ and the forces $|f_{n+1}> = |f(|x_{n+1}>)$ from the interaction function.

3. Derive the new search directions $|p_{n+1}>$ from

$$|p_{n+1}> = |f_{n+1}> + \beta_n |p_n>$$  \hspace{1cm} (2.11.3.10)

where

$$\beta_n = \frac{<f_{n+1} | f_{n+1}>}{<f_n | f_n>}$$  \hspace{1cm} (2.11.3.11)

We see that each new direction of search is partly be determined by previous search directions. The weight depends on the relative size of the forces at $|x_{n+1}>$ and at $|x_n>$. Although this CG algorithm performs at least two function evaluations ($V(|x>)$) per minimization step, it is generally more efficient than the SD algorithm, which performs one function evaluation per minimization step [MD80.3].

CG energy minimization is selected in subroutine RUNEM by setting switch NTEM = 2. The CG step-size $\Delta x = DX0$ should be carefully chosen. It should be just large enough to avoid iterations (NITI) when establishing bounds a and b on $s_{\text{min}}$ (step 1a), but small enough to allow for efficient convergence near a local minimum. If the energy decreases slowly near a local minimum, one may try to reach it faster by restarting the algorithm with a smaller $\Delta x$. It may be useful to limit the number of steps taken with a given series of search direction vectors; that is, after every NCYC minimization step we take

$$\beta_n = 0$$  \hspace{1cm} (2.11.3.12)
instead of (2.11.3.71), discarding the contribution of previous search
directions to $|p_{n+1}\rangle$ in (2.11.3.10). The energy minimization is terminated
when the number of EM steps reaches the value NSTLIM or when the potential
energy change between two subsequent steps is less than the value DELE. The
final configuration is stored (see program PROEM).

2.11.4 Steepest descents minimization with constraints (SHAKE)

The essential feature of the SHAKE method for conserving constraints is
that after each minimization step, the constraints are satisfied by adding
displacement vectors to the position vectors of the atoms that result from a
nonconstrained minimization step. The added displacement vectors are
determined such that the constraints are satisfied at the final positions
(see Chapter 2.9.2).

SHAKE can be incorporated directly into the steepest descents energy
minimization algorithm in the following way. The initial positions $|x_0\rangle$
must be made to satisfy the constraints as is discussed in Chapter 2.9.2.1.
Shaking of the initial positions is selected by taking INIT = 1 in
subroutine RUNEM. Then, the computational scheme for the (n+1)-th
minimization step reads:

1. Derive the non-constrained forces $|f'_n\rangle = |f'(|x_n\rangle\rangle$ from the interaction
   function $V(|x\rangle)$ (see eq. (2.1.1)), from which the terms acting only along
   the constrained degrees of freedom are excluded (NTF = NTC, see Table
   2.9.3.1).

2. Compute the nonconstrained positions $|x'_{n+1}\rangle$ from

$$
|x'_{n+1}\rangle = |x_n\rangle + \Delta x \langle f'_{n} | f'_{n} \rangle^{-1/2} |f'_{n}\rangle
$$

(2.11.4.1)
where the step-size is denoted by $\Delta x$. The positions $|x_{n+1}'>$ do not, in general, satisfy the constraints, as the forces normally contain components in the constrained directions.

3. The positions are made to satisfy the constraints by performing

$$\text{SHAKE } (|x_n>, |x_{n+1}'>, |x_{n+1}'>)$$

(2.11.4.2)

The number of iterations in SHAKE is printed under NIP1 for the solutes and under NIS1 for the solvent molecules.

The other parameters of the SD algorithm with SHAKE are identical to those discussed in Chapter 2.11.2 for the nonconstrained SD algorithm. An additional feature is that the step-size $\Delta x$ is halved when the number of iterations in SHAKE exceeds 100. The constrained forces $|f_n'>$ can be obtained from solving (2.11.4.1) for $|f_n'>$ and using $|x_{n+1}'>$ instead of $|x_{n+1}'>$:

$$|f_n'> = [|x_{n+1}'> - |x_n'>] / [\Delta x \langle f_n' | f_n' \rangle^{-1/2}]$$

(2.11.4.3)

The quantity $\langle f_n' | f_n' \rangle^{1/2}$ is denoted by RMS-F-CONS, and $\langle f_n' | f_n' \rangle^{1/2}$ by RMS-F-FREE in the print out of subroutine RUNEM.

2.11.5 Conjugate gradients minimization with constraints (SHAKE)

Incorporation of SHAKE into the conjugate gradients algorithm is more complex than in the case of steepest descents, since there only the positions $|x_{n+1}'>$ had to be shaken. Here the search direction, which is composed of the force direction and former search directions, also must be
chosen such that it does not contain components along the constraints.

Hence, \( |f'_{n+1}\rangle \) and \( |p'_{n+1}\rangle \) have to be shaken too.

The initial positions \( |x_0\rangle \) must be made to satisfy the constraints as is discussed in Chapter 2.9.2.1. Shaking of the initial positions is selected by taking INIT = 1 in subroutine RUNEM. The initial nonconstrained forces \( |f'_0\rangle \) can be shaken by the procedure, which was used to remove components along the constraint directions from the velocities in Chapter 2.9.2.2:

a. Compute

\[
|x'_1\rangle = |x_0\rangle + \Delta x' |f'_0\rangle
\]

where \( \Delta x' = \Delta x |f'_0\rangle |f'_0\rangle^{-1/2} \) and \( \Delta x \) is the conjugate gradient step-size.

b. Perform

\[
\text{SHAKE} \ (|x_0\rangle, |x'_1\rangle, |x''_1\rangle)
\]

c. Obtain the constrained or shaken forces \( |f_0\rangle \) from

\[
|f_0\rangle = [|x''_1\rangle - |x_0\rangle] / \Delta x'
\]

The initial search direction is taken along \( |f_0\rangle \), that is, \( |p_0\rangle = |f_0\rangle \).

Then, the computational scheme for the \((n+1)\)-th minimization step reads:

1. Find the minimum of the interaction function \( V(|x\rangle) \) on the line through \( |x_n\rangle \) in the direction \( |p_n\rangle \); that is, determine \( s_{\text{min}} \) such that
\[ V_n(s_{\text{min}}) = V(|x_n\rangle + s_{\text{min}}|p_n\rangle) \]  
(2.11.5.4)

Find its minimum, or such that

\[ g_n(s_{\text{min}}) = \langle p_n|f(|x_n\rangle + s_{\text{min}}|p_n\rangle)\rangle = 0 \]  
(2.11.5.5)

When the component along the constraints have been eliminated from \(|p_n\rangle\) and \(f(|x_n\rangle + s_{\text{min}}|p_n\rangle)\rangle\) the conditions (2.11.5.4) and (2.11.5.5) are not equivalent and \(V_n(s_{\text{min}})\) may have a minimum with \(g_n(s_{\text{min}}) \neq 0\). The determination of \(s_{\text{min}}\) is done in two stages; the first establishes bounds \(a\) and \(b\) on \(s_{\text{min}}\) (\(a < s_{\text{min}} < b\)) and the second interpolates its value in the interval \((a,b)\).

a. As a first guess for the bounds, take \(a = 0\) and \(b = \Delta x < p_n\rangle|p_n\rangle^{-1/2}\).

We know \(V_n(a)\) and the constrained (shaken) forces \(|f_n(a)\rangle\), so \(g_n(a)\) is easily obtained. The same quantities have to be examined at \(b\), so we compute

\[ |x_n(b)\rangle = |x_n\rangle + b|p_n\rangle \]  
(2.11.5.6)

and evaluate \(V_n(b)\) and the forces \(|f_n'(b)\rangle\) at \(|x_n(b)\rangle\) from the interaction function, from which the terms acting only along the constrained degrees of freedom are excluded. The components along the directions of the constraints are removed by the procedure (2.11.5.1-3), that is, by putting

\[ |x_{n+1}(b)\rangle = |x_n(b)\rangle + \Delta x'|f_n'(b)\rangle \]  
(2.11.5.7)

where
\[ \Delta x' = b <p_n | p_n' >^{1/2} / \langle f_n'(b) | f_n(b) >^{1/2}. \]  

(2.11.5.8)

SHAKE \((|x_n(b)>, |x'_{n+1}(b)>, |x''_{n+1}(b)>)\)  

(2.11.5.9)

and

\[ |f_n(b) > = [ | x''_{n+1}(b) > - | x_n(b) >] / \Delta x'. \]  

(2.11.5.10)

The number of iterations in SHAKE is printed under NIF1 for the solutes and under NIS1 for the solvent molecules. With these shaken forces \(g_n(b)\) can be computed. The values \(a\) and \(b\) are accepted as bounds on \(s_{\text{min}}\) if

\[ g_n(b) < 0 \text{ or } V_n(b) \geq V_n(a). \]  

(2.11.5.11)

Otherwise \(s_{\text{min}}\) lies beyond \(b\), so \(b\) is too small (and so is \(\Delta x\)) and the process of looking for bounds on \(s_{\text{min}}\) is to be repeated (repetition count = NITI) with \(a\) taken equal to \(b\) and \(b\) doubled.

However, when the number of iterations in SHAKE exceeds 10, it is likely that SHAKE applied to the next repetition will not converge, since \(b\) is being doubled; in that case \(a\) and \(b\) are accepted as bounds on \(s_{\text{min}}\).

b. Use the cubic interpolation formulae (2.11.3.6-8) in order to find \(s_{\text{min}}\) in the interval \((a,b)\) and find the new positions

\[ |x'_{n+1} > = | x_n > + s_{\text{min}} | p_n >, \]  

(2.11.5.12)

which have to be shaken:
SHAKE \((|x_n>, |x_{n+1}>, |x_{n+2}>)\)

The number of iterations in SHAKE is printed under NIP2 for the solutes and under NIS2 for the solvent molecules.

2. Derive the new energy \(V_{n+1} = V(|x_{n+1}>)\) and the nonconstrained forces \(|f'_n> = f'(|x_{n+1}>)\) from the interaction function, excluding the interaction terms that act only along the constraints. The components along the constraints are removed as above by performing:

\[
|x_{n+2}'> = |x_{n+1}'> + \Delta x'|f'_{n+1}>
\]  

where

\[
\Delta x' = s_{\min} \frac{<p_n|p_n>'^{1/2} <f'_{n+1}|f'_{n+1}>'^{1/2}}{<f'_{n+1}|f'_{n+1}>'^{1/2}}
\]  

SHAKE \((|x_{n+1}'>, |x_{n+2}'>, |x_{n+2}'>)\)

and

\[
|f'_{n+1}>' = [|x_{n+2}'> - |x_{n+1}'>]/\Delta x'.
\]

The number of iterations in SHAKE is printed under NIP3 for the solutes and under NIS3 for the solvent molecules.

3. The new search direction \(|p_{n+1}>'\) could be directly obtained from \((2.11.3.10)\). It is a linear combination of \((n+2)\) shaken vectors. As \(n\) increases, the components of \(|p_{n+1}>'\) along the constraint directions will
grow, unless the search direction is also shaken at every minimization step by putting (inverting (2.11.5.12) with $|x_{n+1}^\prime\rangle$ instead of $|x_{n+1}\rangle$:

$$|p_n\rangle = [(|x_{n+1}^\prime\rangle - |x_n\rangle)/s_{\min}$$

(2.11.5.16)

Then, the new search direction becomes

$$|p_{n+1}\rangle = |r_{n+1}\rangle + \beta_n|p_n\rangle$$

(2.11.5.19)

with $\beta_n$ given by (2.11.3.11).

The parameters of the CG algorithm with SHAKE are identical to those discussed in Chapter 2.11.3 for the nonconstrained algorithm. The inner product of the constrained forces $<r_n^\prime|f_n^\prime>^{1/2}$ is denoted by RMS-F-CONS, and that of the nonconstrained forces $<r_n'|f_n'|^{1/2}$ by RMS-F-FREE in the print-out of subroutine RUNEM. Since when applying constraints, conditions (2.11.5.4) and (2.11.5.5) are not equivalent, it can occur that

$$z^2 - g_n(a)g_n(b) < 0$$

(2.11.5.20)

in (2.11.3.7), in which case the minimization is terminated.
3. GROMOS Data Structure

3.1 Introduction

GROMOS knows different types of data files, which are described in this chapter. The information concerning a molecular system is distributed over two distinct data files: a Molecular Topology (MT) file and an Atomic Coordinate (AC) file. A molecular topology file contains information about the topology of a molecular system: data on the covalent structure, atomic masses, charges, van der Waals parameters, etc. An atomic coordinate file contains information about one or many configurations of a molecular system: Cartesian coordinates, velocities, fluctuations of atoms, size of the computational (periodic) box etc. These two types of information are separately stored, since configurations change continuously during a simulation, whereas the molecular topology generally does not change. Both types of files for a specific molecular system are related through the requirement that in both the sequence of the atoms of the system is the same. This could be checked e.g. by comparing atom names from the molecular topology file with those from the atomic coordinate file. However, in order to maintain maximum flexibility, this is not done in the GROMOS programs. When molecular information like residue numbers and names or atom sequence numbers or names is present both in the molecular topology file and in an atomic coordinate file of a molecular system, the program generally uses the data from the molecular topology file and ignores the corresponding data on the atomic coordinate file.

Molecular topologies are described in Chapter 3.3, atomic coordinate files in Chapter 3.4. Other types of data files are described in Chapters 3.5-3.7.
3.2 Title record on GROMOS files

Each GROMOS file, of any type, begins with a title record. It may contain any character type of data and is meant to specify the content of a GROMOS file. When reading a GROMOS file, the title record is always printed in order to check whether the wanted file has been assigned to a specific unit number.

In the programs an array TITLE (16) has been defined (REAL*8). It is written in binary form by the statement

```
WRITE (unit) TITLE
```
and in formatted form by the statement

```
WRITE (unit, 40) TITLE
```
40 FORMAT (15A5)

It is read in binary form by

```
READ (unit) TITLE
```
and in formatted form by

```
READ (unit, 40) TITLE
```
40 FORMAT (16A5)
3.3 Molecular Topologies

3.3.1 Introduction

A molecular topology file contains information about the topology of a molecular system. In its simplest form it would contain lists of covalent bonds, angles, masses, charges, etc. for all the atoms in the molecular system. When the system contains topologically identical molecules, like water molecules in an aqueous solution or corresponding molecules in different asymmetric units in a unit cell in a crystal, these atom lists would contain redundant information. The topological information of topologically identical molecules need only to be stored for one molecule. Since a solvent generally consists of simple molecules like H$_2$O or CCl$_4$, it would generally be advantageous to avoid the overhead of handling the possibility of occurrence of internal dihedral degrees of freedom, non-bonded interactions, etc. within a solvent molecule. Therefore, a distinction is made between a general part (solute) and a more restricted part (solvent) of a molecular topology file.

For historical reasons the general part of a molecular topology file is denoted by the notation "solute" molecular topology, although it may contain any collection of molecules including solvent molecules. The restricted part of a molecular topology file is denoted by the notation "solvent" molecular topology. In general this part contains topological data on a solvent molecule, unless a solvent molecule does not fit with the following restrictions:
- a solvent molecule must be rigid:
  - no internal interactions like bond-stretch, bond-angle bending, (improper)
  - dihedral torsion and non-bonded interactions are allowed
- the internal structure of a solvent molecule is maintained by application of distance constraint forces (subroutine SHAKE) between its atoms.
- a solvent molecule consists of one charge group, the position of the first atom of a solvent molecule is taken to represent the position of this charge group.
- the residue or molecule name cannot be specified, it is predefined as SOL.
- position restraining can only be applied to the first atom of a solvent molecule.
- distance restraining cannot be applied to atoms of solvent molecules.

If a solvent molecule does not comply with these rules, its topological data must be included in the general or solute part of the molecular topology file.

A molecular topology file may contain less atoms than a corresponding atomic coordinate file. Let's assume that the former contains a set of molecules forming a "solute" of NRP atoms and a solvent molecule with NRAM atoms. In order to match this molecular topology file, an atomic coordinate file must contain the following sequence of atoms:

1. if the molecular system contains NPM "solute" molecules, the atomic coordinates of the NPM*NRP "solute" atoms (NPM*the "solute" topology sequence)
2. if the molecular system contains NSM solvent molecules, the atomic coordinates of the NSM*NRAM solvent atoms (NSM*the solvent topology sequence)

So, solvent coordinates appear at the rear of atomic coordinate files. The "solute" and solvent parts of the molecular topology file are to be chosen the smallest topologically identical units of each type.
In Chapter 3.3.2 the content of a molecular topology file is specified. In Chapter 3.3.3 its formatted form is discussed. In some applications of GROMOS, like calculating the free energy difference between two different states A and B of a system, it is required to change the molecular topology of the system from one corresponding to state A to another corresponding to state B. In section 3.3.4 the way a perturbation (change from A to B) of a molecular topology is to be specified, will be discussed.

3.3.2 Molecular topology, binary form

A molecular topology file is written in binary form by the following sequence of statements.

```
WRITE (unit) NRATT
WRITE (unit) (TYPE (K), K=1, NRATT)
WRITE (unit) NRAA2
WRITE (unit) (AANM (K), K=1, NRAA2)
WRITE (unit) NRP
WRITE (unit) (WINV(K), PANM(K), MRES(K), IAC(K), K=1, NRP)
WRITE (unit) NBTY
WRITE (unit) CB(K), BO(K), K=1, NBTY
WRITE (unit) NBONH
WRITE (unit) (IBH(K), JBH(K), ICBH(K), K=1, NBONH)
WRITE (unit) NBON
WRITE (unit) (IB(K), JB(K), ICB(K), K=1, NBON)
WRITE (unit) NTTY
WRITE (unit) (CT(K), TO(K), K=1, NTTY)
```
WRITE (unit) NTHEH
WRITE (unit) ITH(K), JTH(K), KTH(K), ICTH(K), K=1, NTHEH)
WRITE (unit) NTHE
WRITE (unit) IT(K), JT(K), KT(K), ICT(K), K=1, NTHE)
WRITE (unit) NQTY
WRITE (unit) (CQ(K), QO(K), K=1, NQTY)
WRITE (unit) NQHIH
WRITE (unit) (IQH(K), JQH(K), KQH(K), LQH(K), ICQH(K), K=1, NQHIH)
WRITE (unit) NQHI
WRITE (unit) (IQ(K), JQ(K), KQ(K), LQ(K), ICQ(K), K=1, NQHI)
WRITE (unit) NPTY
WRITE (unit) (CP(K), PD(K), NP(K), K=1, NPTY)
WRITE (unit) NPHIH
WRITE (unit) (IPH(K), JPH(K), KPH(K), LPH(K), ICPH(K), K=1, NPHIH)
WRITE (unit) NPHI
WRITE (unit) (IP(K), JP(K), KP(K), LP(K), ICP(K), K=1, NPHI)
WRITE (unit) NCAG
WRITE (unit) (INC(K), K=1, NCAG)
WRITE (unit) NRATTQ
WRITE (unit) (MPAC(K)< K=1, NRATTQ
WRITE (unit) NRATT2
WRITE (unit) (C12(K), C6(K), K=1, NRATT2)
WRITE (unit) NRP
WRITE (unit) (CG(K), INE(K), KNE(K), K=1, NRP)
WRITE (unit) NAEX
WRITE (unit) (JSNE(K), K=1, NAEX)
WRITE (unit) NRATT2
WRITE (unit) (CS12(K), CS6(K), K=1, NRATT2)
WRITE (unit) NRP
WRITE (unit) (INE14(K), KNE14(K), K=1, NRP)
WRITE (unit) NAEX14
WRITE (unit) (JSNE14(K), K=1, NAEX14)
WRITE (unit) NRAM
WRITE (unit) (WINVS(K), ANMS(K), CGS(K), IACS(K), K=1, NRAM)
WRITE (unit) NCONS
WRITE (unit) (ICONS(K), JCONS(K), CONS(K), K=1, NCONS)

This occurs in programs:

PROGMT, PROGM, PROMMT, PRORMT

GROMOS is delivered with two sets of array sizes for the molecular topology arrays:

- NORMAL (standard version)
- LARGE (L-version, subdirectory LARGE)

which are specified in chapter 6.1.

Reading of a molecular topology in binary form is done correspondingly.

This occurs in programs:

PROAHB, PROAJC, PROAVN, PROAVQ, PROAVS, PROAVX, PROBOX, PROCAB, PROCHB,
PROCMT, PROCOB, PROCOC, PROCOD, PROCOQ, PROCOX, PROCPS, PROCRY, PRODR,
PROEM, PROGCA, PROGCH, PROION, PROMD, PROMHE, PROMMT, PRONBL, PROPDF,
PROPSF, PROSD, PROTCF
Most programs do require both the solute and the solvent part of the molecular topology file to be read, even when the solvent part will not be used.

The variables and arrays of the molecular topology file have the following meaning.

NRP = number of solute atoms

WINV[1..NRP] = inverse of the mass of solute atoms

PANM[1..NRP] = atom name of solute atoms (REAL*8)

IAC[1..NRP] = integer atom code of solute atoms, determining the type of van der Waals interaction of an atom (.ie.WNATT)

INE[1..NRP] = number of neighbour atoms that are excluded from the non-bonded interaction with a solute atom

KNE[1..NRP] = denotes where the atom sequence numbers of the excluded neighbours of a solute atom can be found in array JSNE (solute, .ie.NAEX)

JSNE[1..NAEX] = excluded neighbours (solute, .ie.NRP); sequence numbers J of atoms that are excluded from the non-bonded interaction with the atom with sequence number I, are positioned at positions KNE[I]+1, ..., KNE[I]+INE[I] in array JSNE; it is assumed that I<J

NAEX = total number of excluded atoms (solute)

INE14[1..NRP] = number of third neighbour atoms of solute atoms, for which special 1-4 van der Waals interaction parameters are used when evaluating the non-bonded interaction
KNE14[1..NR] denotes where the atom sequence numbers of the third
neighbours of a solute atom can be found in array JSNE14
(solute, i.e. NAEX14)

JSNE14[1..NAEX14] = third neighbours (solute, i.e. NR);

sequence numbers J of atoms for which the 1-4 van der
Waals parameters are used when calculating the non-
bonded interaction with the atom with sequence number I,
are positioned at positions
KNE14[I]+1, ..., KNE14[I]+NE14[I]
in array JSNE14;

it is assumed that I<J

NAEX14 = total number of third neighbour atoms (solute)

NRATT = number of (van der Waals) atom types

TYPE[1..NRATT] = names of the different atom types as a function of the
integer atom code that defines an atom type (REAL*8)

MPAC[1..NRATT,1..NRATT] = pair codes for atom pairs as a function of their
integer atom codes I and J (.i.e. NRATT*(NRATT+1)/2),
the pair code is defined as
I+J*(J-1)/2 when I < J and as
J+I*(I-1)/2 when J < I

Cl2[1..NRATT*(NRATT+1)/2] = coefficient of the 1/r**12 term in the non-
bonded interaction as a function of the occurring pair
codes; so, the sequence of atom pairs with integer atom
codes ranging from 1 to NRATT is: 1-1, 1-2, 2-2, ..., 1-NRATT, 2-NRATT, ... NRATT-NRATT

C6[1..NRATT*(NRATT+1)/2] = coefficient of the -1/r**6 term in the non-
bonded interaction as a function of the occurring pair
codes.
CS12[1..NRATT*(NRATT+1)/2] = coefficient of the $1/r^{12}$ term in the 1-4 non-bonded interaction between third neighbour atoms as a function of the occurring pair codes.

CS6[1..NRATT*(NRATT+1)/2] = coefficient of the $-1/r^{6}$ term in the 1-4 non-bonded interaction between third neighbour atoms as a function of the occurring pair codes.

CG[1..NRP] = charge of solute atoms

NCAG = number of charge groups in a solute

INC[1..NCAG] = charge group pointer-list; INC[I] specifies the position of the last atom of the I-th charge group in the atom sequence (I.le.NRP); the atoms of a charge group must have sequential atom sequence numbers.

NRAA2 = number of residues in a solute

MRES[1..NRP] = residue sequence number of solute atoms (I.le.NRAA2)

AANM[1..NRAA2] = residue names as a function of the residue sequence number (REAL*8)

NBTY = number of covalent bond types

CB[1..NBTY] = force constant of the harmonic bond-stretching term of the interaction as a function of the bond-type code

BO[1..NBTY] = bond-length at minimum energy of the harmonic bond-stretching term as a function of the bond-type code

NBONH = number of bonds involving H-atoms in the solute

IBH,JBH[1..NBONH] = atom sequence numbers of the atoms forming a bond i-j as a function of the bond sequence number (I.le.NRP)

ICBH[1..NBONH] = bond-type code as a function of the bond sequence number (I.le.NBTY)

NBON = number of bonds involving no H-atoms in the solute
\[ IB,JB[1..NBON] = \text{atom sequence numbers of the atoms forming a bond } i-j \]
\[ \text{as a function of the bond sequence number } (\text{.le.NRP}) \]

\[ IC[1..NBON] = \text{bond-type code as a function of the bond sequence number } (\text{.le.NBTY}) \]

\[ NTTY = \text{number of bond-angle types} \]

\[ CT[1..NTTY] = \text{force constant of the harmonic bond-angle bending term of the interaction as a function of the bond-angle type code} \]

\[ TO[1..NTTY] = \text{bond-angle at minimum energy of the harmonic bond-angle bending term as a function of the bond-angle type code} \]

\[ NTHEH = \text{number of bond-angles involving } H\text{-atoms in the solute} \]

\[ ITH,JTH,KTH[1..NTHEH] = \text{atom sequence numbers of the atoms forming a bond-angle } i-j-k \text{ as a function of the bond-angle sequence number } (\text{.le.NRP}) \]

\[ ICTH[1..NTHEH] = \text{bond-angle type code as a function of the bond-angle sequence number } (\text{.le.NTTY}) \]

\[ NTHE = \text{number of bond-angles involving no } H\text{-atoms in the solute} \]

\[ IT,JT,KT[1..NTHE] = \text{atom sequence numbers of the atoms forming a bond-angle } i-j-k \text{ as a function of the bond-angle sequence number } (\text{.le.NRP}) \]

\[ ICT[1..NTHE] = \text{bond-angle type code as a function of the bond-angle sequence number } (\text{.le.NTTY}) \]

\[ NQTY = \text{number of improper dihedral types} \]

\[ CQ[1..NQTY] = \text{force constant of the harmonic improper dihedral term in the interaction as a function of the improper dihedral type code} \]

\[ QQ[1..NQTY] = \text{improper dihedral at minimum energy of the harmonic} \]
improper dihedral term as a function of the improper
type code

NQHIH = number of improper dihedrals involving H-atoms in the
solute

IQH,JQH,KQH,LQH[1..NQHIH] = atom sequence numbers of the atoms forming
improper dihedral i-j-k-l as a function of the improper
dihedral sequence number (.le.NRP)

ICQH[1..NQHIH] = improper dihedral type code as a function of the
improper dihedral sequence number (.le.NQTY)

NQHI = number of improper dihedrals involving no H-atoms in
the solute

IQ,JQ,KQ,LQ[1..NQHI] = atom sequence numbers of the atoms forming improper
dihedral i-j-k-l as a function of the improper dihedral
sequence number (.le.NRP)

ICQ[1..NQHI] = improper dihedral type code as a function of the
improper dihedral sequence number (.le.NQTY)

NPTY = number of dihedral types

CP[1..NPTY] = force constant of the sinusoidal dihedral term in the
interaction as a function of the dihedral type code

PD[1..NPTY] = phase shift of the sinusoidal dihedral term in the
interaction as a function of the dihedral type code

NP[1..NPTY] = multiplicity of the sinusoidal dihedral term in the
interaction as a function of the dihedral type code

NPHIH = number of dihedrals involving H-atoms in the solute

IPH,JPH,KPH,LPH[1..NPHIH] = atom sequence numbers of the atoms forming
dihedral i-j-k-l as a function of the dihedral
sequence number (.le.NRP)

ICPH[1..NPHIH] = dihedral type code as a function of the dihedral
sequence number (.le.NPITY)

NPHI = number of dihedrals involving no H-atoms in the solute

IP,JP,KP,LP[1..NPHI] = atom sequence numbers of the atoms forming dihedral
1-j-k-l as a function of the dihedral sequence number
(.le.NRP)

ICP[1..NPHI] = dihedral type code as a function of the dihedral
sequence number (.le.NPITY)

NRAM = number of atoms per solvent molecule

WINVS[1..NRAM] = inverse of the mass of solvent atoms

ANMS[1..NRAM] = atom name of solvent atoms (REAL*8)

IACS[1..NRAM] = integer atom code of solvent atoms; determining the
type of van der waals interaction of an atom (.le.NRATT)

CGS[1..NRAM] = charge of solvent atoms

NCONS = number of distance constraints within a solvent molecule

ICONS,JCONS[1..NCONS] = atom sequence numbers of the atoms forming the
constraint i-j as a function of the constraint sequence
number (.le.NRAM)

CONS[1..NCONS] = square of the constraint length as a function of the
constraint sequence number

We note that when reading a molecular topology in binary form the
arrays must be sufficiently large. A molecular topology written with NORMAL
molecular topology array sizes can be read with NORMAL or with LARGE array
sizes. However, a molecular topology written with LARGE molecular topology
array sizes can be read with LARGE array sizes, but not with NORMAL array
sizes. Secondly, we note that on 32-bit machines the arrays TITLE, TYPE,
AANM, PANM and ANMS must be defined REAL*8.
Examples of binary molecular topology files are named:

MT*****.BIN

3.3.3 Molecular topology, formatted form

A molecular topology can be converted from binary form to formatted form.

This occurs in program:

PROCMT

A molecular topology can also be converted from formatted form to binary form.

This occurs in program:

PRORMT

The structure of the formatted form is described in the heading comment of program PRORMT (Chapter 16), the variable names are identical to
those described in the previous paragraph. A few variables of the molecular
topology do not need to be specified, since they can be derived from other
data. These are MFAC[1..NRATT], NAEX, KNE[1..NRP], NAEX14,
KNE14[1..NRP], NCAG and INC[1..NCAG]. The latter two are derived from the
atomic charge group codes ICG[1..NRP]. The atoms of a charge group must
have sequential atom sequence numbers. The last atom of a charge group is
defined by ICG=1, the other must have ICG=0.

Examples of formatted molecular topology files are named:
MT*****.FMT or IRMT***.DAT

3.3.4 Perturbation of a molecular topology

When simulating a molecular system or when analyzing a set of
conformations of a molecule, the molecular topology file of the system
remains unchanged. This is the rationale for separating topological and
force field information resident in a molecular topology file from
conformational information resident in atomic coordinate files. If a change
of topological data or force field parameters is required, a new changed
molecular topology file has to be generated by one of the molecular topology
building programs.

However, when applying the thermodynamic integration formalism in
order to determine the difference in free energy between two states A and B
of a molecular system, the molecular topology of the system has to change in the course of a MD simulation in a continuous way from the one corresponding to state A to the one corresponding to state B or vice versa. In general the difference between state A and state B is limited to a restricted part of the system, that is, a few tens of atoms. Therefore, this difference is represented by a perturbation potential, which contains information on how to change or perturb the molecular topology of state A in order to obtain the one of state B.

The implementation of the parametrization of the Hamiltonian of a molecular system in terms of a parameter $\lambda$ has been described in Chapter 2.13.2. It has been implemented in program PROMD and its subroutines. The contents of the file containing the perturbation potential is described in the heading comment of program PROMD (see Chapter 16) under TAPE27. Here, a few comments are given.

- Since the perturbation cannot create extra atoms, the state of the molecular system that contains the largest number of atoms must be chosen to be state A.
- The molecular topology that is read, the unperturbed one, corresponds to state A.
- The value $\lambda = \text{RLAM} = 0$ corresponds to state A of the system (unperturbed molecular topology); the value $\lambda = \text{RLAM} = 1$ corresponds to state B of the system (perturbed molecular topology).
- Since the molecular topology of state A is known in PROMD, only the sequence numbers JLA of the perturbed atoms (number = NJLA) with their integer atom codes IACB (determining the van der Waals parameters), charges CGS and masses WMB in state B must be specified.
- The change from state A to state B may involve the breaking or formation of a covalent bond between two atoms. In that case, the excluded
neighbours and the third neighbours of these atoms will be different in state A and in state B. The type of interaction, i.e. normal interaction, 1-4 or third neighbour interaction, must be changeable. The standard non-bonded interaction subroutines (NSPML, NONBML, NBFAL, NONBAL, NONBB, NONBP) do only allow for a continuous change from one integer atom code (IACA) to another (IACB), but not for a change of type in the sense of normal, third neighbour or excluded neighbour interaction. A change of type is implemented by specifying all (number = NEB) pairs of atoms (IEB, JEB) for which the type is to be changed when moving from state A to state B. The perturbation potential file contains the variable IETA and IETB for each pair, denoting which of the three types of interaction is applicable in state A and in state B. The interaction for these specified atom pairs are evaluated in a special subroutine (NONBPL), so in order to avoid double counting, all these specified pairs must be excluded atom pairs in the unperturbed molecular topology (state A).

- The perturbation of the bond-stretching, bond-angle bending, improper dihedral or dihedral interaction terms is specified by giving the sequence numbers of the atoms involved with the force field parameters in state A as well as in state B.
- The switch NTUG can be used to change the units of the perturbation potential file such that they match the units of the unperturbed molecular topology file.

The heading comment of program PROMD specifies the formats of the perturbation potential file.
3.4 Atomic Cartesian coordinates and related quantities

Here, it is described in which form the atomic coordinates, velocities, mobilities and related quantities are stored on disk by the programs of GROMOS such that these can be read by the programs of GROMOS. Two types of data files are distinguished. Firstly, files that contain data concerning only a single configuration or time frame of a molecular system. Secondly, when performing MD a whole time series of configurations or trajectory of a molecular system is produced. Since trajectories require much more storage capacity, they are stored in a reduced, trajectory GROMOS form, which is different from the standard single configuration GROMOS form. The latter contains more molecular information, but requires more storage space per configuration. Note that any GROMOS file begins with a title record.
3.4.1 Storage of a single configuration or time frame

A single configuration of a molecular system can be stored in two forms, standard binary form and standard formatted form. Other non-standard forms are discussed in section 3.4.1.3.

3.4.1.1 Standard binary form

A molecular configuration containing NR atoms with Cartesian coordinates X[1..3*NR] is written in standard binary form by the statement:

```
WRITE (unit) (X[I3], I3 = 1, 3*NR)
```

This occurs in programs:

```
PROBOX, PROCRY, PROEM, PROGCA, PROGCH, PROGWH
```

Reading these coordinates is done by the statement:

```
READ (unit) (X[I3], I3 = 1, 3*NR)
```

This occurs in programs:

```
PROBOX, PROCHB, PROCMT, PROCOD, PROCOQ, PROCOQ, PROCOX, PROFPS,
PROCRY, PROEM, PROGCA, PROGCH, PROGWH, PROION, PROMD, PRONBL,
PROPDF, PROPSF, PROSD
```

When storing intermediate configurations of a MD simulation, the atomic velocities V[1..3*NR] at time t-\(dt/2\) are stored together with the
atomic coordinates $X$ at time $t$, where $dt$ is the MD time step. In that case writing is done by

```
WRITE (unit) (X[13], I3 = 1, 3*NR), (V[13], I3 = 1, 3*NR)
```

This occurs in program:

```
PROMD
```

Reading is done by

```
READ (unit) (X[13], I3 = 1, 3*NR), (V[13], I3 = 1, 3*NR)
```

This occurs in program:

```
PROMD, PROSD
```

When performing stochastic dynamics (SD) using the leap-frog algorithm, the integrals of the stochastic forces over time are correlated between successive time steps [MD87]. Therefore, when storing intermediate configurations $X[1..3*NR]$ (at time $t$) and velocities $V[1..3*NR]$ (at time $t-dt/2$) of an SD simulation, the **stochastic integrals** $SX[1..3*NR]$ (over the time interval ($t-dt/2$, $t$)) are added to each configuration by

```
WRITE (unit) (SX[13], I3 = 1,3*NR).
```

The current **random number generator seed** $IG$ is added by

```
WRITE (unit) IG
```
in order to be able to continue the generation of sequence of random numbers.

This occurs in program:

PROSD

Reading of the stochastic integrals and random number generator seed is done by the corresponding READ statement.

This occurs in program:

PROSD

Atomic mobilities or fluctuations are stored in binary form by writing the second moment $X^2X$ (and sometimes higher moments $X^3$ and $X^4$) of the coordinate distribution. Writing (average) atomic coordinates $X[1..3, 1..NR]$ and the higher moments of the distribution is done by

\[
\text{WRITE (unit) } ((X[K1,I], K1=1,3), I=1, NR)\\
\text{WRITE (unit) } (((X[K1,I]*X[K2,I], K1=1,K2), K2=1,3), I=1, NR)\\
\text{WRITE (unit) } ((((X[K1,I]*X[K2,I]*X[K3,I], K1=1,K2), K2=1,K3), K3=1,3), I=1, NR)\\
\text{WRITE (unit) } ((((X[K1,I]*X[K2,I]*X[K3,I]*X[K4,I], K1=1,K2), K2=1,K3), K3=1,3), I=1, NR)
\]
This occurs in program:

PROAVX

When averaging solvent coordinates only the sum of the diagonal elements of the second moment tensor of the coordinate distribution is written after the averaged coordinates:

WRITE (unit) ((X[K1,I],K1=1,3),I=1,NR)
WRITE (unit) (X[1,I]**2 + X[2,I]**2 + X[3,I]**2,I = 1, NR)

This occurs in program:

PROAVS

Reading of this type of information is done by the corresponding READ statements.

This occurs in programs:

PROAVS, PROAVX, PROCAB, PROCOB, PROCOC, PROCO, PROCPS

When performing MD at constant pressure, the dimensions of the (periodic) box that contains the molecular system are a function of time. In that case the x-, y- and z-box-lengths BOX[1..3] are added to each configuration by

WRITE (unit)(BOX[M], M = 1,3)
This occurs in program:

PROMD

Reading of the box-lengths is done by the corresponding READ statement.

This occurs in programs:

PROCOD, PROCOQ, PROCOX, PROMD.

When performing MD at constant pressure and with restraining atoms to a specified set of restrained atom reference positions, these reference positions are a function of time. In that case the Cartesian coordinates of the reference positions X(1..3*NR) are added to each configuration by

WRITE (unit)(X(I3),I3=1,3*NR)

This occurs in program:

PROMD

Reading of the reference positions is done by the corresponding READ statement.

This occurs in programs:

PROMD
When applying a perturbation potential in a MD simulation in order to determine a free energy difference between two states of a molecular system, the perturbation parameter RLAM (lambda), the cumulative value of the derivative of the perturbation potential energy DSVDL and of the perturbation kinetic energy DSKDL with respect to lambda are a function of time. In that case these quantities are added to each configuration by

```
WRITE (unit) RLAM, DSVDL, DSKDL
```

This occurs in programs:

```
PROMD, PROSD
```

Reading of these quantities is done by the corresponding READ statement.

```
This occurs in programs:
PROMD, PROSD
```

### 3.4.1.2 Standard formatted form

A molecular configuration containing NR atoms with Cartesian coordinates $X[1..3,1..NR]$ is written in standard formatted form by the statements:

```
WRITE (unit, 12) NR
DO 10 J = 1, NR
  10 WRITE (unit, 12) MRES[J], AANMA[J], PANM[J], J, (X[M, J], M=1, 3)
  12 FORMAT(I5,2A5,I5,3F8.3)
```
Here, MRES[J] is the residue (or molecule) sequence number, AANMA[J] is the (amino acid) residue or molecule name, PANM[J] is the (protein) atom name of the atom with atom sequence number J.

This occurs in programs:

PROBOX, PROBRK, PROEM, PROCCA, PROION, PROPSF, PROSSC

Reading of atomic coordinates is done by the corresponding READ statement.

This occurs in programs:

PROBOX, PROCHE, PROCGD, PROCOQ, PROCOX, PROCP, PROEM, PROCCA, PROGMT, PROION, PROMD, PROMMT, PRONBL, PROPBF, PROPSF, PROSD, PROSSC

Note the following. When a program has read a molecular topology file it takes the topology information MRES[J], AANMA[J], PANM[J] and J from there and it ignores these quantities on the coordinate file.

When storing intermediate configurations of a MD simulation, the atomic velocities $V[1..3,1..NR]$ at time $t-dt/2$ are stored together with the atomic coordinates $X$ at time $t$, where $dt$ is the MD time step. In that case writing is done as follows:

```
WRITE (unit, 12) NR
DO 10 J = 1, NR
  10 WRITE (unit, 12) MRES[J], AANMA[J], PANM[J], J,
     (X[M,J], M=1,3), (V[M,J], M=1,3)
```

12 FORMAT (I5, 2A5, I5, 3F8.3, 3F8.4)
This occurs in programs:

PROCMT, PROMD

Reading of velocities is done correspondingly.

This occurs in programs:

PROCMT, PROMD, PROSD

When storing intermediate configurations of a SD simulation, the stochastic integrals $S_X[1..3, 1..NR]$ over the time interval $(t-dt/2, t)$ and the random number generator seed IG are added to each configuration as follows:

```
DO 10 J = 1, NR
10 WRITE (unit, 12) MRES[J], AANMA[J], PANM[J], J, SX[M, J], M = 1, 3
WRITE (unit = 13) IG
12 FORMAT (I5, 2A5, I5, 3E16.6)
13 FORMAT (I10)
```

This occurs in program:

PROSD
Reading of stochastic integrals and random number generator is done correspondingly.

This occurs in program:

```
PROSD
```

Atomic mobilities or fluctuations are stored in formatted form by writing isotropic B-factors BFAC[1..NR] and 8*pys**2 times the square of the atomic positional fluctuation DX[1..3,1..NR] (anisotropic case).

In the isotropic case writing is done by

```
WRITE (unit, 12) NR
DO 10 J = 1, NR
10 WRITE (unit, 12) MRES[J], AANMA[J], PANM[J], J,
   \( X(M,J), M = 1,3 \), BFAC[J]
12 FORMAT (I5, 2A5, I5, 3F8.3, F8.4)
```

This occurs in programs:

```
PROAVS, PROCMT, PROCRY, PROC81, PROGCH, PROGWH
```

Reading of isotropic B-factors is done correspondingly.

This occurs in programs:

```
PROCMT, PROCOB, PROCOC, PROCOS, PROCPS, PROCRY, PROEM, PROGCH,
PROGWH, PROMD, PROSD
```
In the anisotropic case writing is done by

```
WRITE (unit, 12) NR
DO 10 J = 1, NR
10 WRITE (unit, 12) MRES[J], AANMA[J], PANM[J], J,
    (X[M,J], M=1,3), BFAC[J],
    ((B*py**2*DX[K1,J]*DX[K2,J], K1=1, K2), K2=1,3)
```

12 FORMAT (I5, 2A5, I5, 3F8.3, F8.4, 6F8.4)

This occurs in programs:

PROAVX, PROC32

Reading of anisotropic B-factors is done correspondingly.

This occurs in programs:

PROCAB

When performing MD at constant pressure, the dimensions of the (periodic) box that contains the molecular system are a function of time. In that case the x-, y- and z-box-lengths BOX[1..3] are added to each configuration by

```
WRITE (unit, 13) (BOX[M], M=1,3)
```

13 FORMAT (3F10.5)
This occurs in programs:

PROMD

Reading of the box-lengths is done correspondingly.

This occurs in programs:

PROCOD, PROCQ, PROCX, PROMD

When performing MD at constant pressure and with restraining atoms to a specified set of restrained atom reference positions, these reference positions are a function of time. In that case Cartesian coordinates of the reference positions $XC[1..3,1..NR]$ are added to each configuration by

DO 20 J = 1, NR
20 WRITE (unit, 22) MRES[J], AANMA[J], PANM[J], J, (XC[M,J],M=1,3)
22 FORMAT (I5, 2A5, I5, 3F8.3)

This occurs in programs:

PROMD

Reading of reference positions is done correspondingly.

This occurs in programs:

PROMD
When applying a perturbation potential in a MD simulation in order to
determine a free energy difference between two states of a molecular system,
the perturbation parameter RLAM (lambda), the cumulative value of the
derivative of the perturbation potential energy DSVDL and of the
perturbation kinetic energy DSKDL with respect to lambda are a function of
time. In that case these quantities are added to each configuration by

{\texttt{WRITE (unit, 14) RLAM, DSVDL, DSKDL}}

{\texttt{14 FORMAT (3E14.7)}}

This occurs in programs:
{\texttt{PROMD, PROSD}}

Reading of these quantities is done correspondingly.

This occurs in programs:
{\texttt{PROMD, PROSD}}

When applying SD, atomic friction coefficients GAM[1..NR] must be
defined in some way. They may either be calculated in subroutine FRIC, or
specified in an atomic friction coefficient file, which is read by the
statements:

{\texttt{READ (unit, 11) NR}}

{\texttt{DO 10 J = 1, NR}}

{\texttt{10 READ (unit, 12) GAM(J)}}

{\texttt{11 FORMAT (I5)}}

{\texttt{12 FORMAT (44X, F8.4)}}
This occurs in program:

PROSD

When a molecular system is reduced by selecting only part of the atoms, this can be done by applying either a distance criterion or an atom (selection) pointer list JLIST[1..NLIS]. The atom sequence numbers of the NLIS atoms that are to be selected are stored in an atom pointer list file by the statements:

WRITE (unit, 11) NLIS
DO 10 N = 1, NLIS
10 WRITE (unit, 12) JLIST[N]
11 FORMAT (I5)
12 FORMAT (15X, I5)

This occurs in program:

PROCMT

Reading of an atom pointer list is done correspondingly.

This occurs in programs:

PROCMT, PROMCP
Examples of formatted atomic coordinate files are named:

**x**.DAT

### 3.4.1.3 Other non-standard forms

Program PROPDF writes an atomic Cartesian coordinate file in a specific form, using MCF (master coordinate file) format, which form is required for transferring coordinates to the Groningen picture system.

Programs PROBRK, PROCSI, and PROCIS2 can read a variety of strange atomic coordinate formats and can convert those to GROMOS standard formatted form.

### 3.4.2 Storage of trajectories or series of configurations and related quantities

A trajectory or a series of configurations of a molecular system can be stored in three forms, trajectory binary form, trajectory packed form and trajectory formatted form. Writing and reading of these three forms is done by subroutine PACK. Other (non-standard) forms of storing trajectories are discussed in Chapter 3.4.2.2. Storage of energies belonging to a trajectory is discussed in Chapter 3.4.2.3.
3.4.2.1 Storage of trajectories by subroutine PACK

Trajectories that are generated in a MD simulation can be stored by calling subroutine PACK every NTWX time steps to store a single configuration (atomic Cartesian coordinates) \(X[1..3*NR]\) of NR atoms or every NTWV time steps to store a set of atomic velocities \(V[1..3*NR]\). When performing MD at constant pressure, the box-lengths BOX \([1..3]\) have to be stored together with the coordinates \(X\). In that case each call of PACK with argument \(X[1..3*NR]\) is followed by a call of PACK with argument BOX\([1..3]\).

Subroutine PACK is called at the required time steps in subroutine RUNMD which is in turn called by program PROMD to perform a MD simulation.

Subroutine PACK may write \((X\text{ or } V\text{ or } \text{BOX, etc.) in binary form}
\[
\text{WRITE (unit) (X[I3], I3=1,3*NR)}
\]
or in formatted form
\[
\text{WRITE (unit, 6) (X[I3], I3=1,3*NR)}
\]
\[
6 \text{ FORMAT (10F8.3)}
\]
or it may pack more than one real in one word using CDC Cyber 170/760 assembler routines SIZEI, PCK and UCK.

Writing by subroutine PACK occurs in programs:

\[
PROMCF, \text{ PROMD(RUNMD), PROSD(RUNSD)}
\]

Subroutine PACK may read \(X\text{ or } V\text{ or } \text{BOX by the corresponding READ statements or by unpacking (Cyber 170/760). Besides, PACK can read atomic coordinates }X\text{ in standard formatted form as defined in Chapter 3.4.1.2.}
Reading by subroutine PACK occurs in programs:

PROAHB, PROAJC, PROAVN, PROAVQ, PROAVS, PROAVX, PROCC, PRODR,
PROMCF, PROMHB, PROPSF, PROTCF

Examples of trajectory files written by subroutine PACK are named:

***R***.DAT

3.4.2.2 Storage of trajectories in other non-standard forms

Program PROPSF writes a trajectory file in the form of oblique contra-
variant fractional coordinates following a special format, which is required
by a structure factor calculating program.

3.4.2.3 Storage of energies belonging to trajectories

The total energies that are calculated from different terms in the
potential function in a MD simulation and some other quantities collected in
array ENER[1..60], can be stored every NTWE time steps in formatted form:

WRITE (unit, 18) (ENER[k], k=1,60)

18 FORMAT (5E16.8)

The quantities are:

ENER [1] = total energy of the molecular system
ENER [2] = total kinetic energy of the molecular system
ENER [3] = total kinetic energy of solutes
ENER [4] = total kinetic energy of the solvent
ENER [5] = scale factor for scaling the solute temperature
ENER [6] = scale factor for scaling the solvent temperature
ENER [7..9] = box-lengths of periodic box
ENER [10] = volume of periodic box
ENER [11..13] = pressure along x-, y- and z-axes
ENER [14] = total pressure
ENER [15..17] = total kinetic energy of centers of mass of all molecules along x-, y- and z-axes
ENER [18] = total kinetic energy of centers of mass of all molecules
ENER [19..21] = virial along x-, y- and z-axes
ENER [22] = total virial
ENER [23] = total potential energy of the molecular system
ENER [24] = total energy of bond-stretching terms involving H-atoms (solutes)
ENER [25] = total energy of bond-stretching terms involving non-hydrogen atoms (solutes)
ENER [26] = total energy of bond angle bending terms involving H-atoms (solutes)
ENER [27] = total energy of bond-angle bending terms involving non-hydrogen atoms (solutes)
ENER [28] = total energy of improper (harmonic) dihedral angle terms involving H-atoms (solutes)
ENER [29] = total energy of improper (harmonic) dihedral angle terms involving non-hydrogen atoms (solutes)
ENER [30] = total energy of (sinusoidal) dihedral angle terms involving
H-atoms (solutes)

ENER [31] = total energy of (sinusoidal) dihedral angle terms involving non-hydrogen atoms (solutes).

ENER [32..41] = total electric energy between atoms belonging to four different groups (G_1..G_4) of which the fourth group contains all solvent molecules; the order is the following:
G_1-G_1, G_1-G_2, G_2-G_2, G_1-G_3, G_2-G_3, G_3-G_3, G_1-G_4, G_2-G_4, G_3-G_4, G_4-G_4

ENER [42..51] = idem but for the total Lennard-Jones energy

ENER [52] = total energy of atom-atom distance restraint term

ENER [53] = total energy of atom position restraining term

ENER [54] = total energy of dihedral restraining term

ENER [55] = perturbation parameter lambda

ENER [56] = derivative of the potential energy with respect to lambda

ENER [57] = derivative of the kinetic energy with respect to lambda

ENER [58] = ENER [55] * delta-lambda

ENER [59] = ENER [56] * delta-lambda

ENER [60] = sum over all previous ENER [57] values

ENER [61] = run over all previous ENER [58] values

Writing of energies occurs in programs:

PROMD(RUNMD), PROSD(RUNSD)
3.5 Residue Topology Building Blocks

3.5.1 Introduction

Most programs of GROMOS do require a binary molecular topology file containing the topological and force field data concerning the molecular system that is considered. Such a binary molecular topology file can be obtained by converting a formatted one using program PRODRT. However, specifying a complete molecular topology for a large molecule like a protein in formatted form is a tedious task. Long lists of atomic properties have to be typed. Therefore, GROMOS contains a program PROCMRT that is able to generate a complete molecular topology from molecular topology building blocks, that is, molecules or parts of molecules like amino acid residues, nucleotides, etc. which are constituting the molecular system that is considered. The building blocks are linked in order to form the wanted molecular topology.

Linking of building blocks consisting of separate, non-covalently connected, molecules is straightforward. This will be discussed in Chapter 3.5.2 together with the content and format of a residue topology building block file. The linking of covalently connected building blocks by PROCMRT demands a set of rules to be satisfied by the residue topology building blocks. These rules will be discussed in Chapters 3.5.3-3.5.6. The basic information concerning the contents, format and restrictions of a residue topology building block file can be found in the heading comment of program PROCMRT (Chapter 16).
Reading occurs in program:
PROGMT

Examples of residue topology building block files are named:
RT***.DAT

3.5.2 Separate molecules

The contents of the residue topology building block file is described in the heading comment of program PROGMT (see Chapter 16) under TAPE11. Here, a few comments are given.
- The file begins with defining atom types (TYPE, REAL*8) in terms of integer atom codes and atomic masses WASS for the different atom types (number=NRATT). TITLE arrays (REAL*8) can be used to store comments. This part of the file must be identical to the first part of the corresponding interaction function parameter file (Chapter 3.6) as far as the definition of integer atom codes and masses is concerned.
- The variable NRNE controls nearest neighbour exclusions when linking building blocks. It will be discussed in the next paragraph.
- NRES yields the total number of residue topology building blocks in the file. Subsequently, the data of the various building blocks are
- Each building block has a residue code NMRC and a residue name RNME(REAL*8). These names are to be used in the input of PROGRT to identify the wanted building block.
- The number of atoms (NMAT), covalent bonds (NMB), bond-angles (NMBA), improper dihedrals (NMIDA) and dihedrals (NMDA) is specified.
- When a force field contains a special hydrogen bonding interaction function term, the number of potential hydrogen bond donors (NMPD) and acceptors (NMPA) must be specified. The GROMOS force field does not contain such a term, so NMPD = NMPA = 0 in the GROMOS residue topology building block files.
- On the interaction function parameter file (Chapter 3.6) the type of bond, bond-angle, improper dihedral or dihedral interaction is characterized by a sequential code, the sequential bond, bond-angle, improper dihedral or dihedral code. In a residue topology building block these sequential codes can be explicitly specified for all occurring bonds (MCBL), bond-angles (MCBA), improper dihedrals (MCIA) or dihedrals (MCD). The alternative is to derive the sequential bond, bond-angle, improper dihedral or dihedral angle codes from the integer atom codes of the atoms that are forming the bond, bond-angle, improper dihedral or dihedral angle. This is only possible if the atom types uniquely define the type of bond, bond-angle or (improper) dihedral angle. This condition is not always satisfied. Examples are the bonds in the building block retinol (RTOL) or the dihedral angles in the nucleotides (DADE, DCYT, DGUA, DTHY) on the GROMOS residue topology building block files. Whether the sequential codes are read or derived from integer atom codes is controlled by the switches NTBL, NTBA, NTIDA and NTDA.
- The atom sequence numbers of the atoms involved in covalent bonds are denoted by MB. Correspondingly those involved in bond angles, improper dihedrals and dihedral angles are denoted by MBA, MIDA and MDA.

- Since the GROMOS force field does not contain a special hydrogen bonding interaction term, no donor or acceptor atom sequence numbers (MPD, MPA) are specified on the GROMOS residue topology building block files.

- Finally, atomic properties are given. Atom names ANM (REAL*8) can be chosen freely. Integer atom codes IAMC will determine the van der Waals interaction parameters and the atomic mass. The charges are denoted by CGM and the charge group codes by ICGM. The atoms forming a charge group must have sequential sequence numbers. The last atom of a group is denoted by ICGM = 1, the others must have ICGM = 0.

- The number of neighbours excluded from non-bonded interaction with an atom is given in MAE. The sequence numbers J of the excluded atoms belonging to the atom with sequence number I are given in MSAE (columnwise). It is required that I < J.

An example of a residue topology building block containing a separate molecule is the trimethoprim building block to be found under the name TMP or TMPH in the RT37*.DAT files.

3.5.3 Amino acid residues

When amino acid residues serve as residue topology building blocks, a few extra rules must be satisfied compared to the case of separate molecules discussed in the previous paragraph. These rules are due to the fact that in a chain of amino acid residues the bonds, bond-angles and (improper) dihedral angles involve atoms from different building blocks. Also excluded
neighbours may reside in different building blocks. These rules can be found in the heading comment of program PROGMT (see Chapter 16) under TAPE11. Here, a few comments are given.

- The **sequence numbers** of the atoms within an amino acid residue follows the order:
  
  -N-H-CA-sidechain-C=O-

- When listing a bond (I-J), bond-angle (I-J-K), improper dihedral (I-J-K-L) or dihedral (I-J-K-L) connecting atoms with sequence numbers I, J, K or L in two residues with residue sequence numbers M-1 and M or M and M+1, through a peptide C-N link, the following rules apply: for the bond I-J, neither I nor J may lie in residue M-1 and only J may lie in residue M+1; for the bond-angle I-J-K, only I may lie in residue M-1, and only K may lie in residue M+1; for the improper dihedral I-J-K-L, only J or K may lie in residue M-1, and only I or J or K or L may lie in residue M+1; for the dihedral I-J-K-L, only I or J and K may lie in residue M-1, when I is the CA-atom of residue M-1 it must be specified as -2, and only L may lie in residue M+1.

- **Disulfide bridges** can be made between the sidechains of residues with names CYS1 and CYS2. In the CYS1 residue topology building block the atoms of the CYS2 residue building block are identified by a negative sign of the atom sequence numbers. The rules for listing the bond, bond angles and dihedrals of a S-S link between CYS1 and CYS2 residues are the following: for the bond I-J, only in CYS1 -J may denote atom J in CYS2; for the bond-angle I-J-K, only in CYS1 -K or -J and -K may denote atoms J and K in CYS2; for the dihedral I-J-K-L, only in CYS1 -L or -K and -L or -J, -K and -L may denote atoms J, K and L in CYS2.

- The **covalent link between a Histidine and the heme-group** follows the rules
for the S-S link with HIS1 = CYS1 and HEME = CYS2. Here HIS1 is the
residue topology building block representing the covalently bound
Histidine and the name of the heme-group reads HEME. For the improper
dihedral I-J-K-L, only in HIS1 -I or -L may denote atoms I and L in HEME.

- The rules for specifying excluded neighbours in adjacent residues will
depend on how many nearest neighbours are to be excluded. When NRNE = 2,
only the first and second covalently bound neighbours are excluded. In
MAE the exclusions of the C- and the O-atom in residue M-1 must precede
those of the atoms of residue M. The exclusions of the C- and O-atom in
residue M cannot be specified since they will depend on the type of
residue M+1. The GROMOS force fields assume NRNE = 2.

- When NRNE = 3, the first, second and third covalently bound neighbours are
excluded. In this case the excluded neighbours of the CA-atom of residue
M-1, denoting atoms beyond the first atom (N) of residue M, must precede
those of the C- and O-atom of residue M-1 and those of the atoms of
residue M. M must be <42. Again, the exclusions of the C- and O-atom in
residue M cannot be specified. The exclusions of the CA-atom of residue M
are specified as far as they belong to residue M. The GROMOS force fields
do not use this option.

- For the disulfide link and the His1–Heme link, the rules for specifying
excluded neighbours are the following. In CYS1 or in HIS1 the excluded
neighbours residing in CYS2 or HEME are denoted by a negative sign of
their atom sequence numbers.

As an example of a residue topology building block containing an amino acid
residue is the Alanine building block to be found under the name ALA in the
RT***.DAT files.
3.5.4 Nucleotides

When nucleotides serve as residue topology building blocks, the rules for connecting the phosphor-oxyer O-P links are identical to those for connecting the peptide C-N links between amino acid residues. These rules can again be found in the heading comment of program PROGMP (see Chapter 16) under TAPE11. Here, a few additional comments are given.

- The sequence numbers of the atoms within a nucleotide follow the order: 
  -P-OP1-OP2-OS*-C5*-sugar ring plus base-C3*-O3*-

- The rules for listing the bonds, bond-angles, improper dihedrals and dihedral angles involving the O3*-P link are identical to those for the C-N link between residues.

- The rules for specifying excluded neighbours are also identical. When NRNE = 2, the exclusions of the C3*- and the O3*-atom in nucleotide M-1 must precede those of the atoms of residue M, etc.

- The option NRNE = 3 has not been implemented for nucleotides.

An example of a residue topology building block containing a nucleotide is the Adenine building block to be found under the name DADE in the RT37*.DAT files.

3.5.5 Glucose units

When glucose units serve as residue topology building blocks, the rules for connecting the glucosidic O-C links are identical to those for connecting the peptide C-N links between amino acid residues. These rules
can again be found in the heading comment of program PROGMT (Chapter 16) under TAPE11. Here, a few additional comments are given.

- The sequence numbers of the atoms within a glucose unit follow the order: C1-05-C5-rest of glucose-C4-04- in the case of a 1-4 linkage.

- The rules for listing the bonds, bond angles, improper dihedrals and dihedral angles involving the C4-C1 link are identical to those for the C-N link between residues.

- The rules for specifying excluded neighbours are also identical. When NRNE = 2, the exclusions of the C4- and the O4-atom in glucose unit M-1 must precede those of the atoms of glucose unit M, etc.

- The option NRNE = 3 has not been implemented for glucose units.

- Other types of linkage, like 1-3 or 1-2, follow the same rules.

An example of a residue topology building block containing a glucose unit is the sugar building block to be found under the name GLCA in the RT37*.DAT files.

3.5.6 Other linear chain building blocks

From the previous paragraphs it has become clear that program PROGMT may link any type of residue topology building blocks into a linear covalently connected chain, as long as the characteristics of the link satisfies the rules that govern the peptide C-N links between amino acid residues.

A Styrene residue topology building block may serve as an example.
By choosing the displayed sequence of atoms and giving the first atom the name CA, all the rules for connecting these building blocks into a polystyrene chain using PROGMT are satisfied. A few comments are made.

- Any other type of sidechain can be handled, as long as the main chain atom CA is the first atom of the building block and the main chain atom CB is the last one.

- The rules for listing the bonds, bond-angles, improper dihedrals and dihedrals angles involving the CB-CA link are identical to those for the C-N link between residues.

- The rules for specifying excluded neighbours are also identical. When NRNE = 2, the exclusions of the second last atom (no exclusions) and of the last atom (CB) in styrene unit M-1 must precede those of the atoms of styrene unit M, etc.

- The option NRNE = 3 should be used with care.

- It should always be checked in the complete molecular topology generated from the building blocks by program PROGMT whether the linking has correctly been performed.
3.6 Interaction Function Parameters

3.6.1 Introduction

The molecular topology file of a molecular system does not only contain topological information about the system, but also force field parameters. These parameters have been listed in Chapter 3.3.2. They can be directly specified as part of a formatted molecular topology as described in Chapter 3.3.3. For small molecules containing not too many different atom types and force field parameters, this is a feasible procedure. However, when applying an elaborate force field with tens of atom types and hundreds of parameters, this procedure becomes inefficient. An alternative is the following set up. All the force field parameters that belong to a specific force field are kept in two different files. The force field parameters that are related to the molecular topology, like atomic charges and third or excluded nearest neighbour information, are included in the residue topology building block file, which has been described in Chapter 3.5. The remaining force field parameters, which are independent of the molecular topology, are kept in another file, the interaction function parameter file. Both files are used by program PROGMT to generate a complete molecular topology file (Chapter 3.3.2) corresponding to the molecular system that is considered.

Various aspects of storing force field or interaction function parameters will be discussed in Chapters 3.6.2-3.6.5. The basic information concerning the contents and format of an interaction function parameter file can be found in the heading comment of program PROGMT (Chapter 16).
3.6.2 Bond-stretch, bond-angle bending and (improper) torsional interaction parameters

The parameters concerning the bond-stretching, bond-angle bending and (improper) dihedral torsional terms in the interaction function are kept in an interaction function parameter file. The contents of this file is described in the heading comment of program PROGNT (Chapter 16) under TAPE12. Here, a few comments are given.

- The file begins with defining atom types (TYPE, REAL*8) in terms of integer atom codes and atomic masses WASS for the different atom types (number = NRATT). TITLE arrays (REAL*8) can be used to store comments. This part of the file must be identical to the first part of the corresponding residue topology (building block) file (Chapter 3.5) as far as the definition of integer atom codes and masses is concerned.
- The number of bond types (NRBT = NBTY), bond-angle types (NRBAT = NTY), improper dihedral types (NRIDAT = NQTY) and of dihedral angles types (NRDAT = NPTY) is to be specified.

- When a force field contains a special hydrogen bonding interaction function term, the number of types of hydrogen bonds (NRHBT) must be specified. The GROMOS force field does not contain such a term, so NRHBT = 0 in the GROMOS interaction function parameter files.

- NTRIP is a variable required for connecting the bond-angle sequential code to the integer atom codes (atom types) of the atoms forming a bond angle. It must be chosen such that NTRIP .GE. NRATT*(NRATT+1)/2.

- NTRNB is a control parameter for reading van der Waals interaction parameters, which will be described in the next paragraph.

- For each bond type (number = NRBT) the following data are read.
  Sequence number N which will be taken to be the sequential bond code, the integer atom codes of the atoms I and J forming bond I-J and the force constant CB and the bond-length at minimum energy BO of the harmonic bond-stretching interaction term.

- For each bond-angle type (number = NRBAT) the following data are read.
  Sequence number N which will be taken to be the sequential bond-angle code, the integer atom codes of the atoms I, J and K forming bond angle I-J-K and the force constant CT and the bond-angle at minimum energy TO of the harmonic bond-angle bending interaction term.

- For each improper dihedral angle type (number = NRIDAT) the following data are read.
  Sequence number N which will be taken to be the sequential improper dihedral code, the integer atom codes of the atoms I and L involved in improper dihedral I-J-K-L and the force constant CQ and the improper dihedral at minimum energy QO of the harmonic improper dihedral angle.
torsion interaction term.

- For each dihedral angle type (number = NRDAT) the following data are read.
  Sequence number N which will be taken to be the sequential dihedral code,
  the integer atom codes of the atoms J and K involved in dihedral I-J-K-L
  and the force constant CP, the phase shift PD and the multiplicity NP of
  the sinusoidal dihedral angle torsion interaction term. If there is more
  than one dihedral type belonging to the atom types (integer atom codes) of
  the two central atoms J and K, the variable NNH is used to specify which
  one will be taken as default (NNH = 0).

3.6.3 Van der Waals interaction parameters and integer atom codes

The last part of an interaction function parameter file contains
information on the van der Waals interaction parameters. These are C12
(I,J), the coefficient of the 1/r**12 term and C6(I,J) the coefficient of
the -1/r**6 term in the non-bonded interaction. These coefficients depend
on the integer atom codes (1..NRATT) of atoms I and J. In a formatted
molecular topology file these parameters are stored in the arrays C12,
C6[1..NRATT*(NRATT+1)/2]. The corresponding parameters for the 1-4 or third
neighbour non-bonded interaction are stored in arrays CS12,
CS6[1..NRATT*(NRATT+1)/2].

On the interaction function parameter file the information on the van
der Waals parameters is stored in a different, restricted way. This is
specified in the heading comment of program PROGTM under TAPE12.

1. If NTRNB = 0, for each integer atom code N (N = 1..NRATT) the following
data are read.
  Sequence number N which must equal the integer atom code, the atom type
TYPE[N](REAL*8), the atomic polarisability POL[N], the number of
effective outer shell electrons EFE[N] and the atomic van der Waals
radius RVDW[N]. Using the Slater-Kirkwood formula [MD82,8] the van der
Waals parameters C12(N,M) and C6(N,M) are calculated in program PROGMT.

2. If NTRNB = 1 or 2, for each integer atom code N (N = 1, ..NRATT) the
following data are read. Sequence number N which must equal the integer
atom code, the atom type TYPE[N](REAL*8), the value C6^{1/2}(N) - (POL[N]),
three values of C12^{1/2}(N) (BPAIR[N,1..3]) and a pointer array
LPAIR[N,1..NRATT] which contains an index with value 1, 2 or 3 by which
one of the three values of C12^{1/2}(N) in BPAIR[N,1..3] is selected for
each integer atom code pair (N,M) with M = 1, NRATT. The C12(N,M)
parameter is obtained by multiplication of the selected C12^{1/2}(N) and
C12^{1/2}(M) values. Likewise, the C6(N,M) parameter is obtained by
multiplication of the C6^{1/2}(N) and C6^{1/2}(M) values. This uncommon
representation of information on the van der Waals C12 parameters is due
to the development of the GROMOS force field parameters, as discussed in
Chapter 2.7.

3. If NTRNB = 2, for each integer atom code N (N = 1, ..NRATT) the C6^{1/2}(N)
and C12^{1/2}(N) values that determine the 1-4 or third neighbour
interaction are read. The CS6 and CS12 arrays containing the C6(N,M) and
C12(N,M) 1-4 parameters are calculated as C6^{1/2}[N]*C6^{1/2}[M] and
C12^{1/2}[N]*C12^{1/2}[M].

When a force field contains a special hydrogen bonding interaction
function term consisting of a 1/r**12 repulsion and a 1/r**10 attraction,
the coefficients of these terms can be calculated when the distance at
minimum energy RMIN and the corresponding minimum energy VMIN are given.
These data may be specified at the rear of the interaction function
parameter file. However, the option controlling the special hydrogen
bonding interaction term has not been maintained in recent versions of
GROMOS, since the GROMOS force fields do not contain such interaction terms.

3.6.4 Atomic charges and charge group codes

The atomic charges and the charge group codes are to be specified
with the atoms of residue topology building blocks, in the residue topology
file. This is discussed in Chapter 3.5.

3.6.5 Excluded neighbours

The information about which atoms J will be excluded from non-bonded
interaction with atom I based on the proximity of atoms I and J measured
along the covalently bound chain (nearest neighbours), is to be specified
with the atomic information in the residue topology building blocks in the
residue topology file. This is discussed in Chapter 3.5.
3.7 Other types of data

Apart from the basic types of data files described in the previous paragraphs, there are a few other types of data files which are generally less frequently used. These are discussed below.

3.7.1 Atom-atom distance restraints

When performing energy minimisation or a MD or SD simulation a special term like an atom-atom distance restraining term, can be added to the interaction function. As discussed in Chapter 2.8.3-4, such a term may be used to make a molecule satisfy a given set of atom-atom distance constraints. Here, the structure of a data file containing atom-atom distance restraints is described. A slight complication is that an atom involved in an atom-atom distance restraint pair may be a virtual or pseudo atom (Chapter 2.8.4). In terms of the molecular topology file or an atomic coordinate file such an atom is non-existing. As discussed in Chapter 2.8.4, its geometric position is defined in terms of the positions of its non-virtual neighbour atoms. For a virtual or pseudo atom the atom-atom distance restraint file will contain the atom sequence number of the real atoms that define the virtual or pseudo atom position together with a code denoting the specific geometrical definition.

The heading comments of subroutine DISRE and programs PROEM, PRODR, PROMD and PROSD contain information about the structure of an atom-atom distance restraint file. As usual it begins with a TITLE-record. Subsequently the data are read as follows.

```
READ (unit, 12) NDR
DO 10 N = 1, NDR
```
10 READ (unit, 12) IDR1[N], JDR1[N], KDR1[N], LDR1[N], ICDR1[N],
    IDR2[N], JDR2[N], KDR2[N], LDR2[N], ICDR2[N],
    RO[N], W0[N]
12 FORMAT (5I5, 5X, 5I5, 2F10.5)

Here, NDR denotes the number of atom-atom distance restraint pairs per
"solute" molecule. The two real, virtual or pseudo atoms forming a
restraint atom pair are denoted as atom 1 and atom 2. If atom 1 is a real
atom, its atom sequence number is specified in IDR1 and its code ICDR1 = 0
(JDR1, KDR1, LDR1 are not used). Likewise, when atom 2 is a real one: IDR2
= atom sequence number and ICDR2 = 0. When atom 1 or 2 is a virtual or
pseudo atom its code ICDR .NE. 0 and the atom sequence numbers IDR, JDR, KDR
and LDR of the real neighbour atoms I, J, K and L that define the position
of atom 1 or 2 must be specified. As discussed in Chapter 2.7.4 the allowed
geometries are the following ones. The notation is given in terms of
hydrogen atoms.

ICDR = 0: real H-atom; its atom sequence number = IDR.

ICDR = 1: virtual H-atom, aliphatic CH₃; it is bound to real atom I
(carbon) and the three covalently bound real neighbours
of atom I are the real atoms J, K and L.

ICDR = 2: virtual H-atom, aromatic CH₂; it is bound to real atom I
(carbon) and the two covalently bound real neighbours of
atom I are the real atoms J and K (LDR is not used).

ICDR = 3: pseudo H-atom, geometric mean of the two H-atoms of an
aliphatic CH₂; it is (pseudo) bound to real atom I
(carbon) and the two covalently bound real neighbours of
atom I are the real atoms J and K (LDR is not used).

ICDR = 4: virtual H-atom, one of the two H-atoms of an aliphatic
CH₂; it is bound to real atom I (carbon) and the two
covalently bound real neighbours of atom I are the real
atoms J and K (LDR is not used); the definition is the
following: looking along covalent bond vector J-I from atom
J to the central (carbon) atom I, the direction of the
virtual bond I-H is obtained from the direction of the
bond I-K by a counter-clockwise rotation over 120° around
bond J-I; the other virtual H-atom can be obtained by
interchanging the sequence numbers JDR and KDR.

ICDR = 5: pseudo H-atom, geometric mean of the three H-atoms of a CH₃
group; it is (pseudo) bound to real atom I (carbon) and the
one covalently bound real neighbour of atom I is the real
atom J (KDR and LDR are not used).

ICDR = 6: pseudo H-atom, geometric mean of the six H-atoms of two CH₃
groups that are both bound to a common third carbon atom; it
is (pseudo) bound to this real third carbon atom I and the
carbon atoms of the two CH₃ group are the real atoms J and K
(LDR is not used).

The restraint atom-atom distance beyond which the restraining forces become
non-zero is specified as R0. If R0 > 0, attractive distance restraining is
applied. If R0 < 0, repulsive distance restraining is applied. Optionally,
each atom-atom distance restraint pair can be given an individual weight-
factor W₀ by which the distance restraint interaction term is multiplied.
Application of the weight-factors is controlled by switch NTDR in programs PROEM, PRODR, PROMD and PROSD. If NTDR = 0 or 1, W0-values do not need to be specified.

Reading of an atom-atom distance restraint file occurs in programs:

PROEM, PRODR, PROMD, PROSD

Examples of an atom-atom distance restraint file are named:

***DR*.DAT

3.7.2 Internal coordinates

When averaging internal coordinates, like bond-lengths, bond-angles or (improper) dihedrals using program PROAVQ, these averaged quantities can be stored in a file, which may subsequently be read by program PROCOQ in order to compare them to corresponding quantities obtained from other molecular configurations.

Internal coordinates AQ[1..6, 1..NRQ, 1], energies AQ[1..6, 1..NRQ, 2] and total energies AE[1..6] are written in binary form by the statements:
WRITE (unit) (((AQ[K1, K2, K3], K1 = 1,6), K2 = 1, NRQ), K3 = 1,2).
WRITE (unit) (AE(K), K = 1,6).

This occurs in program:

PROAVQ

The internal quantity Q is selected by switch NTQ in program PROAVQ. It may be the molecular bond-lengths (NTQ = 1: H-atoms; NTQ = 2: no H-atoms), bond-angles (NTQ = 3: H-atoms; NTQ = 4: no H-atoms), improper dihedral angles (NTQ = 5: H-atoms; NTQ = 6: no H-atoms), or dihedral angles (NTQ = 7: H-atoms; NTQ = 8: no H-atoms). The number of quantities Q in the molecule is denoted by NRQ. When K3 = 1, the first index K1 denotes \( <Q>, \langle Q^{**2}\rangle, \langle Q^{**3}\rangle, \langle Q^{**4}\rangle \), minimum value of Q and the maximum value of Q for the individual Q-distributions. Averaging is denoted by the symbol \( <\ldots> \). When K3 = 2, index K1 denotes the corresponding energies, \( \langle E(Q)\rangle, \langle E(Q)^{**2}\rangle, \) etc. The corresponding values of the total molecular energy summed over all NRQ quantities Q are specified in AE.

Reading of internal coordinates is done correspondingly.

This occurs in program

PROCOQ
3.7.3 **Single quantity trajectories**

Program PROTCF may calculate from a trajectory file containing a time series of atomic coordinates or velocities, a time series and time correlation functions for a whole variety of quantities $Q$ which are defined in terms of atomic coordinates or velocities. Such a quantity $Q$ may be a dihedral angle, a vector defined by two atoms in the molecule, an inner product of such vectors, etc. The possible choices of $Q$ are listed in the heading comment of program PROTCF (Chapter 16) and are discussed in Chapter 5.19. The time series or trajectory of the selected quantity $Q$ can be stored in a file, which may subsequently be read by a plotting program or by PROTCF itself in order to recalculate e.g. the time correlation function.

A single quantity trajectory $Q[1..NRQV, 1..NTQ6, 1..NTVC]$ is written in binary form using the statement:

```
WRITE (unit (((Q[K1,K2,K3], K1 = 1, NRQV), K2 = 1, NTQ6), K3 = NTVC))
```

This occurs in program:

```
PROTCF
```

The quantity $Q$ is selected by the switches NTQ1, NTQ2, NTQ3, NTQ4 and NTQ5 in program PROTCF. The number of time points in the time series of $Q$-values is denoted by NRQV. If NTQ6 = 1, one time series of $Q$-values is stored. If NTQ6 = 2, a time series of $Q_1$-values and a time series of $Q_2$-values is stored in order to allow for calculation of a cross correlation function.
If NTVC = 1, Q is a scalar quantity. If NTVC = 3, the three values of index K3 denote the x-, y- and z-components of the vector quantity Q.

Reading of a single quantity trajectory is done correspondingly.

This occurs in program:

```
PROTCF
```

3.7.4 Solute constraints

When applying bond-length constraints to the molecules in a molecular system, the fixed lengths to which the different bonds are constrained are taken from the ideal bond-length array B0 in the molecular topology file using the sequential bond codes ICBH and ICB for the bonds occurring in the "solute". The contents of these variable arrays are specified in Chapter 3.3.2. In old versions of GROMOS the constraint lengths could also be specified on a special solute constraint file. Although this option is being phased out, it has not yet been deleted from GROMOS. Therefore, it will be briefly described here.

A solute constraint file can be read in binary form using the statement:

```
READ (unit) CONP, ICOG, JCOG, NCONG
```

NCONG = the number of solute constraints N.

ICOG[N], JCOG[N] = atom sequence numbers of the solute atoms I and J forming
constraint N

\[ \text{CONP}[N] = \text{square of the length of constraint } N \]

In programs PROEM, PROMD and PROSD the length of the arrays ICOG, JCOG and CONP is denoted by MAXCON.

---

**Reading of a solute constraint file occurs in programs:**

PROEM, PROMD, PROSD

---

3.7.5 Dihedral restraints

When performing an energy minimisation or a MD or SD simulation, a dihedral restraining term can be added to the interaction function. As discussed in Chapter 2.8.6, such a term may be used to make molecule satisfy a given set of dihedral angle constraints. Here, the structure of a data file containing restrained dihedrals is described.

The heading comments of programs PROEM, PROMD and PROSD contain information about the structure of a restrained dihedrals file. As usual it begins with a TITLE-record. Subsequently the data are read as follows.

```
READ (unit, 12) NDLR
DO 10 N = 1, NDLR
10 READ (unit, 12) IPLR[N], JPLR[N], KPLR[N], LPLR[N], CPLR[N], PDLR[N], NPLR[N]
12 FORMAT (4I5, 2F10.5, I5)
```
Here, NDLR denotes the number of restrained dihedrals. A dihedral $i-j-k-l$
is defined by the atom sequence numbers IPLR, JPLR, KPLR and LPLR. The
interaction parameters $K^{dh}$, $\delta$ and $n$ in formula (2.8.5.1) are denoted by
CPLR, PDLR and NCLR. Application of dihedral restraining is controlled by
switch NTDLR in programs PROEM, PROMD and PROSD.

Reading of a restrained dihedrals file occurs in programs:

PROEM, PROMD, PROSD
4. Standard GROMOS files

The GROMOS program package comes with a number of standard data files. These fall into two categories. Residue topology files and interaction function parameter files containing various versions of the GROMOS force fields, which are listed in Chapter 4.1, and secondly, standard configurations of molecules or molecular systems which are required by programs like PROBOX and PROSSC. These are listed in Chapter 4.3. Chapter 4.2 contains a brief list of the residue topology building blocks that are present in the GROMOS residue topology files RT37*.DAT.

4.1 GROMOS force field files

Each version of the GROMOS force field consists of one residue topology (building block) file named RT***.DAT and one interaction function parameter file named IFP***.DAT. The GROMOS force field is continuously being tested and improved. From time to time a new version is brought out. However, old force field versions are kept with the new versions of the GROMOS package in order to allow for the analysis of trajectories produced with old force field versions using new programs.

Currently there exist five versions of the GROMOS force field. The C-versions are the basic force fields designed for molecules in solution or in crystalline form. The D-versions are derived from the C-versions in order to be used for simulating molecules in vacuo. The atomic charges and van der Waals parameters are changed such that charged atom groups are neutralized while maintaining the hydrogen-bonding capacity of the individual atoms.

The five versions are:
- the 26C1 force field of June 1981
- the 37C2 force field of January 1983 and extended in November 1983
- the 37D2 force field of January 1983 and extended in November 1983
- the 37C4 force field of November 1983 and revised in December 1985
- the 37D4 force field of November 1983 and revised in December 1985

The first three ones are old versions which should not be used. The 37D4 version is meant for use in vacuo. The 37C4 version for all other cases.

4.1.1 The 26C1 force field

This is the first GROMOS force field, consisting of files RT26C.DAT and IFP26C1.DAT. Only 26 atom types were defined. The residue topology file RT26C.DAT contains only amino acid residues and a heme group. It was meant for simulation of proteins in aqueous solution or crystalline form.

4.1.2 The 37C2 force field

This is an extension of the 26C1 force field in order to allow for simulation of nucleotides, sugars, etc. Eleven new atom types were added. Some interaction function parameters were slightly changed, which made the version number change from 1 to 2. It consists of files RT37C.DAT and IFP37C2.DAT. The residue topology file RT37C.DAT contains many more building blocks. It was meant for simulation of proteins, DNA, sugars in aqueous solution or crystalline form.
4.1.3 The 37D2 force field

This version is the one corresponding to the 37C2 one, but adapted in order to be used for simulations of molecules in vacuo. It consists of files RT37D.DAT and IP37D2.DAT.

4.1.4 The 37C4 force field

In the previous force fields the repulsive part of the van der Waals interaction between third neighbour atoms (1-4 interaction) was too large, in case one or both of the atoms involved was an extended (CH1, CH2, CH3, CR51, CR61, CS1, CS2) carbon atom and the torsion angle 1-2-3-4 was in a cis-conformation. This effect has been redressed by changing the programs such that for 1-4 or third neighbour interactions van der Waals parameters can be used, which are different from the normal ones. These extra 1-4 van der Waals parameters had to be given on the interaction function parameter file. This change made the force field version number change from 2 to 4. The force field consists of files RT37C.DAT and IP37C4.DAT. It is meant for simulation of proteins, DNA, sugars, etc. in solution or crystalline form.

4.1.5 The 37D4 force field

This version is the one corresponding to the 37C4 one, but adapted in order to be used for simulations of molecules in vacuo. It consists of files RT37D.DAT and IP37D4.DAT.
4.2 GROMOS residue topology building blocks

The GROMOS residue topology building block files RT***.DAT contain the building blocks for a number of important types of molecules, such as proteins, DNA, RNA, sugars, etc. Below we list for each building block of the files RT37*.DAT the building block name, the residue code, and a description of the residue, nucleotide, glucose unit or molecule it is representing.

<table>
<thead>
<tr>
<th>Name</th>
<th>code</th>
<th>description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ALA</td>
<td>1</td>
<td>amino acid residue (L) Alanine</td>
</tr>
<tr>
<td>ARG</td>
<td>2</td>
<td>Arginine (protonated; charge+1)</td>
</tr>
<tr>
<td>ARGN</td>
<td>70</td>
<td>Arginine (deprotonated; neutral)</td>
</tr>
<tr>
<td>ASN</td>
<td>3</td>
<td>Asparagine</td>
</tr>
<tr>
<td>ASNI</td>
<td>31</td>
<td>Asparagine (coordinated withZN)</td>
</tr>
<tr>
<td>ASP</td>
<td>4</td>
<td>Aspartic acid (deprotonated; charge=1)</td>
</tr>
<tr>
<td>ASPH</td>
<td>5</td>
<td>Aspartic acid (protonated; neutral)</td>
</tr>
<tr>
<td>CYS</td>
<td>6</td>
<td>Cysteine (deprotonated; charge=1/2)</td>
</tr>
<tr>
<td>CYSH</td>
<td>7</td>
<td>Cysteine (protonated; neutral)</td>
</tr>
<tr>
<td>CYS1</td>
<td>8</td>
<td>Cysteine (1st member of S-S bridge)</td>
</tr>
<tr>
<td>CYS2</td>
<td>9</td>
<td>Cysteine (2nd member of S-S bridge)</td>
</tr>
<tr>
<td>GLN</td>
<td>10</td>
<td>Glutamine</td>
</tr>
<tr>
<td>GLU</td>
<td>11</td>
<td>Glutamic acid (deprotonated; charge=1)</td>
</tr>
<tr>
<td>GLUH</td>
<td>12</td>
<td>Glutamic acid (protonated; neutral)</td>
</tr>
<tr>
<td>GLY</td>
<td>13</td>
<td>Glycine</td>
</tr>
<tr>
<td>HISA</td>
<td>14</td>
<td>Histidine (protonated at ND1; neutral)</td>
</tr>
<tr>
<td>HISH</td>
<td>15</td>
<td>Histidine (protonated at NE2; neutral)</td>
</tr>
<tr>
<td>HISH</td>
<td>16</td>
<td>Histidine (protonated at ND1 and NE2; charge+1)</td>
</tr>
</tbody>
</table>
HIS  30  Histidine (coupled to HEME at NE2; neutral)
HYP  17  Hydroxyproline
ILE  18  Isoleucine
LEU  19  Leucine (CD1 and CD2 reverse of IUPAC-IUB)
LYS  20  Lysine (deprotonated; neutral)
LYSH 21  Lysine (protonated; charge +1)
MET  22  Methionine
PHE  23  Phenylalanine
PRO  24  Proline
SER  25  Serine
THR  26  Threonine
TRP  27  Tryptophan
TYR  28  Tyrosine
VAL  29  Valine (CG1 and CG2 reverse of IUPAC-IUB)

ABU  74  L-2-amino-butanolic acid
MERMT 75  (4R)-4-[(E)-2-butanyl]-4-N-dimethyl-L-threonine
MELEU 73  N-methyl-L-leucine (CD1 and CD2; CG1 and CG2 reverse of IUPAC-IUB)
MEVAL 72  N-methyl-L-valine of IUPAC-IUB
SAR  71  Sarcosine or N-methylglycine
<table>
<thead>
<tr>
<th>Abbreviation</th>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>DADE</td>
<td>nucleotide 2'-deoxyadenosine 5'-phosphoric acid</td>
<td></td>
</tr>
<tr>
<td>DQUA</td>
<td>2'-deoxyguanosine 5'-phosphoric acid (DNA; charge-1)</td>
<td></td>
</tr>
<tr>
<td>DCYT</td>
<td>2'-deoxycytidine 5'-phosphoric acid charge-1)</td>
<td></td>
</tr>
<tr>
<td>DTHY</td>
<td>2'-deoxythymidine 5'-phosphoric acid</td>
<td></td>
</tr>
<tr>
<td>ADE</td>
<td>adenosine 5'-phosphoric acid</td>
<td></td>
</tr>
<tr>
<td>GUA</td>
<td>guanosine 5'-phosphoric acid (RNA; charge-1)</td>
<td></td>
</tr>
<tr>
<td>CYT</td>
<td>cytidine 5'-phosphoric acid</td>
<td></td>
</tr>
<tr>
<td>URA</td>
<td>uridine 5'-phosphoric acid</td>
<td></td>
</tr>
<tr>
<td>HEME</td>
<td>heme group (charge-2, acidic groups deprotonated)</td>
<td></td>
</tr>
<tr>
<td>FMNO</td>
<td>flavin mononucleotide (oxidized, deprotonated at FN5 and FN1; charge-1, [\text{OPO}_2\text{H}^-])</td>
<td></td>
</tr>
<tr>
<td>FMNS</td>
<td>flavin mononucleotide (semi-reduced, protonated at FN5; charge-1, [\text{OPO}_2\text{H}^-])</td>
<td></td>
</tr>
<tr>
<td>FMNR</td>
<td>flavin mononucleotide (reduced, protonated at FN5 and FN1; charge-1, [\text{OPO}_2\text{H}^-])</td>
<td></td>
</tr>
<tr>
<td>PFN</td>
<td>proflavin (protonated at FN5; charge+1)</td>
<td></td>
</tr>
<tr>
<td>NADP</td>
<td>nicotinamide adenine dinucleotide (NAD+; charge-1)</td>
<td></td>
</tr>
<tr>
<td>NADH</td>
<td>nicotinamide adenine dinucleotide (NADH; charge-2)</td>
<td></td>
</tr>
<tr>
<td>NDPH</td>
<td>nicotinamide adenine dinucleotide phosphate (NADPH; charge-3; [\text{OPO}_2\text{H}^-])</td>
<td></td>
</tr>
<tr>
<td>NDPP</td>
<td>nicotinamide adenine dinucleotide phosphate (NADP+; charge-2, [\text{OPO}_2\text{H}^-])</td>
<td></td>
</tr>
<tr>
<td>NDPHN</td>
<td>nicotinamide adenine dinucleotide phosphate (NADPH; neutral, [\text{OPO}(\text{OH})_2])</td>
<td></td>
</tr>
<tr>
<td>Symbol</td>
<td>Charge</td>
<td>Description</td>
</tr>
<tr>
<td>--------</td>
<td>--------</td>
<td>-------------</td>
</tr>
<tr>
<td>CYT*</td>
<td>50</td>
<td>3',5'-O-(tetral isopropyl-1,3-dioisoxanediyl) cytidine (neutral)</td>
</tr>
<tr>
<td>MTXH</td>
<td>52</td>
<td>methotrexate (protonated at N1; charge-2)</td>
</tr>
<tr>
<td>FOL</td>
<td>53</td>
<td>folate (charge-2)</td>
</tr>
<tr>
<td>DHF</td>
<td>54</td>
<td>7,8-dihydrofolate (charge-2)</td>
</tr>
<tr>
<td>THF</td>
<td>55</td>
<td>5,6,7,8-tetrahydrofolate (charge-2)</td>
</tr>
<tr>
<td>TMP</td>
<td>56</td>
<td>trimethoprim (deprotonated at N1; neutral)</td>
</tr>
<tr>
<td>TMPH</td>
<td>57</td>
<td>trimethoprim (protonated at N1; neutral)</td>
</tr>
<tr>
<td>TMHPH</td>
<td>58</td>
<td>trimethoprim (protonated at N1; charge+1)</td>
</tr>
<tr>
<td>PDG</td>
<td>61</td>
<td>3-phospho-D-glycerate (charge-2)</td>
</tr>
<tr>
<td>ATP</td>
<td>62</td>
<td>adenosine 5'-triphosphate (ATP; charge-3)</td>
</tr>
<tr>
<td>PMB</td>
<td>66</td>
<td>p-methylbenzyl alcoholate (charge-1)</td>
</tr>
<tr>
<td>PMBH</td>
<td>67</td>
<td>p-methylbenzyl alcohol (neutral)</td>
</tr>
<tr>
<td>RTOL</td>
<td>69</td>
<td>retinol (neutral)</td>
</tr>
<tr>
<td>GLCA</td>
<td>81</td>
<td>α-D-glycopyranosyl (1-4 linkage; neutral)</td>
</tr>
<tr>
<td>GLCB</td>
<td>84</td>
<td>β-D-glycopyranosyl (1-2 linkage; neutral)</td>
</tr>
<tr>
<td>GALB</td>
<td>85</td>
<td>β-D-galactopyranosyl (1-3 linkage; neutral)</td>
</tr>
<tr>
<td>TEMP</td>
<td>86</td>
<td>tetramethyl pyrrolinyl (nitroxide spin label; neutral)</td>
</tr>
<tr>
<td>DMSO</td>
<td>53</td>
<td>dimethylsulfoxide (neutral)</td>
</tr>
<tr>
<td>ETH</td>
<td>64</td>
<td>ethyl alcoholate (charge-1)</td>
</tr>
<tr>
<td>ETHH</td>
<td>65</td>
<td>ethyl alcohol (neutral)</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>-----</td>
<td>----</td>
<td>------------------</td>
</tr>
<tr>
<td>MTH</td>
<td>83</td>
<td>methanol (neutral)</td>
</tr>
<tr>
<td>H2O</td>
<td>82</td>
<td>water (neutral)</td>
</tr>
<tr>
<td>SO4</td>
<td>68</td>
<td>$\text{SO}_4^{2-}$ ion (charge=2)</td>
</tr>
<tr>
<td>ZN</td>
<td>76</td>
<td>zinc ion (charge=2)</td>
</tr>
<tr>
<td>NA</td>
<td>77</td>
<td>sodium ion (charge=1)</td>
</tr>
<tr>
<td>CL</td>
<td>78</td>
<td>chlorine ion (charge=1)</td>
</tr>
<tr>
<td>CA</td>
<td>79</td>
<td>calcium ion (charge=2)</td>
</tr>
<tr>
<td>MG</td>
<td>80</td>
<td>magnesium ion (charge=2)</td>
</tr>
</tbody>
</table>

We note the following,

- **IUPAC-IUB nomenclature** has been used throughout, except for the side-chain atoms of LEU, MELEU (CD1 and CD2) and VAL, MEVAL (CG1 and CG2), the names of which have unfortunately been interchanged in the GROMOS residue topology building blocks (RT***.DAT files).
- Brookhaven protein data bank nomenclature has been used when no IUPAC-IUB rules were defined (HEME group).
- The contents of the residue topology building block file is listed in a more readable form in the output of program PROGMT, see files OUTGMT***.LIS.

### 4.3 GROMOS standard configurations

The GROMOS package comes with a standard configuration of a box with water molecules and with a file containing standard **amino acid** configurations.

#### 4.3.1 Water
When simulating a molecule in solution or in crystalline form, the initial positions of solvent molecules surrounding it are to be generated in some way. This can be done using program PROBOX, which requires a standard GROMOS atomic coordinate file containing a number of solvent molecules.

The file SPC216.DAT contains an equilibrium (T = 300 K) configuration of 216 SPC (simple point charge) water molecules written in standard formatted form. The edges of the cubic periodic box have the length 1.86206 nm. The atomic coordinates are in nm.

4.3.2 Amino acids

Sometimes the crystallographic coordinates of a protein are incomplete. Coordinates of mobile side-chain atoms may be missing. In that case initial positions for these atoms need to be generated in some way. This can be done using program PROSSC, which requires an atomic coordinate file containing standard configurations of amino acid residues.

The file STRESC.DAT contains standard configurations for 20 amino acid residues, without hydrogen coordinates. The atomic coordinates are in Angstroms.
5. GROMOS library architecture

5.1 Introduction

GROMOS has been set up using a modular architecture. The modules, viz.
programs and subroutines are generally independent of each other. This
modular architecture renders the possibility of using only specific parts of
the library, or of replacing parts (modules) by user supplied subroutines.

GROMOS presently consists of about 100 modules that must be combined by
the user to fit the task he wants to perform. Chapter 1.2 gives an overview
and list of GROMOS programs. There are six types of GROMOS programs.

Programs that

1. build a molecular topology (MT)
2. transform a given atom coordinate sequence and format to the atom
   sequence in the molecular topology and to the GROMOS standard coordinate
   format
3. generate atom coordinates
4. minimize or perform simulations
5. analyse configurations, or sequences of configurations
6. merge or reduce coordinate files or transform atom coordinates to a
   special format (interfacing).
The relation between the different types of programs is shown below

Artbitrary atom coordinate sequence and format

MOL. TOP. builders
PRORMT: from scratch
PRORMT: from building blocks
PRORMT: merging
PRORMT: reduction + format

ATOMIC COORD. resequencers
PROBRX: special cases
PROCS1: most powerful

MOL. TOP. files
MT****.BIN
MT****.FMT

ATOMIC COORD. files
***X***.DAT

ATOMIC COORD. generators
PROCGA: Cartesian from internal
PROSSC: substitute side chains
PROGCH: polar hydrogens (solute)
PROGWH: polar hydrogens (water)
PROCRY: crystal symmetry transformation
PROBOX: immersing solute in solvent
PROION: solvent-ion replacement

ATOMIC COORD. files
***X***.DAT
Minimizers and simulators

PROEM: energy minimization
PROMD: molecular dynamics
PROSD: stochastic dynamics

SINGLE CONFIG. files
***x***.DAT

TRAJECTORY files
***RMD*.DAT
***RSD*.DAT

Analysers

PROAVX: averages solute positions
PROAVQ: averages solute internal coords.
PROAJC: J-coupling const.
PRODR: dist. restraints
PROAVN: number of neighbours
PROAHB: hydrogen bonds
PROMHB: monitors
PROCOC: averages site occupancies
PROAVS: solvent positions
PROCOX: compares solute positions
PROCOQ: int. coords.
PROCOB: isot. B-factors
PROCAB: anisot.
PROCOD: distances
PRONBL: prints neighbour list
PROCHBB: hydrogen bonds
PROCPS: solute-solv. neighbours
PROCOS: compares solvent positions
PROTCF: time correlation functions
The abilities of the six types of programs will be briefly discussed in Chapters 5.2-7. Examples of their application will be given in Chapter 7.
7. GROMOS program descriptions with tutorial examples

7.1 Introduction and overview of the examples

GROMOS is delivered with a number of tutorial examples, at least one for each program. Four different molecules are used as examples:
- the penta-peptide Val-Tyr-Arg-Lys-Gln (PCL, PCI, PNC)
- a penta-glycine-peptide (GLY)
- the protein bovine pancreatic trypsin inhibitor (PTI)
- an 8 base pair (CGCAACGC) DNA fragment (DNA)

The molecules are denoted by a three-character code (YYY):

PCL: penta-peptide, charged, linear in water
PCI: penta-peptide, charged, linear, with ions
PNC: penta-peptide, neutral, cyclic (D-Val) in vacuo
GLY: penta-glycine, linear
PTI: protein PTI
DNA: 8 base pair DNA fragment
DNPL: DNA + PCL

In most examples PCL, PCI and PNC are used. The former two are simulated in water and the latter in vacuo.

Below all 53 examples are listed together with their command-, input-, output- and data-files. The structure of the filenames is the following, where the molecular three-character code is denoted by YYY, the three-character program mnemonic is QQQ, and the two-character interaction function code is ZZ (C4 or D4).
- Command file (job control)
- Input file (program control)
- Output file (listing)
- Molecular Topology file (binary)
- Molecular Topology file (formatted)
- Atom coordinate file:
  - single configuration
  - trajectory
  - averaged configuration
- Perturbation potential file
- Atom-atom distance restraints file

1. Examples of molecular topology builders

**Program PROGMT:**

<table>
<thead>
<tr>
<th>Unit</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>5</td>
<td>program control, residue sequence</td>
</tr>
<tr>
<td>11</td>
<td>residue topology building blocks</td>
</tr>
<tr>
<td>12</td>
<td>interaction function parameters</td>
</tr>
<tr>
<td>13</td>
<td>atomic coordinates (optional)</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Unit</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td>generated molecular topology (binary)</td>
</tr>
</tbody>
</table>

**Example PCL:**

<table>
<thead>
<tr>
<th>Command</th>
<th>JGMTPCL.COM</th>
</tr>
</thead>
</table>

<table>
<thead>
<tr>
<th>In:</th>
<th>Unit 5: JGMTPCL.DAT</th>
</tr>
</thead>
<tbody>
<tr>
<td>11</td>
<td>RT37C.DAT</td>
</tr>
<tr>
<td>12</td>
<td>IFP37CH4.DAT</td>
</tr>
<tr>
<td>13</td>
<td>not used</td>
</tr>
</tbody>
</table>

| Out:     | Unit 10: MTPCLCH4.BIN |
Example PCI:  
command:  
output:  
scratch:  
in:  
out:  

Example PNC:  
command:  
output:  
scratch:  
in:  
out:  

Example PTI:  
command:  
output:  
scratch:  
in:  
out:  
Example DNA: command: JGMTDNA.COM
in: unit 5: IGMTDNA.DAT
    11: RT37C.DAT
    12: IFP37C4.DAT
    13: not used
out: unit 10: MTDNAC4.BIN
     output: OUTGMTDNA.LIS
scratch: unit 9: deleted

Program PROMMT: in: unit 5: program control
                   11: first molecular topology (binary)
                   12: second molecular topology (binary)
                   13: atomic coordinates (optional)
out: unit 10: merged molecular topology (binary)
     output:

Example DNP: command: JMMTDNP.COM
in: unit 5: IMMDNP.DAT
    11: MTDNAC4.BIN
    12: MTPCLC4.BIN
    13: not used
out: unit 10: MTDNPC4.BIN (deleted)
     output: OUTMMTDNP.LIS

Program PROCMT: in: unit 5: program control
                   11: initial molecular topology (binary)
12: atom coordinates (for reduction, optional)
13: atom pointer list (for reduction, optional)

out: unit 10: (reduced) molecular topology (binary)
20: (reduced) molecular topology (formatted)
21: atom coordinates of reduced system
22: (new) sequence numbers of shell atoms
23: (original) seq. numbers of selected atoms

output:

Example PNC: command: JCMTTPNC.COM
in: unit 5: ICMTTPNC.DAT
   11: MTPNHC4.BIN
   12: not used
   13: not used

out: unit 10: deleted
   20: MTPNHC4.FMT
   21: not used
   22: not used
   23: not used

output: OUTCMTTPNC.LIS

Program PRORMT: in: unit 5: molecular topology (formatted)

out: unit 20: molecular topology (binary)

output:

Example PNC: command: JRMTTPNC.COM
in: unit 5: IRMTTPNC.DAT

out: unit 20: MTPNCC4.BIN
output: OUTFMTPNC.LIS

Note: The file IHMTPNC.DAT is obtained from the formatted topology
MTPNHC4.FMT by changing Val to D-Val and making the peptide
cyclic.

2. Examples of coordinate resequencers

Program PROBRK: in: unit 5: program control
                    11: atom coordinates (Brookhaven)
                    out: unit 12: resequenced and reformatted coordinates
output:

Example PTI: command: JBRKPTI.COM
in: unit 5: IBRKPTI.DAT
     11: -
     12: -
output: -

Program PROCS1: in: unit 5: program control
                  11: atom coordinates (CA-N-C-O-sidechain)
                  out: unit 12: resequenced coordinates (N-CA-SC-C-O)
output:

Example PCL: command: JCS1PCL.COM
in: unit 5: ICS1PCL.DAT
     11: PCLXTEST0.DAT
     out: unit 12: PCLXTEST1.DAT

output: OUTCS1PCL.LIS

Note: The file PCLXTES0.DAT is obtained from file PCLXMOD2.DAT by randomizing the atom sequence.

Program PROC32: in:
  unit 5: program control, new sequence (optional)
  11: atom coordinates (N-CA-sidechain-C=O)
out:  unit 12: resequenced (sidechain) coordinates
output:

Example PCL: command: JCS2PCL.COM
  in:  unit 5: ICS2PCL.DAT
        11: PCLXTES1.DAT
out:  unit 12: PCLXTES2.DAT
output: OUTCS2PCL.LIS

3. Examples of coordinate generators

Program PROCGA: in:
  unit 5: program control, internal coordinates
  10: molecular topology (binary, optional)
  11: atom coordinates (optional)
out:  unit 21: generated Cartesian coordinates
output:

Example PNG: command: JGCAPNC.COM
  in:  unit 5: IGCAPNC.DAT
        10: MTPNCC4.BIN
        11: not used
Example GLY: command: JGCAGLY.COM
in: unit 5: IGCAGLY.DAT
10: not used
11: not used
out: unit 21: GLYXMOD1.DAT
output: OUTGCAGLY.LIS

Program PROSSC: in: unit 5: program control
11: atom coordinates (arbitrary sc. coords.)
12: standard residue coordinates
out: unit 21: regenerated atom coordinates
output:

Example PCL: command: JSSCPCL.COM
in: unit 5: ISSCPCL.DAT
11: PCLXMOD1.DAT
12: STREC.DAT
out: unit 21: PCLXMOD2.DAT
output: OUTSSCPCL.LIS

Note: The file PCLXMOD1.DAT is obtained from file GLYXMOD1.DAT by inserting empty records for each side-chain atom in PCL.

Program PROGCH: in: unit 5: program control, atom types
10: molecular topology (binary)
11: atom coordinates without polar hydrogens
out:  unit 21: atom coordinates with polar hydrogens
output:

Example PCL: command: JGCHPCL.COM
in:  unit 5: IGCH37.DAT
     10: MTPCLO4.BIN
     11: PCLXMOD2.DAT
out:  unit 21: PCLXMOD3.DAT
     output: OUTGCHPCL.LIS

Program PROGWH: in:  unit 5: program control
out:  unit 11: atom water oxygen coordinates
     unit 21: atom water coordinates (+ H atoms)
     output:

Example PTW: command: JOGWHPTW.COM
in:  unit 5: IGWHTW.DAT
     11: PTWXRV2.DAT
out:  unit 21: PTWXRV3.DAT
     output: OUTGWHPTW.LIS

Note: The file PTWXRV2.DAT is obtained from file PTIXRV3.DAT by selecting the oxygen atoms of the four internal water molecules of PTI.

Program PROCRY: in:  unit 5: program control, symmetry transformations
10: molecular topology (binary)
Example PTI: command: JCRYPTI.COM
in: unit 5: ICYPTI.DAT
10: MPTID4.BIN
11: PTIXRV3.DAT
out: unit 21: PTIXRV4.DAT (deleted)
output: OUTCRYPTI.LIS

Program PROBOX: in: unit 5: program control
10: molecular topology (binary)
11: atom coordinates (solute)
12: standard solvent configuration
out: unit 21: solute plus solvent coordinates
output:

Example PCL: command: JBOXPCL.COM
in: unit 5: IBOXPCL.DAT
10: MTPCLG4.BIN
11: PCLXMOD3.DAT
12: SPC216.DAT
out: unit 21: PCLXMOD5.DAT
output: OUTBOXPCL.LIS

Program PROION: in: unit 5: program control
10: molecular topology (binary)
11: solute plus solvent coordinates
out: unit 21: generated solute, ions, solvent coords.
output:

Example PCL: command: JIONPCL.COM
in: unit 5: IIONPCL.DAT
10: MTPCLC4.BIN
11: PCLXMOD6.DAT
out: unit 21: PCLXMOD7.DAT
output: OUTIONPCL.LIS

Note: The file PCLXMOD6.DAT is obtained by energy minimizing the
configuration PCLXMOD5.DAT

4. Examples of minimizers and simulators

Program PROEM: in: unit 5: program control
20: molecular topology (binary)
21: initial atom coordinates
23: not used
24: restraining reference positions (optional)
25: sequence numbers of restr. atoms (optional)
26: distance restraint atom pairs (optional)
28: restrained dihedral angles (optional)
out: unit 31: final atom coordinates
output:

Example PCL6: command: JEMPCL6.COM
Example PCI:  command:  JEMPPCI.COM

in:  unit 5:  IEMPCLI.DAT
20:  MTPCIC4.BIN
21:  PCLXMOD7.DAT
23:  not used
24:  PCLXMOD5.DAT
25:  PCLXMOD3.DAT
26:  not used
28:  not used

out:  unit 31:  PCIXEM1.DAT
output:  OUTEMPCLI.LIS

Example PNC:  command:  JEMPNC.COM

in:  unit 5:  IEMPNC.DAT
20:  MTPNC4.BIN
21:  PNXMOD1.DAT
23:  not used
24: not used
25: not used
26: PNCDR.DAT
28: not used

out: unit 31: PNCXEM1.DAT
output: OUTMPNC.LIS

Program PROMD: in: unit 5: program control
20: molecular topology (binary)
21: initial atom coordinates
23: not used
24: restraining reference positions (optional)
25: sequence numbers of restr. atoms (optional)
26: distance restraint atom pairs (optional)
27: perturbation of mol. topology (optional)
28: restrained dihedral angles (optional)

out: unit 31: final coordinates, etc.
12: MD trajectory coordinates (optional)
13: MD trajectory velocities (optional)
15: MD trajectory energies (optional)

output:

Example PCII: command: JMDPCI1.COM
in: unit 5: IMDPCI1.DAT
20: MTPCIC4.BIN
21: PCIXEM1.DAT
23: not used
24: not used
25: not used
26: not used
27: not used
28: not used

out:  unit 31: PCIXMD1.DAT
      12: PCIRMD1.DAT
      13: not used
      15: not used

output: OUTMDPCI1.LIS

Note: This is a MD start-up job.

Example PCI2: command: JMDPCI2.COM

in:  unit 5: IMDPCI2.DAT
      20: MTPCIC4.BIN
      21: PCIXMD1.DAT
      23: not used
      24: not used
      25: not used
      26: not used
      27: not used
      28: not used

out:  unit 31: PCIXMD2.DAT
      12: PCIRMD2.DAT
      13: not used
      15: not used

output: OUTMDPCI2.LIS
Note: This is a MD continuation job.

Example PCG1: command: JMDPCG1.COM

in:  unit  5: IMDPCG1.DAT
      20: MTPCIC4.BIN
      21: PCIXEM1.DAT
      23: not used
      24: not used
      25: not used
      26: not used
      27: PCLPER.DAT
      28: not used

out:  unit  31: PCGXMD1.DAT
       12: not used
       13: not used
       15: not used

output: OUTMDPCG1.LIS

Note: This is a MD start-up job.

Example PCG2: command: JMDPCG2.COM

in:  unit  5: IMDPCG2.DAT
      20: MTPCIC4.BIN
      21: PCGXMD2.DAT
      23: not used
      24: not used
      25: not used
      26: not used
27: PCLPER.DAT
28: not used

out: unit 31: PGCXMD2.DAT
12: not used
13: not used
15: not used

output: OUTMDPCG2.LIS

Note: This is a MD continuation job.

Example PNC1: command: JMDPNC1.COM
in: unit 5: IMDPNC1.DAT
20: MTPNCC4.BIN
21: PNCSXEM1.DAT
23: not used
24: not used
25: not used
26: PNCDR.DAT
27: not used
28: not used

out: unit 31: PNCXMD1.DAT
12: PNCRMD1.DAT
13: not used
15: not used

output: OUTMDPNC1.LIS

Note: This is a MD start-up job.
Example PNC2: command: JMDPNC2.COM

in:  unit 5: IMDPNC2.DAT
     20: MTPNC04.BIN
     21: PNCXMD1.DAT
     23: not used
     24: not used
     25: not used
     26: PNCDR.DAT
     27: not used
     28: not used

out: unit 31: PNCXMD2.DAT
     12: not used
     13: not used
     15: not used

output: OUTMDPNC2.LIS

Note: This is a MD continuation job.

Program PROSD: in:  unit 5: program control

20: molecular topology (binary)
21: initial atom coordinates
22: atom friction coefficients (optional)
23: not used
24: restraining reference positions (optional)
25: sequence numbers of restr. atoms (optional)
26: distance restraint atom pairs (optional)
27: perturbation of mol. topology (optional)
28: restrained dihedral angles (optional)
out:  unit 31: final coordinates, etc.
12: SD trajectory coordinates (optional)
13: SD trajectory velocities (optional)
15: SD trajectory energies (optional)

output:

Example PNC1: command: JSDPNC1.COM
in:  unit 5: ISDPNC1.DAT
20: MTPNCC4.BIN
21: PNCXEM1.DAT
22: not used
23: not used
24: not used
25: not used
26: PNCDR.DAT
27: not used
28: not used

out:  unit 31: PNCXSD1.DAT
12: not used
13: not used
15: not used

output: OUTSDPNC1.LIS

Note: This is a SD start-up job.

Example PNC2: command: JSDPNC2.COM
in:  unit 5: ISDPNC2.DAT
20: MTPNCC4.BIN
21: PNCXSD1.DAT
22: not used
23: not used
24: not used
25: not used
26: PNCDR.DAT
27: not used
28: not used
out: unit 31: PNCXSD2.DAT
12: not used
13: not used
15: not used
output: OUTSDPNC2.LIS

Note: This is a SD continuation job.

5. Examples of analysers

Program PROAVX: in: unit 5: program control
10: molecular topology (binary)
11-40: (MD/SD trajectory) coordinates
out: unit 41: averaged coordinates (binary)
42: averaged coordinates (formatted)
output:

Example PCI: command: JAVXPCI.COM
in: unit 5: IAVXPCI.DAT
10: MTPCLC4.BIN
11: PCIRMD1.DAT
12: PCIRMD2.DAT
13-40: not used

out: unit 41: not used (deleted)
42: PCLXAV02.DAT
output: OUTAVXPCI.LIS

Example PNC: command: JAVXPNCC.COM
in: unit 5: IAVXPNCC.DAT
10: MTPNCC4.BIN
11: PNCRMD1.DAT
12-40: not used

out: unit 41: not used (deleted)
42: PNCRXAV01.DAT
output: OUTAVXPNCC.LIS

Program PROAVQ: in: unit 5: program control
10: molecular topology (binary)
11-40: (MD/SD trajectory) coordinates

out: unit 41: averaged quantities (binary)
output:

Example PNC: command: JAVQPNC.COM
in: unit 5: IAVQPNC.DAT
10: MTPNCC4.BIN
11: PNCRMD1.DAT
12-40: not used

out: unit 41: not used (deleted)
output: OUTAVQPNC.LIS

Program PROAJC: in: unit 5: program control
10: molecular topology (binary)
11-40: (MD/SD trajectory) coordinates

out: output:

Example PNC: command: JAJCPNC.COM
in: unit 5: IAJCPNC.DAT
10: MTPNCC4.BIN
11: PNCRMD1.DAT
12-40: not used

out: output: OUTAJCPNC.LIS

Program PRODR: in: unit 5: program control
10: molecular topology (binary)
11-40: (MD/SD trajectory) coordinates
46: distance restraint atom pairs

out: output:

Example PNC: command: JDRPNC.COM
in: unit 5: IDRPN.C.DAT
10: MTPNCC4.BIN
11: PNCRMD1.DAT
12-40: not used
46: PNCDR.DAT

out: output: OUTDRPN.C.LIS
Program PROAVN:  
in:  
  unit 5: program control  
  10: molecular topology (binary)  
  11-40: (MD/SD trajectory) coordinates  

out: output:  

Example PCI:  
command: JAVNPCI.COM  
in:  
  unit 5: IAVNPCI.DAT  
  10: MTPCIC4.BIN  
  11: PCIRMD1.DAT  
  12: PCIRMD2.DAT  
  13-40: not used  

out: output: OUTAVNPCI.LIS  

Program PROAHB:  
in:  
  unit 5: program control  
  10: molecular topology (binary)  
  11-40: (MD/SD trajectory) coordinates  

out: output:  

Example PNC:  
command: JAHBPNC.COM  
in:  
  unit 5: IAHPNPC.DAT  
  10: MTPNC4.BIN  
  11: PNCRMD1.DAT  
  12-40: not used  

out: output: OUTAHBPNC.LIS  

Program PROMHD:  
in:  
  unit 5: program control  
  10: molecular topology (binary)  
  11-40: MD/SD trajectory coordinates
Example PNC: command: JMHBPN.C.COM
in: unit 5: JMHBPN.DAT
10: MTPNCC4.BIN
11: PNCRMD1.DAT
12-40: not used
out: output: OUTJMHBPN.LIS

Program PROCCG: in: unit 5: program control
9: reference coordinates
10: molecular topology (binary)
11-40: MD/SD trajectory coordinates
out: output:

Example PCI: command: JCOCPCI.COM
in: unit 5: JCOCPCI.DAT
9: PCIXEM1.DAT
10: MTPCIC4.BIN
11: PCIRMD1.DAT
12: PCIRMD2.DAT
13-40: not used
out: output: OUTCOCPCI.LIS

Program PROAVS: in: unit 5: program control
10: molecular topology (binary)
11-40: MD/SD trajectory coordinates
out: unit 41: averaged solvent coordinates (binary)
42: averaged solvent coordinates (formatted)

output:

**Example PCI:**

**command:** JAVSPCI.COM

**in:**

- unit 5: IAVSPCI.DAT
- 10: MTPCIC4.BIN
- 11: PCIRM1D1.DAT
- 12: PCIRM1D2.DAT
- 13-40: not used

**out:**

- unit 41: not used (deleted)
- 42: PCXAVD02.DAT
- output: OUTAVSPCI.LIS

**Program PROCX:**

**in:**

- unit 5: program control
- 10: molecular topology (binary)
- 11: first configuration
- 12: second configuration

**out:**

output:

**Example PNC:**

**command:** JCDOXPNC.COM

**in:**

- unit 5: ICOXPNC.DAT
- 10: MTPNCC4.BIN
- 11: PNCXMOD1.DAT
- 12: PNCXMD1.DAT

**out:**

output: OUTCOXPNC.LIS

**Program PROCQ:**

**in:**

- unit 5: program control
- 10: molecular topology (binary)
11: first configuration
12: second configuration

Example PNC: command: JCOQPNC.COM
in: unit 5: ICOQPNC.DAT
10: MTPNN4.BIN
11: PNCXMOD1.DAT
12: PNCXMD1.DAT
out: output: OUTCOQPNC.LIS

Program PROC3B: in: unit 5: program control
10: molecular topology (binary)
11: first configuration
12: second configuration
out: output:

Example PCL: command: JCOBPCL.COM
in: unit 5: ICOBPCL.DAT
10: MTPPCL4.BIN
11: PCLXMOD3.DAT
12: PCLXAV02.DAT
out: output: OUTCOBPCL.LIS

Program PROCAB: in: unit 5: program control
10: molecular topology (binary)
11: first configuration
12: second configuration
Example PCL: command: JCAEPCL.COM
in: unit 5: ICABPCL.DAT
     10: MTPCLC4.BIN
     11: PCLXMOD3.DAT
     12: PCLXAV02.DAT
out: output: OUTCABPCL.LIS

Program PROCOD: in: unit 5: program control
               10: molecular topology (binary)
               11: first configuration
               12: second configuration
out: output:

Example PNC: command: JCODPNC.COM
in: unit 5: ICODPNC.DAT
     10: MTPNCC4.BIN
     11: PNCXMOD1.DAT
     12: PNCXMD1.DAT
out: output: OUTCODPNC.LIS

Program PRONBL: in: unit 5: program control
                 10: molecular topology (binary)
                 11-26: solute (+solvent) coordinates
                 27: solvent coordinates (optional)
out: output:
Example PNC: command: JNBLPNC.COM
in: unit 5: INBLPNC.DAT
10: MTPNC4.BIN
11: PNCXMD1.DAT
12-26: not used
27: not used
out: output: OUTNBLPNC.LIS

Program PROCHB: in: unit 5: program control
10: molecular topology (binary)
11-26: solute (+solvent) coordinates
27: solvent coordinates (optional)
out: output:

Example PCI: command: JCHBPICI.COM
in: unit 5: ICHBPICI.DAT
10: MTPCIC4.BIN
11: PCIXMD1.DAT
12-26: not used
27: not used
out: output: OUTCHBPICI.LIS

Program PROCS: in: unit 5: program control
10: molecular topology (binary)
11-26: solute (+solvent) coordinates
27: solvent coordinates (optional)
out: output:
Example PCL: command: JCPSPCL.COM
in:  unit 5: ICPSPLC.PLAT
     10: MTPXCLQ4.BIN
     11: PCLXAVO2.DAT
     12-26: not used
     27: PCWXAOO2.DAT
out: output: OUTCPSPCL.LIS

Program PROCOS: in:  unit 5: program control
                  11: first solvent configuration
                  12: second solvent configuration.
out: output:

Example PCL: command: JCOSPCL.COM
in:  unit 5: ICOSPCL.DAT
     11: PCWXMOD7.DAT
     12: PCWXAO02.DAT
out: output: OUTCOSPCL.LIS

Program PROTCF: in:  unit 5: program control
                  10: molecular topology (binary)
                  11-40: MD/SD trajectory coordinates
out:  unit 41: time series (binary)
     output:

Example PCI: command: JTCFPCI.COM
in:  unit 5: ITCFPCI.DAT
     10: MTPCIC4.BIN
11: PCIRMD1.DAT
12: PCIRMD2.DAT
13-NO: not used

cut: unit 41: not used (deleted)
output: OUTTCFPCL.LIS
6. Examples of coordinate transformers or reducers

**Program PROMCF:** in: unit 5: program control

9: atom pointer list (for reduction, optional)
11-40: MD/SD trajectory coordinates

out: unit 41: merged (reduced) trajectory

**Example PCI:** command: JMCFFPCI.COM

in: unit 5: IMCFFPCI.DAT
9: not used
11: PCIRMD1.DAT
12: PCIRMD2.DAT
13-40: not used

out: unit 41: PCIRMD02.DAT

output: OUTMCFFPCI.LIS

**Program PROPSF:** in: unit 5: program control

10: molecular topology (binary)
11-40: MD/SD trajectory coordinates
50: X-ray coordinates (optional)
51-60: averaged coordinates (optional)

out: unit 41: formatted coordinates (MCF)

output:

**Example PNC:** command: JPSFPNCC.COM

in: unit 5: IPSFPNCC.DAT
10: MTPNCC4.BIN
VII-31

11: PNCRM01.DAT
12-40: not used
50: not used
51-60: not used

unit 41: PNCRM01.FMT (deleted)
output: OUTPSFFNC.LIS

Program PROPDF: in:

unit 5: program control
10: molecular topology (binary)
11-21: atom coordinates
21: solvent configuration (optional)
22: reference configuration (optional)

out:

unit 41: MCF coordinates
output:

Example PNC: command:

JPDPFPNC.COM

in:
unit 5: IPDFPNC.DAT
10: MTPNCC4.BIN
11: PNCXAV01.DAT
12-21: not used
21: not used
22: PNCPMOD1.DAT

out:
unit 41: PNCXAV01.MCF
output: OUTPDFPNC.LIS