MacroModel 10.5

User Manual

Schrödinger Press

MacroModel User Manual Copyright © 2014 Schrödinger, LLC. All rights reserved.

While care has been taken in the preparation of this publication, Schrödinger assumes no responsibility for errors or omissions, or for damages resulting from the use of the information contained herein.

Canvas, CombiGlide, ConfGen, Epik, Glide, Impact, Jaguar, Liaison, LigPrep, Maestro, Phase, Prime, PrimeX, QikProp, QikFit, QikSim, QSite, SiteMap, Strike, and WaterMap are trademarks of Schrödinger, LLC. Schrödinger, BioLuminate, and MacroModel are registered trademarks of Schrödinger, LLC. MCPRO is a trademark of William L. Jorgensen. DESMOND is a trademark of D. E. Shaw Research, LLC. Desmond is used with the permission of D. E. Shaw Research. All rights reserved. This publication may contain the trademarks of other companies.

Schrödinger software includes software and libraries provided by third parties. For details of the copyrights, and terms and conditions associated with such included third party software, use your browser to open third_party_legal.html, which is in the docs folder of your Schrödinger software installation.

This publication may refer to other third party software not included in or with Schrödinger software ("such other third party software"), and provide links to third party Web sites ("linked sites"). References to such other third party software or linked sites do not constitute an endorsement by Schrödinger, LLC or its affiliates. Use of such other third party software and linked sites may be subject to third party license agreements and fees. Schrödinger, LLC and its affiliates have no responsibility or liability, directly or indirectly, for such other third party software and linked sites, or for damage resulting from the use thereof. Any warranties that we make regarding Schrödinger products and services do not apply to such other third party software or linked sites, or to the interaction between, or interoperability of, Schrödinger products and services and such other third party software.

August 2014

Contents

Document Conventions xi

Chapter 1: MacroModel Overview 1

1.1 MacroModel User Manual 1

1.2 MacroModel 1

1.3 MacroModel and Maestro Interaction 2

1.4 Calculation Preparation and Submission 2

1.5 Command-Line Utilities 3

1.6 Running Schrödinger Software 4

1.7 Starting Jobs from the Maestro Interface 5

1.8 Citing MacroModel in Publications 7

Chapter 2: Basic Molecular Modeling 9

2.1 MacroModel Force Field Implementation 9

2.2 Parameter Quality Considerations 11

2.3 MacroModel Solvation Treatment 13

2.4 Truncation of Electrostatic Interactions 14

2.4.1 Energetics of Charges 15
2.4.2 Molecular Mechanics Description of Charge 15
2.4.3 Bond Dipole Cutoffs (BDCO) 16
2.4.4 BDCO and Molecular Modeling 16

2.5 General Guidelines for Convergence 17

2.5.1 Minimization 17
2.5.2 Conformational Searching 17
2.5.3 Molecular and Stochastic Dynamics 18

2.6 Problem Size: Limits of Operation 19

2.7 Nonbonded Cutoffs 19

2.8 Electrostatic Treatments 20

2.9 File Conversion 20

2.10 Modified Force Field, Solvent, or Atom Type Files 21

2.11 Hydrogen Treatment 21

2.11.1 Hydrogens on Hetero Atoms........ 21

2.11.2 Hydrogen Treatment in the AMBER* Force Field........ 21

Chapter 3: Running MacroModel From the Command Line........ 23

3.1 Preparing for MacroModel Calculations........ 23

3.1.1 Environment Variables 23

3.1.2 The Command File 24

3.1.3 The Command File Format 24

3.1.4 The Simplified Input File 26

3.2 Submitting Jobs From the Command Line........ 26

3.3 Running Remote and Distributed Calculations 28

3.4 Interacting With a Running MacroModel Job........ 29

Chapter 4: General Settings 31

4.1 General MacroModel Calculation Setup........ 31

4.1.1 **Specifying the Structure Input Source** 31

4.2 The Potential Tab........ 32

4.3 Fixing and Freezing Atoms in Large Systems 33

4.3.1 Methods for Freezing and Fixing Atoms 34

4.3.2 Freezing and Fixing Individual Atoms 35

4.3.3 The Substructure Facility 35

4.4 The Constraints Tab........ 37

4.4.1 Constraining Atoms, Distances, Angles, and Torsions........ 37

4.4.2 Freezing Atoms........ 39

4.5 The Substructure Tab 40

4.5.1 Defining a Substructure 40

4.5.2 Creating a Shell of Atoms 41

4.5.3 Reading and Writing Substructures 42

4.6 Setting Up, Running, And Monitoring Jobs 43

Chapter 5: Current Energy Calculations 45

5.1 The Current Energy Panel 45

5.2 The ECalc Tab 45

5.3 Command File Examples 46

5.4 The Force Field Viewer 48

5.5 Checking and Interpreting Results 50

Chapter 6: Minimizations 51

6.1 The Minimization Panel 51

6.2 Setting Minimization Parameters 52

6.2.1 Minimization Methods 52

6.2.2 Convergence Parameters 54

6.2.3 Using Several Different Minimization Stages During a Minimization 54

6.3 Setting Up Minimization of Conformers with Redundant Conformer Elimination 55

6.4 Partition Coefficient Estimation 57

6.5 Automatic Setup (AUTO) 58

6.6 Preparation of Multi-Ligand Structure Files With premin 59

6.7 Command File Examples 60

6.7.1 Constrained Minimization 60

6.7.2 Multiple-Method (MWRT) Minimizations 61

6.7.3 Energy Minimization of Multiple Non-Conformers 62

6.7.4 Multiple Conformer Minimization With Automatic Redundant Conformer Elimination 63

6.7.5 Multiple Minimization with Substructures Defined by ASL 64

6.7.6 Partition Coefficient Estimation 65

6.8 Checking and Interpreting Results 66

Chapter 7: Coordinate Scans 69

7.1 Performing a Coordinate Scan in Maestro 69

7.2 Command File Examples 71

7.3 Checking and Interpreting Results 72

Chapter 8: Conformational Searches 75

8.1 Conformational Search Methods 75

8.2 Performing Conformational Searches 76

8.2.1 Conformational Search Method 77

8.2.2 Automatic Setup of Conformational Search Variables 79

8.2.3 Global Search Restrictions 81

8.2.4 Low-Mode Parameters 82

8.2.5 Setting Conformational Search Variables Manually 82

8.2.5.1 Ring Closures 83

8.2.5.2 Torsion Rotations 84

8.2.5.3 Molecule Trans/Rot 85

8.2.5.4 Comparison Atoms 87

8.2.5.5 Chiral Atoms 88

8.2.5.6 Distance Check 89

8.2.5.7 Torsion Check 90

8.2.5.8 Ligand Bonds 90

8.3 Command File Examples 91

8.3.1 Conformational Search Using MCMM 91

8.3.2 Multi-Structure Conformational Search Using LMOD 93

8.3.3 Mixed MCMM/Low-Mode Search Using a Substructure File 94

8.3.4 Conformational Searches Using Multiple Minimization Methods (MWRT) 95

8.4 Checking and Interpreting Results 96

Chapter 9: Dynamics Calculations 97

9.1 The Dynamics Panel 97

9.2 Performing a Dynamics Calculation 97

9.2.1 The Monitor Tab 97

9.2.1.1 Surface Areas 99

9.2.1.2 Distances 99

9.2.1.3 Angles 100

9.2.1.4 Torsions 100

9.2.1.5 H-Bonds 101

9.2.1.6 Monitoring Output 102

9.2.2 The Dynamics Tab 102

9.3 Command File Examples 104

9.3.1 Stochastic Dynamics 104

9.3.2 Simulated Annealing 105

9.4 Checking and Interpreting Results 106

Chapter 10: MC/SD Calculations **107**

10.1 The MC/SD Panel 107

10.2 Setting Up MC/SD Calculations 107

10.3 File Examples 109

10.4 Checking and Interpreting Results 110

Chapter 11: Minta Calculations **111**

11.1 The MINTA Panel 111

11.2 Command File Examples 113

11.3 Checking and Interpreting Results 114

Chapter 12: Protein Loop Construction **115**

12.1 Performing LOOP Calculations 115

12.1.1 Input Restrictions 115

12.1.2 Structure Handling Considerations 116

12.2 Example .lsq Input File 116

12.3 Command File Examples 117

12.3.1 LOOP Job Using the Input Structure Sequence 117

12.3.2 LOOP Job Using the Sequence From a File 119

12.4 Example LOOP Job Output 119

12.4.1 Atom Renumbering Using Input Protein Structure 119

12.4.2 Conformational Search Using Input Protein Structure 119

12.4.3 LOOP Run Using an .lsq File as Input 120

Chapter 13: Embrace **121**

13.1 Minimizations With Embrace 121

13.2 Conformational Searches With Embrace 123

13.3 Specifying a Substructure for Embrace 126

13.3.1 Using a Ligand To Assist in Creating a Receptor Substructure 127

13.3.2 Creating a Substructure with All Receptor Atoms Fixed or Frozen 127

13.4 Command File Examples 128

13.4.1 Embrace Minimization Calculations 128

13.4.1.1 Interaction Mode 128

13.4.1.2 Energy Difference Mode 131

13.4.2 Conformational Searches With Embrace 134

13.4.2.1 MCMM Conformational Search 134

13.4.2.2 Low-Mode Conformational Search 137

13.4.2.3 MCMM Conformation Search With COPY/ALGN 139

13.4.3 Distributed Embrace Calculations 141

Chapter 14: Free Energy Simulations **145**

14.1 Free Energy Perturbation 145

14.2 Setting Up FEP Calculations 146

14.2.1 The Structure File 146

14.2.2 The Command File 147

14.2.3 Other Possible Command Files 149

14.2.4 The Output File 150

14.3 Criteria for a Successful Simulation 152

14.4 Other Types of Free Energy Calculations 154

14.4.1 Hydrogen Bonding Preference of a Glycyl Lactam in Organic Solution 154

14.4.2 Conformational Free Energies for a Diamide in Solution 155

14.5 Literature 156

Chapter 15: Redundant Conformer Elimination **159**

15.1 Eliminating Redundant Conformers Using Maestro 159

15.2 Command File Examples 161

Chapter 16: Macrocycle Conformational Sampling **163**

16.1 The Macrocycle Conformational Sampling Panel 163

16.2 The macrocycle_conformational_sampling.py script 165

Chapter 17: Additional Features **167**

17.1 Geometry Calculations 167

17.2 Calculating Interaction Energies Using ASET 167

17.3 Rewinding the Output File for Additional Minimization 168

17.4 Visualizing Vibrational Modes 169

17.5 Alignment of Structures 170

17.6 autoref: Restrained Minimizations 171

17.7 serial_split: Split Serial Search Output Structure Files 172

17.8 ring_templating: Creating Ring Conformation Templates 172

17.8.1 Generating Ring Conformation Templates 173

17.8.2 Using Ring Conformation Templates 174

17.9 Python Scripts 175

Getting Help **177**

References **181**

Index **185**

Document Conventions

In addition to the use of italics for names of documents, the font conventions that are used in this document are summarized in the table below.

Font	Example	Use
Sans serif	Project Table	Names of GUI features, such as panels, menus, menu items, buttons, and labels
Monospace	`$SCHRODINGER/maestro`	File names, directory names, commands, environment variables, command input and output
Italic	*filename*	Text that the user must replace with a value
Sans serif uppercase	CTRL+H	Keyboard keys

Links to other locations in the current document or to other PDF documents are colored like this: Document Conventions.

In descriptions of command syntax, the following UNIX conventions are used: braces `{ }` enclose a choice of required items, square brackets `[ ]` enclose optional items, and the bar symbol `|` separates items in a list from which one item must be chosen. Lines of command syntax that wrap should be interpreted as a single command.

File name, path, and environment variable syntax is generally given with the UNIX conventions. To obtain the Windows conventions, replace the forward slash / with the backslash \ in path or directory names, and replace the $ at the beginning of an environment variable with a % at each end. For example, `$SCHRODINGER/maestro` becomes `%SCHRODINGER%\maestro`.

Keyboard references are given in the Windows convention by default, with Mac equivalents in parentheses, for example CTRL+H (⌘H). Where Mac equivalents are not given, COMMAND should be read in place of CTRL. The convention CTRL-H is not used.

In this document, to *type* text means to type the required text in the specified location, and to *enter* text means to type the required text, then press the ENTER key.

References to literature sources are given in square brackets, like this: [10].

MacroModel Overview

1.1 MacroModel User Manual

This manual contains an introduction to the MacroModel molecular mechanics program. For detailed information about command-line MacroModel and MacroModel operation codes, see the *MacroModel Reference Manual*. For tutorial exercises that demonstrate Maestro and MacroModel functionality, see the *MacroModel Quick Start Guide*. For information on using Maestro, the graphical user interface (GUI) for MacroModel, see the Maestro online help or the *Maestro User Manual*.

Chapter 1 provides a brief introduction to MacroModel and describes how it interacts with Maestro. A good general background to force-field-based molecular modeling, particularly as it is implemented in MacroModel, is given in Chapter 2. Running MacroModel from the command-line (without Maestro) is described in Chapter 3. Chapters 4-15 provide more information on how to run particular classes of calculations. Chapter 17 describes additional features: geometry queries and interaction energy calculations. Information on how to get additional help for performing MacroModel calculations is given at the back of this manual.

1.2 MacroModel

MacroModel is a general purpose, force-field-based molecular modeling program with applicability to a wide range of chemical systems. MacroModel provides researchers with multiple advanced methods to aid the understanding of chemical structure, energetics, and dynamics. A large selection of force fields is available in MacroModel, including the latest technical advances introduced into OPLS_2005, a force field that Schrödinger is actively developing. Numerous minimization methods are available, enabling geometry optimizations for a broad selection of structural classes. A wide range of methods is available for conformational searching, allowing efficient sampling of the potential energy surface for low-energy structures, for systems ranging from small molecules to entire proteins. Solvation effects can be accounted for using the efficient continuum solvation models employed by MacroModel. Additional advanced features include molecular dynamics simulations, free energy perturbation simulations, and pure and mixed methods for ensemble sampling.

1.3 MacroModel and Maestro Interaction

Maestro is the graphical user interface for MacroModel. MacroModel runs calculations as independent tasks and consequently does not tie up Maestro during lengthy computations. Maestro monitors MacroModel tasks so that both numerical and structural information may be viewed while the tasks are running. Such monitoring is the default mode of operation for newly started tasks, although monitoring can be broken off and reestablished at a later time. Thus, you can initiate several MacroModel tasks, disconnect from them, carry out graphical modeling operations, and periodically reconnect to and examine the progress of each of the previously submitted MacroModel tasks.

1.4 Calculation Preparation and Submission

MacroModel calculations can be prepared and launched from the Maestro GUI or from the command line. To run a MacroModel job on a remote computer, you must have access to a hosts file, named `schrodinger.hosts`, that lists hosts on which Schrödinger software is installed for execution. Details of setting up this file can be found in Chapter 7 of the *Installation Guide*.

Separate panels for current energy, energy minimization, dihedral driving, conformational search, ligand torsion search, multiple minimization, dynamics, MC/SD, and MINTA are available in Maestro. To set up a calculation, display the relevant panel and adjust the settings as desired, then launch the job. Alternatively, you can choose to write out the structure and command files that you will need to launch the job later from the command line. Details of how to do this are given in Section 1.7 on page 5.

Once you set up a job in Maestro and start it or write out the input, Maestro writes out a molecular structure file and a command file. These files are required to run MacroModel calculations. The molecular structure file contains the structures to be used as input in the calculation. The command file contains the name of the input structure file, the name of the output structure file, and an ordered list of operation codes (*opcodes)* for the calculations.

For some types of calculations, Maestro writes out a simplified input file, which consists of keyword-value pairs in free format, and which can be used instead of the command file.

Some types of MacroModel calculations cannot be prepared or submitted from Maestro. For these jobs, you must manually create the necessary command file, which provides MacroModel with the instructions it needs to perform the related calculation. For this task, you might find it easiest to use a Maestro-generated command file as a template.

The *MacroModel Reference Manual* contains detailed information about all of the MacroModel commands and operation codes. Examples of command files for various standard modeling operations are provided at the end of relevant chapters in this manual. Because all jobs are handled by the Schrödinger Job Control facility, MacroModel jobs can be monitored from Maestro even when jobs are launched from the command line.

1.5 Command-Line Utilities

MacroModel is distributed with the following command-line utilities, which are stored in the `utilities` directory of the software installation.:

`autoref`	Performs a restrained minimization of a protein-ligand structure using MacroModel. See Section 17.6 on page 171.
`premin`	Robustly minimizes ligand structures in a multistructure Maestro file using MacroModel. Problematic structures are separated out for examination. See Section 7.5 on page 64.
`serial_split`	Splits the output structure file from a serial search into separate structure files for each individual search performed. See Section 17.7 on page 172.

The following command-line utilities, which are installed with the software, may be useful in conjunction with MacroModel calculations:

`applyhtreat`	Adds or removes hydrogen atoms, dummy atoms, and lone pairs to structures in a MacroModel or Maestro file and produces a new Maestro file.
`ligfilter`	Selects structures from a multistructure Maestro file based on properties. (Supersedes `propfilter`.)
`maesubset`	Selects a subset of the structures from a multistructure Maestro file.
`proplister`	Lists selected properties present in structures within a multistructure Maestro file.
`sdconvert`	Converts between SD and Maestro or MacroModel file formats.
`sdsubset`	Selects a subset of the structures from a multistructure SD file.
`structconvert`	Converts between various structure formats, including Maestro, MacroModel, SD, and Mol2.

For more information on these utilities, see the *General Utilities* manual. Utilities for the obsolete MacroModel file format are described in Appendix G of the *MacroModel Reference Manual*.

1.6 Running Schrödinger Software

Schrödinger applications can be run from a graphical interface or from the command line. The software writes input and output files to a directory (folder) which is termed the *working directory*. If you run applications from the command line, the directory from which you run the application is the working directory for the job.

Linux:

To run any Schrödinger program on a Linux platform, or start a Schrödinger job on a remote host from a Linux platform, you must first set the `SCHRODINGER` environment variable to the installation directory for your Schrödinger software. To set this variable, enter the following command at a shell prompt:

csh/tcsh:	`setenv SCHRODINGER` *installation-directory*
bash/ksh:	`export SCHRODINGER=`*installation-directory*

Once you have set the `SCHRODINGER` environment variable, you can run programs and utilities with the following commands:

```
$SCHRODINGER/program &
$SCHRODINGER/utilities/utility &
```

You can start the Maestro interface with the following command:

```
$SCHRODINGER/maestro &
```

It is usually a good idea to change to the desired working directory before starting the Maestro interface. This directory then becomes the working directory.

Windows:

The primary way of running Schrödinger applications on a Windows platform is from a graphical interface. To start the Maestro interface, double-click on the Maestro icon, on a Maestro project, or on a structure file; or choose Start → All Programs → Schrodinger-2014-3 → Maestro. You do not need to make any settings before starting Maestro or running programs. The default working directory is the Schrodinger folder in your Documents folder.

If you want to run applications from the command line, you can do so in one of the shells that are provided with the installation and have the Schrödinger environment set up:

- Schrödinger Command Prompt—DOS shell.
- Schrödinger Power Shell—Windows Power Shell (if available).

You can open these shells from Start → All Programs → Schrodinger-2014-3. You do not need to include the path to a program or utility when you type the command to run it. If you want access to Unix-style utilities (such as `awk`, `grep`, and `sed`), preface the commands with `sh`, or type `sh` in either of these shells to start a Unix-style shell.

Mac:

The primary way of running Schrödinger software on a Mac is from a graphical interface. To start the Maestro interface, click its icon on the dock. If there is no Maestro icon on the dock, you can put one there by dragging it from the SchrodingerSuite2014-3 folder in your Applications folder. This folder contains icons for all the available interfaces. The default working directory is the Schrodinger folder in your Documents folder (`$HOME/Documents/Schrodinger`).

Running software from the command line is similar to Linux—open a terminal window and run the program. You can also start Maestro from the command line in the same way as on Linux. The default working directory is then the directory from which you start Maestro. You do not need to set the `SCHRODINGER` environment variable, as this is set in your default environment on installation. To set other variables, on OS X 10.7 use the command

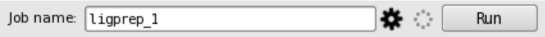
```
defaults write ~/.MacOSX/environment variable "value"
```

and on OS X 10.8 and 10.9 use the command

```
launchctl setenv variable "value"
```

1.7 Starting Jobs from the Maestro Interface

To run a job from the Maestro interface, you open a panel from one of the menus (e.g. Tasks), make settings, and then submit the job to a host or a queueing system for execution. The panel settings are described in the help topics and in the user manuals. When you have finished making settings, you can use the Job toolbar to start the job.



Job name: ligprep_1

You can start a job immediately by clicking Run. The job is run on the currently selected host with the current job settings and the job name in the Job name text box. If you want to change the job name, you can edit it in the text box before starting the job. Details of the job settings are reported in the status bar, which is below the Job toolbar.

If you want to change the job settings, such as the host on which to run the job and the number of processors to use, click the Settings button. (You can also click and hold, and choose Job Settings from the menu that is displayed.)

You can then make the settings in the Job Settings dialog box, and choose to just save the settings by clicking OK, or save the settings and start the job by clicking Run. These settings apply only to jobs that are started from the current panel.

If you want to save the input files for the job but not run it, click the Settings button and choose Write. A dialog box opens in which you can provide the job name, which is used to name the files. The files are written to the current working directory.

The Settings button also allows you to change the panel settings. You can choose Read, to read settings from an input file for the job and apply them to the panel, or you can choose Reset Panel to reset all the panel settings to their default values.

You can also set preferences for all jobs and how the interface interacts with the job at various stages. This is done in the Preferences panel, which you can open at the Jobs section by choosing Preferences from the Settings button menu.

Note: The items present on the Settings menu can vary with the application. The descriptions above cover all of the items.

The icon on the Job Status button shows the status of jobs for the application that belong to the current project. It starts spinning when the first job is successfully launched, and stops spinning when the last job finishes. It changes to an exclamation point if a job is not launched successfully.

Clicking the button shows a small job status window that lists the job name and status for all active jobs submitted for the application from the current project, and a summary message at the bottom. The rows are colored according to the status: yellow for submitted, green for launched, running, or finished, red for incorporated, died, or killed. You can double-click on a row to open the Monitor panel and monitor the job, or click the Monitor button to open the Monitor panel and close the job status window. The job status is updated while the window is open. If a job finishes while the window is open, the job remains displayed but with the new status. Click anywhere outside the window to close it.

Jobs are run under the Job Control facility, which manages the details of starting the job, transferring files, checking on status, and so on. For more information about this facility and how it operates, as well as details of the Job Settings dialog box, see the *Job Control Guide*.

1.8 Citing MacroModel in Publications

The use of this product should be acknowledged in publications as:

MacroModel, version 10.5, Schrödinger, LLC, New York, NY, 2014.

Basic Molecular Modeling

The potential energy model used for MacroModel energy calculations is the classical one known as molecular mechanics. It is an empirical model which is parameterized to reproduce known data from experiment or quantum mechanical calculations. The equation and parameter sets which allow calculation of energy from a molecular geometry are known as force fields.

Generally, MacroModel uses equations and parameters from published standard force fields. However, the MacroModel implementations differ in various ways from the authentic force fields. We distinguish the MacroModel implementation from the original force fields by adding a "*" to the end of the force field name. MacroModel force field parameters and equations selectors are found in force field files having suffixes `.fld` (e.g., `mm2.fld`). Differences between the MacroModel force fields and the standard fields are summarized below.

2.1 MacroModel Force Field Implementation

All MacroModel force field files contain the authentic parameter set published by the original authors of the force field. In addition to these parameters are other parameters from other sources (e.g., the literature or work at Columbia). Parameters in the force field files are labeled as to their origin (O = original from the force field authors, M = modified from the original values, and A = added from some other source where no original parameter exists). They are also labeled by quality (1 = high quality final value, 2 = tentative value based on more than one experimental or quantum calculation, 3 = crude low quality parameter). Sources of A and M parameters are given at the ends of the lines in the force field files.

The MacroModel implementations of standard force fields differ from the authentic force fields in the following ways:

MM2*—All force field equations are identical with those of authentic MM2 from Allinger [1], with the exception of the following:

- The electrostatic equation (MM2* uses partial charges and Coulomb's law, whereas MM2 uses bond dipoles and the Jeans equation).
- The out-of-plane bending equation (MM2* uses an improper torsion while MM2 uses a pyramidalization distance—the difference being insignificant except for substantially distorted sp^2 systems).
- Handling of conjugation (MM2* uses specific V1-V3 torsional terms for various conjugated systems, whereas MM2 uses an SCF π calculation in uncommon systems).

MM3*—All force field equations are identical to those of authentic MM3 from Allinger [2], except for those differences listed above for MM2.

AMBER*—All force field equations are identical to those of authentic AMBER from Kollman. The MacroModel default for hydrogen bonding uses Kollman's recent 6,12-Lennard Jones treatment [3] and an improved peptide backbone parameter set [4].

OPLS*—All force field equations are identical to those of OPLS/AMBER from Jorgensen [5].

OPLS_2001—Also referred to as OPLSAA, this force field, developed by Professor W. Jorgensen of Yale University, is probably the best available for condensed-phase simulations of peptides. All force-field equations are identical to those of authentic OPLSAA [6]. MacroModel's implementation has been validated by comparison to BOSS OPLSAA calculations for a wide variety of organic systems. Comparisons to ab initio calculations and experiment show that OPLS_2001 reproduces conformational energies well for systems for which it has been specifically parameterized. Especially good results can be expected for proteins. With the exception of improved charge, van der Waals and torsion parameters for sulfur in thiols and thiol ethers [7], all parameters are native OPLS_2001. The new thio parameters, which use appreciably smaller charges on sulfur and which have been validated in liquid-phase simulations on thiols and thiol ethers, significantly improve the conformational energetics of CYS and MET residues in proteins.

Note: The OPLS_2001 force field is now considered obsolete and is no longer available. It is superseded by OPLS_2005. The information above is retained for its connection with OPLS_2005.

OPLS_2005—OPLS_2005 is an enhanced version of the OPLS_2001 all atom force field that was developed by Schrödinger to provide a larger coverage of organic functionality. In particular all torsional parameters have been refit to reproduce the conformational energetics derived at a higher level of quantum theory, additional charges have been fit to support additional organic functionality, and the coverage of elements has been extended. The parameters for proteins have been updated to the ones published more recently [7, 34].

AMBER94—All force field equations and parameters are the same as in Cornell et al. [8], with the following small exceptions:

- In MacroModel partial charges are specified by bond dipoles rather than as charge values. The partial charges may differ slightly between the two implementations; these differences are typically in the fifth significant figure.
- The atoms defining improper torsions are not specified by the AMBER protocol in situations of high local symmetry. This may sometimes give rise to small differences in molecular energies or geometries between the two programs.

- The paper gives the two nitrogen types different van der Waals parameters, but the AMBER 4.1 program uses the same parameters for both. We follow the program's convention.

MMFF—Our implementation is identical to that described by Halgren [9–15]. We supply both MMFF94 and MMFF94s; the latter enforces planarity about delocalized sp^2 nitrogens.

Whenever a current energy calculation (ECalc) is carried out, a listing file (*jobname*`.mmo`) can be produced which contains all parameters used in the calculation along with the origin and quality of each parameter. Note that any torsion parameter where V_1- V_3 are all set to zero will not be included in the output. To include these in the listing it is necessary to include the DEBG 56 command in the command file: see the *MacroModel Reference Manual* for details. This automatic parameter referencing feature provides important information on the quality and reliability of the calculation.

Different force fields use different defaults for their electrostatic treatment (constant or distance-dependent dielectric) and their cutoff distances (van der Waals and electrostatic). It is possible to set such options exactly as in the authentic fields using the Electrostatic Treatment and Cutoff option menus in the Potential tab within Maestro's MacroModel panels.

2.2 Parameter Quality Considerations

Molecular mechanics force fields are empirical, and the accuracy of the results rest entirely on the ability of the parameters and functional forms used to mimic the real potential energy of molecules. If the parameters are deficient, is it impossible to obtain agreement with experiment or to make useful predictions.

When MacroModel performs an energy calculation, the program checks the quality of each parameter in use. Use of low quality (quality = 3) parameters, especially torsional ones, may result in inaccurate conformational energy differences and geometries. Low quality stretches often indicate crude partial charges since charge information often originates from bond dipoles. Using such low quality parameter values can cause charges and solvation energies to be inaccurate. Consequently, whenever MacroModel initiates an energy calculation, a warning and the numbers of low quality stretch, bend, and torsional parameters in use are listed in the MacroModel *jobname*`.log` file and in the Maestro Monitor panel viewing window. An example is shown below:

```
WARNING - Conformational Energies May Not Be Accurate
WARNING - Solvation Energies/Charges May Not Be Accurate
Low quality force field parameters in use:
Number of low quality stretches, bends & torsions =   1   1   8
```

The above message indicates that eight low quality torsions, one low quality bend, and one low quality stretch are in use in the calculation. Consequently, conformational energy differences and solvation energies may be unreliable. By looking in the job's *jobname*`.log` file, you can see which line(s) in the force field file is the source of the low quality parameters. Listings of the specific torsions having low quality, type 3 parameters can be found in the job's `.mmo` file after running the job with a "Complete" energy listing.

If the file notes that low quality parameters are being used, first attempt to find improved parameters by trying the system of interest with each MacroModel force field. MacroModel supports a number of different force fields, and each has its own strengths. If you are unable to find a force field with acceptable parameters for your particular system, you should seriously consider developing your own parameters, or try to obtain parameters from other practitioners in the field.

Typically, low quality torsional parameters are generalized ones (e.g., any bond between two sp^3 carbons has a 3-fold rotational barrier of ~3 kcal/mol). Thus for a specific torsion involving, for example Cl-C(sp^3)-C(sp^3)-O, a new torsional parameter line would be added to the force field file with specific V1-V3 terms which reproduce reliable experimental data or high quality ab initio calculations (e.g., 6-31G*) on, for example, β-chloroethyl methyl ether. To make the calculation more reliable, new parameters that fit experimental or high quality ab initio data need to be determined and added to the force field file. However, if you are comparing conformations that do not involve significant changes in the torsions having low quality parameters, then errors may not be large.

Obtaining correct torsional profiles around rotatable bonds is one of the most commonly encountered problems in reparameterizing a potential energy surface. For example, if a new charge set is introduced, the non-bonded interactions on either side of a rotatable bond may alter the torsional profile. It is important to test that the relative energies of the minima and maxima are in agreement with the results of high-level molecular-orbital calculations on model compounds. In order to make this sort of comparison, the molecular mechanics calculations should be performed in the gas phase; that is, with the solvation model turned off and with constant-dielectric electrostatics. Once the potential-energy surface is correctly described in the gas phase, the solvation model should correctly describe the potential-energy surface in solution.

To determine the force-field quality for the problem at hand, check the summary of low quality parameters in use by using Maestro's Force Field Viewer panel, which is opened from the Tools menu.

2.3 MacroModel Solvation Treatment

While many molecular modeling studies are carried out without including the effect of solvent, the omission is largely one of expedience. Most experimental studies are carried out in solvent, and the solvent medium can have a major effect on molecular structures and energies. For some molecular types such as a small organic molecule with only one polar functional group, solvation does not appear to be a major determinant of conformational energies. However, for molecules having several polar functional groups, the effect of solvent can be dramatic since the electrostatically least stable structures are often the most heavily solvated (stabilized) in a polar solvent.

Using an explicit solvent model is one approach; however, it has its own disadvantages. In particular, “explicit solvation” calculations run much more slowly because there are so many particles when hundreds of explicit solvent molecules are included. Furthermore, convergence is a problem in that longer simulations or different solvent starting configurations often give different final energies. Consequently, simple energy minimization is not useful in an explicit solvent.

MacroModel uses an alternative solution model which treats the solvent as a fully equilibrated analytical continuum starting near the van der Waals surface of the solute. The model is termed the GB/SA model and is described in the literature [16]. MacroModel is provided with parameter files for water (`water.slv`), octanol (`octanol.slv`), and chloroform (`chcl3.slv`). Using the GB/SA model slows calculations by a factor of approximately three relative to the gas phase. However, because of the increased accuracy of modeling in solvent, it is suggested that the GB/SA continuum solvation model be used in all calculations for molecules for aqueous solutions. Solvation is controlled by the Solvent option menu, located in the Potential tab of the MacroModel panels within Maestro, or by the `SOLV` command in MacroModel command files.

The calculation of Born radii in the GB part of the GB/SA model is performed by doing a volume integral. Stretch, bend, and non-bonded pairs (including 1-4 interactions) contribute to this integral. For large systems, such as proteins in which non-bonded cutoffs typically are used and not all non-bonded pairs are included on the non-bonded pair list in order to expedite calculation, Born radii are subject to systematic error. This is because the volume integration is performed by using pairs on the stretch, bend, and non-bonded pair lists, and the latter excludes longer range interactions when non-bonded cutoffs are in use. To correct for this error, MacroModel calculates the contribution from such longer range non-bonded pairs every time a non-bonded pair list update is done. This contribution is taken to be a fixed value and is used in energetic and derivative calculations. Although not an exact correction, inclusion of this contribution in this manner does help significantly. The correction is enabled by default; see the description for debug flags 830 and 832 under the `DEBG` opcode description in the *MacroModel Reference Manual* for information on disabling the correction.

In calculations using continuum solvation, MacroModel uses an approximate solvent accessible surface area function for derivatives and then computes final energies with a more exact area function at the end of the calculation. Consequently, the intermediate energies which are listed during energy minimization iterations will differ from the final energies.

See the *MacroModel Technical Manual* for a discussion on the performance of the GB/SA model with different sources of partial atomic charge.

2.4 Truncation of Electrostatic Interactions

For large systems, interactions between non-bonded pairs separated by more than a given distance need to be ignored in order to make simulations tractable. Such treatment for van der Waals interactions is less necessary since the interaction dies off as $1/r^6$ where r is the interatomic distance. Electrostatic interactions, on the other hand, die off as $1/r$, and pose a greater challenge because of their long range nature.

MacroModel versions prior to 8.1 used residue-based cutoffs for systems that had residue information in their coordinate files and atom-based cutoffs for systems with only one residue or without such information. MacroModel versions from 8.1 on include a new method for the truncation of electrostatic interactions. It is termed “Bond Dipole Cutoffs” (BDCO) and leverages the physics of charge-charge, charge-dipole, and dipole-dipole interactions to give very accurate absolute electrostatic and generalized Born energies. An example is the benzamidine/trypsin complex (PDB code 3ptb), which contains 3245 atoms in an all-atom representation, for which results are given in Table 2.1. In addition to a smaller error in energy by two orders of magnitude than that seen with residue-based cutoffs, the BDCO calculation uses fewer non-bonded pairs. Because of this, the BDCO calculation runs slightly faster than the residue-based cutoff calculation. Similar results are seen for generalized Born solvation polarization energies.

Table 2.1. Use of different cutoffs in a calculation on the benzamdine/trypsin complex.

Method	Number of Non-bonded Pairs in Energy Calculation	Total Electrostatic Energy (kJ/mol)	Error (kJ/mol)
BDCO	2731027	–28697.73	2.08
res[a]	3628479	–29179.45	479.64
all[b]	5254184	–28699.81	0.0

a. Residue-based cutoffs
b. All non-bonded pairs included in calculation

The following subsections provide an overview of the physical basis and implementation of this novel method for truncating electrostatic interactions.

2.4.1 Energetics of Charges

Coulomb's law states that the potential energy of two point charges separated by a distance r scales as 1/r. In contrast, the energy of a point charge interacting with a dipole scales as $1/r^2$ and the energy of two interacting dipoles scales as $1/r^3$. These physical properties determine the length scales for which truncation of the various interaction types is appropriate in order to achieve a given level of accuracy.

2.4.2 Molecular Mechanics Description of Charge

Common molecular mechanics force fields utilize Coulomb's law to describe the electrostatic interaction between atoms due to the uneven distribution of electron density across bonded atoms with different electronegativities. A carbonyl moiety, for example, is described with a partial negative charge located at the oxygen atom center and an equal but opposite charge located at the carbon atomic center. These fixed partial charges can be directly used in Coulomb's law to give molecular mechanics electrostatic energies and the associated forces.

For groups with net charge, such as a carboxyl moiety, the formal charge of the group must also be accounted for. This can be done by delocalizing the formal charge over the appropriate atoms and adding this delocalized formal charge to the partial charges due to the dipolar nature of bonds between different atoms. Thus, for the carboxyl moiety, one can think of the system as two dipoles, one across each C-O bond, as well as point charges of –0.5 located in the oxygen atoms to account for the net negative charge of the group.

Some force fields, such as MMFF, have been conceptualized using this scheme, and the charge parametrization for an arbitrary molecule is achieved by assigning dipoles across bonds as well as delocalized formal charges on atomic centers and summing up these values to obtain fixed partial charges on atomic centers. Other force fields, such as AMBER*, assign charges to arbitrary molecules by dividing a given molecule into small groups of atoms with unit charge, like amino acids, for which there exist fixed partial charge data. Even in the latter case, the fixed partial charges can be transformed into a system of bond dipoles and delocalized formal charges.

In addition to assigning partial charges based on the selection of a force field, MacroModel allows you to specify custom partial charges for a system in the input structure file (see the `CHGF` opcode in the *MacroModel Reference Manual*). MacroModel is capable of decomposing this list of partial charges into delocalized formal charges and bond dipoles. In the event that the partial charges so specified for a molecule do not sum to the net formal charge of that molecule, the input structure partial charges will be modified to meet this constraint before the decomposition is performed.

2.4.3 Bond Dipole Cutoffs (BDCO)

Since a system of atoms can be described as consisting of bond dipoles and atom-centered delocalized formal charges, the energetics of the charges can be considered as due to charge-charge, charge-dipole, and dipole-dipole interactions. This conceptualization is at the heart of the Bond Dipole Cutoff (BDCO) method. By applying different cutoff distance criteria to these three different types of interactions based on how they scale with distance, energetic accuracy is preserved while retaining the benefits of having a short non-bonded pair list. The implementation of BDCO *does not* involve any change in the functional form of a molecular mechanics force field. Coulomb's law in its familiar form is preserved. Rather, the BDCO algorithm applies different, successively shorter, cutoff distances to charge-charge, charge-dipole, and dipole-dipole interactions, where the charges are delocalized formal charges and the dipoles are bond dipoles. The resultant list of interactions is then used to generate charge products for pairs of atoms; the "charge product" is defined as the numerator in Coulomb's law. The charge product for a given pair of atoms is used directly in Coulomb's law to give the electrostatic energy and associated force for that pair.

2.4.4 BDCO and Molecular Modeling

The reason BDCO works is that although long cutoffs are used for charge-dipole and charge-charge interactions, there are relatively few such interactions in proportion to the number of dipole-dipole interactions in systems commonly modeled using molecular mechanics. For example, of the 20 types of amino acids seen in proteins, only 5 usually have charged (Lys, Arg, His, Glu, Asp) side chains. For a uniform distribution of amino acid types, 1/4 of the amino acids in a sequence will be charged. Assuming approximately 15 atoms per amino acid, the delocalized formal charge for a charged amino acid will on average reside on 2 of the 15 atoms. 1/4 * 2/15 = 1/30, so only 1 of every 30 atoms in an average protein is expected to have delocalized formal charge. The number of dipoles in the system is approximately equal to the total number of atoms. Since delocalized formal charges occur on 1/30 of the atoms, the proportion of charge-charge interactions is 1/30 * 1/30 = 1/900 of the total number of charge-charge, charge-dipole, and dipole-dipole interactions. Likewise, the proportion of charge-dipole interactions is 1/30 * 1 = 1/30. The number of charge-charge and charge-dipole interactions in such a system is very small compared to the number of dipole-dipole interactions, and assigning long cutoffs to the former two types of interactions will not impact the size of the non-bonded list. However, use of long cutoffs for such interactions will positively impact the accuracy of the calculation due to the long range nature of these interactions.

Refer to the `BDCO` opcode in the *MacroModel Reference Manual* for more information.

2.5 General Guidelines for Convergence

It is as important to fully explore the potential energy surface as it is to describe that surface correctly by means of a high-quality force field. We use the term “convergence” to describe this issue. A common problem in modeling is obtaining unconverged results: the calculation gives a significantly different answer if the calculation continues or is repeated from different initial conditions. The term “convergence” has somewhat different implications, depending on the nature of the procedure that is being carried out, and for each type of energy calculation, there are different convergence issues. These issues are discussed in the following sections.

2.5.1 Minimization

It is important that, during energy minimization, the energy is minimized to a low gradient norm. Energy minimization calculations generally converge readily, although there is no guarantee that final structures are low in energy relative to the actual global minimum. By default, we set a convergence criterion for minimization of a gradient to < 0.05 kJ/Å-mol, and this is usually satisfactory. However, the default setting for the maximum number of iterations for a minimization calculation may not be sufficient to meet the convergence criterion, particularly for large structures, or ones with poor starting geometries. The latter situation often arises during conformational searching. For this reason, we suggest that all structures located in a conformational search be reminimized with a large value (5000) for the maximum number of iterations and with all the tests for uniqueness in place. This will help ensure that the set of conformations obtained in the conformational search represent unique conformations and not structures that would be duplicates of other structures if completely minimized. The final report for a conformational search highlights with an asterisk any structure that did not converge to the criterion in force during minimization.

2.5.2 Conformational Searching

Conformational searches involving structures with up to a dozen rotatable bonds generally converge easily, but convergence can be problematic with more flexible structures. When performing conformational searches with these more flexible molecules, it is therefore wise to carry out multiple searches with different starting geometries to verify that the same final structures are found.

In addition to ensuring that the final set of conformers obtained from a conformational search is completely minimized, there is another issue of convergence which affects the stochastic searching methods (LMOD, LLMOD, MCMM, and SPMC). For these methods, it is important to establish that the structures obtained in a given search represent a thorough sampling of all possibilities. There is no way to prove that the search was fully convergent, but there are several methods which can at least indicate that it was not.

As the search approaches convergence, the number of new unique conformations which are found should begin to approach zero. That is to say, after the search has been in progress for some time, many, many trials should elapse between the finding of previously unobserved unique structures. In practice, however, this may not be a good test unless all conformations found minimized to low gradients, as described in the previous section.

We generally recommend the following procedure to ensure that a stochastic search has been exhaustive:

- Conduct a search of approximately 1000-2000 MC steps for every variable degree of freedom.
- Ensure all the structures saved are thoroughly minimized as described above.
- Run a second search (possibly “seeded” with the selected structures from the of the first search—see the *MacroModel Reference Manual* for details on how to do this). Ensure that a `SEED` command is used so that a different sequence of random numbers will be used in the search.
- Combine the results of the two searches and reminimize keeping only the unique conformations.
- Repeat this procedure of searching and combining the results until the total number of accumulated conformations (especially those in the low energy region) becomes constant.

In any modeling study, it is up to you to demonstrate that results are converged and what the bounds of uncertainty are. *Unconverged results are meaningless*.

2.5.3 Molecular and Stochastic Dynamics

Obtaining convergence in dynamics simulations has traditionally been problematic, because of the slow frequency at which systems undergoing these processes cross barriers between the various minima on the potential energy surface. Stochastic dynamics may search conformational space more efficiently than does regular molecular dynamics, but neither method gives frequent crossing of barriers much larger that 3 kcal/mol (13 kJ/mol). A much higher rate of convergence can be obtained by the use of the mixed-mode Monte Carlo/Stochastic Dynamics (MCSD) procedure. We therefore recommend using this procedure whenever appropriate, such as for simulations of acyclic systems in which generation of the canonical ensemble is desired and no time-dependent information is required. For cyclic systems, the Jumping Between Wells (JBW) method can be used. However, even with these methods, it is important to test that converged results have been obtained. Converged simulations will satisfy the criteria listed below. Satisfaction of these criteria, like those described in the last section, are necessary, but not sufficient, to prove convergence:

- Average quantities—for example, potential energy, or the fraction of times a hydrogen bond appears—should exhibit numerical stability. For example, once the simulation is converged, doubling the length of the simulation should not change the averages. It is possible to obtain the average values of geometric and hydrogen bond monitors periodically during the simulation.
- Dihedral angle distributions in achiral compounds should be symmetric. For example, dihedral angle distributions produced using the `MDDA` command should exhibit the same populations of +gauche and –gauche values for three-way torsions.
- Simulations starting from different starting geometries should give the same results.

2.6 Problem Size: Limits of Operation

MacroModel is distributed with a number of arbitrary fixed limits on its operation. Some of these restrictions have been lifted since the program has gone to dynamic memory allocation, but some still remain. For example, we distribute a version of MacroModel that can perform Monte Carlo conformational searches on up to 500 rotatable bonds; however, these limits usually far exceed the size of a calculation that can be reasonably performed. It is highly unlikely that a converged conformational search could be performed in a reasonable amount of time on a system with even 100 degrees of rotational freedom. Because of the wide variety of platforms on which MacroModel is used—ranging from Linux-based PCs to super computers or a network of scores of workstations—and also because chemical problems differ greatly, we cannot make categorical pronouncements setting forth the “largest” problem one can reasonably address. Instead, we encourage you to consider the issues of convergence as described below, and use the principles described there as a guide.

2.7 Nonbonded Cutoffs

By default, MacroModel energy calculations are performed with cutoff distances in place for nonbonded interactions. If the distance between any nonbonded pair of atoms is greater than the cutoff distance, then the non-bonded term (electrostatic or van der Waals energies) for that pair will be ignored. These terms decrease in magnitude with increasing interatomic distance, and so terms for nonbonded atom pairs separated by a distance greater than a reasonable cutoff distance will not contribute greatly to the overall energy. The introduction of nonbonded cutoffs greatly reduces the time it takes to compute the energy for a large molecular system. Several implications of using cutoffs, however, deserve special mention.

- For systems with formally charged atoms, the default cutoff distances (7 Å for van der Waals interactions and 12 Å for electrostatics) are probably too small for accurate results. Better values might be 8 Å and 20 Å, respectively. These values can be set by selecting Extended from the Cutoff option menu in the Potential tabs of the MacroModel panels.
- The original MM2 and MM3 programs do not use non-bonded cutoffs. When making comparisons with these programs, MacroModel calculations should be performed using non-bonded cutoff distances which are effectively infinite; that is, the values should be larger than the largest molecular dimension. See Section 2.1 on page 9 for other considerations involving comparisons between MacroModel results and results from the standard implementations of other force fields.
- We recommend that calculations performed using the MacroModel solvation model be carried out with effectively infinite cutoffs.

2.8 Electrostatic Treatments

By default, MacroModel uses a constant dielectric constant during the evaluation of electrostatic interactions between atoms. In most situations the use of a constant dielectric is appropriate for such calculations. One such place is when making comparisons with gas phase ab initio molecular orbital calculations. Additionally, use of the MacroModel GB/SA continuum solvation model also requires the constant-dielectric electrostatic treatment. When this model is utilized, electrostatic interactions are automatically calculated using this method. Alternatively, a distance dependent dielectric electrostatic treatment may be specified by selecting Distance Dependant from the Electrostatic Treatment option menu in the Potential tab of the MacroModel panels.

2.9 File Conversion

Maestro reads structure files in a variety of formats. You can then use these structures in MacroModel calculations. Not all structures are suitable immediately for molecular modeling, and you might have to fix the structure. If you have problems reading a PDB file, see Section 3.1.7 of the *Maestro User Manual*, which explains Maestro's color-coded warning system. It is highly recommended to use the Protein Preparation Wizard to prepare protein structures for molecular modeling. File conversion can also be performed from the command line using the `pdbconvert`, `sdconvert`, `molmmod`, and `mmodmae` scripts in the `$SCHRODINGER/utilities` directory.

2.10 Modified Force Field, Solvent, or Atom Type Files

When a job is launched, MacroModel searches in the current working directory for a copy of the force field (`*.fld`), solvent (`*.slv`), and atom type (`atom.typ`) files. If it finds any of these files there, it uses them to perform the energy calculation. If MacroModel does not find copies of these files locally, it then looks in `$HOME/.schrodinger/macromodel`. Only if MacroModel does not find the files in `.schrodinger` does it use the global settings in `$SCHRODINGER`. If you do create locally modified files, manage them closely. The unintentional use of forgotten locally modified files can give surprising energetic results.

2.11 Hydrogen Treatment

2.11.1 Hydrogens on Hetero Atoms

While the AMBER* and OPLS force fields can be used without explicit hydrogen atoms on carbon atoms, all force fields require the use of explicit hydrogen atoms on hetero atoms. Note also that AMBER* requires explicit lone pairs on sulfur. Original MM2 implementations have lone pairs on sp^3 oxygen; however, we recommend removing them for best results in MM2*.

In Maestro, you can modify a structure's hydrogen treatment using settings in the Add Hydrogens - Advanced panel, which you open from the Edit menu. See Section 5.10 of the *Maestro User Manual* for more information.

2.11.2 Hydrogen Treatment in the AMBER* Force Field

The AMBER* force field file contains charge sets appropriate to either united-atom or all-atom representations; however, by default, the united-atom charges are used. This means that even those carbons having explicit hydrogens attached will, by default, get united-atom charges. In practice, this makes little difference in energetic results, but in some situations—for example, when making comparisons with the original AMBER program, and also when calculating absolute solvation free energies—it is important to use all-atom charges if such carbons exist. This can be done by selecting Force Field Defined from the Electrostatic Treatment option menu in the Potential tab of the MacroModel panels.

When using the united-atom models in AMBER* or OPLS*, the best results are obtained when explicit hydrogen atoms are used on sp^2 carbons. The command:

```
$SCHRODINGER/utilities/applyhtreat in.mae out.mae
```

where *in*`.mae` is the name of a file containing structures to which hydrogen atoms should be added produces a file called *out*`.mae` containing the corresponding structures containing the additional hydrogen atoms.

Running MacroModel From the Command Line

Running MacroModel from the command line gives you the ability to customize batch processing. In addition, some types of MacroModel calculations are not supported by the Maestro GUI. This chapter gives instructions on how to launch MacroModel jobs from the command line. On Windows, you can open a Schrodinger Command Prompt window from the Start menu, and use this window to run command-line jobs. See Section 1.6 on page 4 for more information.

3.1 Preparing for MacroModel Calculations

Before you can submit MacroModel jobs from the command line, you must have a valid input structure file and a command (*jobname*`.com`) file. On Linux and Mac, you must also set the `SCHRODINGER` environment variable; on Windows the `SCHRODINGER` environment variable is already set in the Schrodinger Command Prompt window.

3.1.1 Environment Variables

The `SCHRODINGER` environment variable specifies the installation directory for your Schrödinger software. These files include the force-field files, the solvent files, and the `atom.typ` file. To run MacroModel as a stand-alone application, this environment variable must be defined. See Section 1.6 on page 4 for more information on setting `SCHRODINGER`. Whether you start MacroModel from the UNIX shell or from Maestro, local versions of the force-field, solvent, and `atom.typ` files override versions in the location specified by `$SCHRODINGER`.

You might also want to specify the default location for temporary (scratch) files. The default scratch directory is set in a `schrodinger.hosts` file, which should exist on each host on which you want to run MacroModel. This file can be in a local directory, in your Schrödinger resources directory, or in the installation directory. See Section 7.1 of the *Installation Guide* for more information on the hosts file. You can override the default temporary directory by setting the `SCHRODINGER_TMPDIR` environment variable.

For information on setting environment variables on Windows, see Appendix A of the *Installation Guide*.

3.1.2 The Command File

Every time job files are written or a job is started from Maestro, a MacroModel command file and a structure file are created. The command file contains all necessary information for MacroModel to perform the job as specified. Most MacroModel tasks can be executed from Maestro. However, there are a few tasks that still involve manually setting up and executing a command file.

In this chapter, examples of common command files are given that are set up through Maestro and provide some insight to the different commands included in these files. At the end of the chapter, more advanced command files are presented, including tasks that can not currently be set up from Maestro.

Many of the commands presented in the command files below have additional arguments available. For more information on any of the commands, refer to the *MacroModel Reference Manual*.

An alternative simplified input file is available for certain kinds of calculations. Jobs using this simplified input file can be run with the `$SCHRODINGER/macromodel` command. For details, see Chapter 3 of the *MacroModel Reference Manual*.

3.1.3 The Command File Format

To run MacroModel calculations, a molecular structure file and a command file are required. The molecular structure file contains the structures to be used as input in the calculation. The command file contains the name of the input structure file, the name of the output structure file, and an ordered list of operation codes, or *opcodes*, for the calculations.

A generalized form of a MacroModel command file is given below:

```
jobname.mae
jobname-out.mae
 MMOD       0      1      0      0     0.0000     0.0000     0.0000     0.0000
 FFLD      10      1      0      1     1.0000     0.0000     0.0000     0.0000
 BDCO       0      0      0      0    20.0000 99999.0000     0.0000     0.0000
 READ       0      0      0      0     0.0000     0.0000     0.0000     0.0000
 CONV       2      0      0      0     0.0500     0.0000     0.0000     0.0000
 MINI       1      0    500      0     0.0000     0.0000     0.0000     0.0000
```

This file must have a name in the form of *jobname*`.com`, where *jobname* is replaced with the actual name of the job. Thus, a job that was actually named “jobname” could be submitted from the command prompt with a command such as

```
$SCHRODINGER/bmin jobname
```

Maestro uses a similar command for computations prepared with the interface. The first line of the instruction file above is the name of the input structure file. The input structure file can be named with any valid UNIX filename. The resulting output structure file is given the name listed on the second line of the instruction file. The full path to the structure files may be given if the files are not in the current directory.

In addition, any input substructure (`.sbc`) or velocity (`.vel`) files should contain the same prefix as the input structure file. Similarly the output energy listing (`.mmo`), substructure, and dihedral drive (`.grd`) files have the same base name as the output structure file. The *jobname*`.log` file contains text messages tracing the progress of the job. Experienced users of MacroModel should be aware that this mechanism has not changed from previous versions of MacroModel.

There are some default file naming changes, indicated in the sample file above, that are used by the Maestro interface due to the advent of Maestro formatted structure files. Although not mandated, to be consistent, all Maestro formatted structure files are given the suffix `.mae`. This is a different default behavior from the former MacroModel user interface and the older MacroModel structure format. Previously, input files were given the names *jobname*`.dat` and *jobname*`.out`. Now that all Maestro formatted structure files are, by default, named with the `.mae` extension, Maestro automatically names the input structure file *jobname*`.mae` and the output structure file *jobname*`-out.mae`.

The remaining lines in the command file provide the instructions to MacroModel concerning the type and order of calculations to be performed. The opcode lines must be of the following fixed format:

```
#OPCD 123456 123456 123456 123456 FFFFF.FFFF FFFFF.FFFF FFFFF.FFFF FFFFF.FFFF
```

Each opcode has four letters, and is preceded by a blank space. A specific opcode can be ignored (commented out) in a command file by placing a character other than blank space before the opcode. The eight fields after the opcode are referred to as *arguments*, and are often referred to as simply as “arg”s. Arg1 through arg4 are integer arguments (Fortran I6 format), while arg5 through arg8 are floating point arguments (Fortran F10.4 format). The opcodes indicate individual energetic calculations or options and the arguments indicate additional options or quantify the parameters of the calculation. Many arguments have default values, which are indicated by a value of zero as the argument. Thus is it unnecessary to indicate explicitly all arguments in the instruction file if the default values are sufficient. The default values are included with the opcode descriptions in the *MacroModel Reference Manual*. It is important to strictly adhere to the format of the command file. We recommend using an existing command file as a template, rather than to build one from scratch. Tabs are not allowed in MacroModel instruction files.

3.1.4 The Simplified Input File

MacroModel also has a simplified input file, in which the parameters are set with keyword-value pairs in free format. This file can be used for some but not all MacroModel calculations, which you start with the `$SCHRODINGER/macromodel` command. For details, see Chapter 3 of the *MacroModel Reference Manual*.

3.2 Submitting Jobs From the Command Line

The `bmin` script located in the `$SCHRODINGER` directory should be used to launch MacroModel jobs from any directory, once the `SCHRODINGER` environment variable has been set. To launch a job, enter the following command in a terminal window:

```
$SCHRODINGER/bmin [options] jobname
```

replacing *jobname* with the stem of the input file name (not needed if the `-TEST` or `-HELP` options are specified).

The options for the `bmin` command are listed in Table 3.1, and include options for distributed processing. The standard Job Control options are also supported—see Section 2.3 of the *Job Control Guide*. Additional diagnostic options, which do not run the job but provide information, are also given in the section mentioned. In particular, you should note the syntax of the `-HOST` option, which is used to specify the list of hosts used for the job.

Table 3.1. Options for the bmin command.

Option	Description
`-DEBUGGER` *debugger*	Run `bmin` in the foreground with a debugger called *debugger*.
`-DEBUG2`	Print out debugging information from the scripts used to start `bmin`.
`-first` *firstlig*	First ligand to include. Default 1.
`-h[elp]`, `-HELP`	Show usage message.
`-HOSTFILE` *hostfilename*	The name of the hosts file to use for this run. Default is `schrodinger.hosts`.
`-j` *subjobs*	Only run the specified subjobs. The subjobs list can be a comma-separated list of subjob numbers or ranges in the format *n*`:`*m*. If spaces are included, the list must be quoted. Example: `1:5,14,18`. Default: run all subjobs.
`-JOBCTS` *number*	Maximum number of structures (CTs) in any subjob. Default: 10000.

Table 3.1. Options for the bmin command. (Continued)

Option	Description
`-last` *lastlig*	Last ligand to include. Default: last ligand in the file.
`-MAX_RETRIES` *number*	Maximum number of times a subjob is retried before it is regarded as failed.
`-nc`|`-NC`	Do not clean up by removing intermediate files.
`-NJOBS` *jobs*	Divide the job into at least the specified number of subjobs. The specified number is increased if the limit on the number of structures per subjob is exceeded. The number of subjobs can be greater than the number of processors used. For efficient load balancing the number of subjobs should be several times the number of processors. Default: 1.
`-NOJOBID`	Do not run the job under Job Control, and send output to standard output.
`-nx`	Do not run jobs, but create specified subjob directories and input structure files locally.
`-OUTPUT_ORG` *option*	Produce more than one output structure file. If *option* is `BY_SUBJOB` then produce one output file for each subjob. Otherwise, create a new directory called *option* and create separate output files for each input ligand within this new directory.
`-RESTART`	Restart the previously failed parent job, running only the subjobs that did not finish. This facility works only for local jobs (i.e. jobs started with `-LOCAL`).
`-STRICT_END`	Terminate the job if any of the subjobs fails.
`-SUBHOST` *host-list*	Specify the hosts for the subjobs. *host-list* is a list of one or more hosts. The list must be quoted if multiple hosts are specified: for example, `"hostname1:nprocs1 hostname2:nprocs2..."`. If `-HOST` is also used, it specifies only the host for the driver. If `-SUBHOST` is not used, then `-HOST` specifies the hosts for the subjobs, and the driver is run on the first specified host.
`-TEST`	Run the standard test suite (jobname not needed).
`-DEV_TEST`	Run the developers version of the standard test suite (jobname not needed). Turns on `-TEST`.

Launching a MacroModel job using the `bmin` script creates the output structure file and the *jobname*`.log` file. The `.log` file contains MacroModel job information, such as job number and timings, and any system errors, as well as output from the job.

3.3 Running Remote and Distributed Calculations

Remote MacroModel jobs run on a different host from the one on which the command is entered to start the job. Distributed MacroModel allows certain types of calculations to be run by distributing tasks across a number of different hosts or processors. Remote and distributed MacroModel jobs can be run from Maestro. Distributed jobs can be run locally on Windows hosts, but remote jobs must be submitted to Linux hosts.

Schrödinger's Job Control facility controls both remote and distributed MacroModel jobs. If you intend to run jobs on various hosts, you must set up the hosts for remote access and provide information on the hosts to the Job Control facility through a hosts file, named `schrodinger.hosts`. Instructions for setting up the required information on remote hosts and for configuring batch queues is given in Chapter 7 of the *Installation Guide*.

Any MacroModel job can be run remotely. The normal Linux commands are used to run the job, with additional information specifying which host from the hosts file to use:

`$SCHRODINGER/bmin -HOST` *remote-host-name jobname*

MacroModel can divide some types of calculations into "tasks", either internally (inside `bmin`) or externally (in the job driver). These tasks may be distributed over a number of processors to reduce the calculation time. Calculations that can be distributed using MacroModel include many of the conformational searching methods as well as Free Energy Perturbation (`FEAV`/`FESA`) and Embrace calculations. For a thorough description of the distributed MacroModel calculations, including the specific types of calculations supported, see the Section 2.4 of the *MacroModel Reference Manual*. Below are two simple examples that illustrate how to perform these calculations.

For distributing calculations focused on a single input structure, internal distributed calculations are used. For example, to distribute a conformational search of a protein-ligand complex across two processors, add the line:

```
NPRC      2     20    10     1     0.0000 0.0000 0.0000 0.0000
```

just after the `MMOD` command in the example `.com` file given in Section 8.3.3 on page 94. The job can then be started just like any other MacroModel job:

`$SCHRODINGER/bmin` *jobname*

A type of job that can be distributed externally is MCMM serial searches, in which each input structure is subjected to a separate search (see Section 8.3.2 on page 93). You can do this with the following command:

```
$SCHRODINGER/bmin -NJOBS 5 -HOST "comp1:1 comp2:2" serial-lmcs
```

This command divides the job into five tasks, all of which are run on hosts `comp1` or `comp2`. These hosts must be described in the `schrodinger.hosts` file. At any one time, up to one task is running on `comp1` and up to two tasks are running on `comp2`.

For distributed MacroModel calculations that involve a random or accumulated aspect (e.g. MCMM conformational searches) the results may differ from those obtained from an equivalent non-distributed calculations. However, the different results from either type of calculation are valid.

3.4 Interacting With a Running MacroModel Job

The Schrödinger Job Control facility allows you to interact with running MacroModel jobs. Because all jobs submitted, either from Maestro or the command line, are controlled by the Job Control facility, all jobs can be monitored using the **Monitor** panel in Maestro or using the `jobcontrol` script from the command line. For specific instructions on using the **Monitor** panel or the `jobcontrol` script, see Chapter 3 of the *Job Control Guide*.

General Settings

4.1 General MacroModel Calculation Setup

All Maestro energetic calculation panels have the same basic setup process:

1. Specify the structure input source from the Use structures from option menu (see Section 4.1.1 below).
2. Specify settings in the tabs:
    - For the Potential tab, see Section 4.2 on page 32
    - For the Constraints tab, see Section 4.4 on page 37
    - For the Substructure tab, see Section 4.5 on page 40
    - For tabs specific to the energetic calculation, see the appropriate chapter
3. Click Run to start the job with the current job settings, click the Settings button to set up and launch the job, or choose Write from the Settings button menu to write the files for future use (see Section 4.6 on page 43).

4.1.1 Specifying the Structure Input Source

For many MacroModel jobs run from the Maestro interface, you can use as input either the structures included in the Workspace or the entries selected in the Project Table. If you have a file containing the structures of compounds you want to use, first import the structures into a project using the Import panel, and then select the desired structures in the Project Table. Other MacroModel calculations, such as the minimization panels, allow you to use an input file directly, without having to import the structures into the Project Table.

The default setting for MacroModel structure input is Workspace (included entries). If you want to perform MacroModel calculations only on the structures currently appearing in the Workspace, you do not need to change the Use structures from setting.

If you want to use Project Table entries as input, change the Use structures from setting to Project Table (selected entry) or Project Table (selected entries). When using this setting, make sure that you have selected the desired entries in the Project Table panel.

If the panel supports direct use of a structure file, select File, and then click Browse to open a file selector so that you can navigate to and choose the file. You can also type in the file name directly, in the File name text box.

4.2 The Potential Tab

The Potential tab appears on all of the MacroModel energy panels. This tab contains tools for specifying a force field, a solvent treatment, and an electrostatic treatment for the calculation. The Potential tab is displayed by default when the energetic panel is opened.

Selecting a Force Field: To specify a force field for a MacroModel calculation, select the desired force field from the Force field option menu. The default force field is OPLS_2005. For information about MacroModel implementation of the supported force fields, see Section 2.1 on page 9.

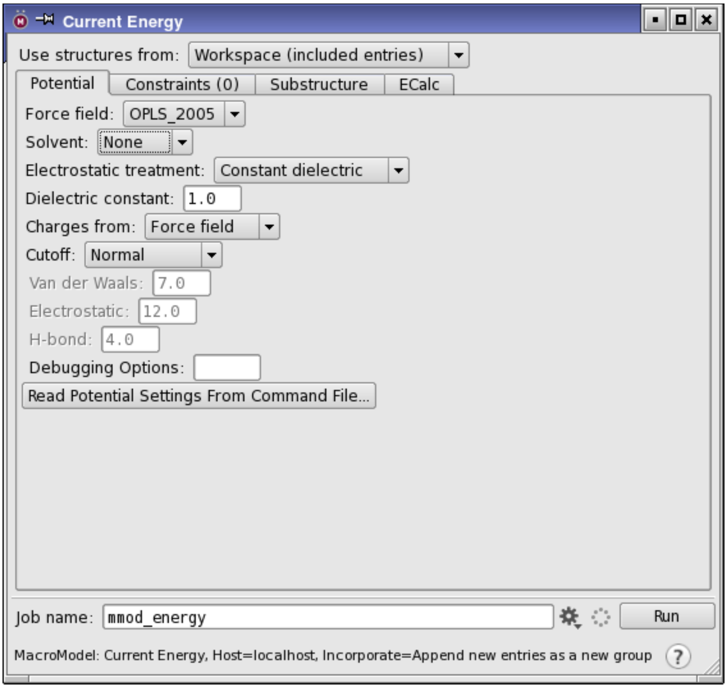
Selecting a Solvation Treatment: MacroModel calculations are run with water as solvent by default, but you may carry out the calculation in the gas phase or with another solvent. MacroModel provides the GB/SA continuum solvation treatment, and water, octanol, or chloroform may be chosen. To change the solvent, select the name of the desired solvent from the Solvent option menu. The dielectric constant for the solvent is set with the solvent choice.

Selecting an Electrostatic Treatment: The default electrostatic treatment for all built-in force fields is to use a dielectric constant of 1.0. You can also select a distance-dependent dielectric from the Electrostatic treatment option menu. You can set the dielectric constant; the default value is 1.0, which corresponds to Coulomb's law in a vacuum. Selecting the force field defined treatment is the same as choosing a constant dielectric with a value of 1.0. The MMFF and MMFFs force fields use a "buffered" constant dielectric treatment; see Halgren [9,10]. If you choose a solvent model, the electrostatic treatment is fixed and you cannot change it.

Using Charges From a Structure File: The charges used in the electrostatic portion of an energy calculation can either be assigned by the force field or obtained from the structure. Regardless of the source, charges are written to the structure file when a job is started. By default, the Force field option is used. To use charge information from the structure, select Structure file from the Charges from option menu.

Choosing a Nonbonded Cutoff Setting: The maximum distances over which hydrogen bonding, van der Waals, and electrostatic contributions to the molecular potential energy are computed may be controlled using the Cutoff option menu. "Normal" cutoff distances are 4 Å for hydrogen bonding, 7 Å for van der Waals and 12 Å for electrostatics, whereas "extended" cutoff distances are the same for H-bonds, 8 Å for van der Waals and 20 Å for electrostatics. The van der Waals and electrostatic cutoff distances are the center of a soft cutoff that starts at 1 Å smaller than the specified distance and ends at 1 Å larger than the specified distance.

Using Potential Settings from an Existing Command File: To read potential energy settings from a MacroModel command file into the Potential tab, click Read Potential Settings From Command File. Setting up MacroModel calculations in this way is helpful because you can easily reproduce your potential energy settings for multiple calculations.



Use structures from: Workspace (included entries)
Force field: OPLS_2005
Solvent: None
Electrostatic treatment: Constant dielectric
Dielectric constant: 1.0
Charges from: Force field
Cutoff: Normal
Van der Waals: 7.0
Electrostatic: 12.0
H-bond: 4.0
Debugging Options:
Job name: mmod_energy
MacroModel: Current Energy, Host=localhost, Incorporate=Append new entries as a new group

Figure 4.1. The Potential tab.

Setting DEBG Options: If you want to add `DEBG` options, you can do so in the Debugging Options text box, by listing the numbers of the `DEBG` options you want to use, separated by either commas or spaces. The `DEBG` opcodes are added at the beginning of the command file.

4.3 Fixing and Freezing Atoms in Large Systems

Two important reasons for imposing constraints on molecular systems are to force the system to meet desired geometric conditions, and to reduce the cost of the calculation by eliminating interactions that are expected to have little influence on the results. When modeling molecular systems, you might want to constrain some of the degrees of freedom, such as atom positions or distances, angles and dihedral angles. Constraints are useful in the following situations:

- When it is inherently hard to get the system to adopt the desired geometries in the absence of constraints
- When the potential functions are inadequate for the system
- When parts of the system that would normally lead to the appropriate geometries are not represented

When modelling a large system, it is often desirable to focus calculations on important regions and ignore any large portions of the system from which no significant influence is expected. For example, in the minimization of a protein-ligand complex, the region of interest is the ligand and the area surrounding the ligand. The large outer regions of the protein may not significantly influence the structure of the minimized ligand or its vicinity, and these regions may therefore be excluded from the calculation. Fixing or freezing remote portions of the system can help further reduce computational time and provide buffer zones between fully moving and ignored regions.

Frozen atoms cannot move, while fixed atoms are restrained to lie close to the requested positions. Moving atoms interact with all other moving, fixed and frozen atoms in the usual way. However, among the fixed or frozen atoms only stretch interactions involving fixed atoms are considered. Eliminating most of the interactions among the fixed or frozen atoms can save a great deal of time and is a useful approximation provided that the potentials restraining the fixed atoms are strong enough to prevent dramatic distortions of the molecular structures.

4.3.1 Methods for Freezing and Fixing Atoms

In MacroModel there are two ways of constraining parts of a molecular system:

- **Using constraints:** Manually fixes atoms (and also distances, angles, or torsions) through the use of `FXAT` (or `FXDI`, `FXBA`, or `FXTA`) commands, either using the Constraints panel in Maestro or by directly modifying a command file. Only the geometry of the items explicitly specified is affected. All other parts of the system interact and move normally.
- **Using substructures:** Uses Maestro's substructure facility to define, within a molecular system, flexible substructures and fixed and frozen shells. You can also directly modify a command file to create or modify substructures. Atoms that are not in the substructure or specified as fixed or frozen are ignored.

To constrain only a few atoms (the first method) using `FXAT` commands is best, and is also easily achieved through point-and-click operations in Maestro. To constrain a larger number of atoms, you will find it is much more efficient to use substructures.

Note: Constraints can be violated if you use automatic setup (see Section 8.2.2 on page 79). If so, MacroModel will fail.

With either method, a positive force constant can be specified for the restraining harmonic potential for fixed atoms. Free movement of the atoms within a specified range can be facilitated with the use of flat-bottomed potentials in which the two halves of the harmonic potential are separated by a specified distance within which the potential is zero.

4.3.2 Freezing and Fixing Individual Atoms

Individual atoms can be constrained either by picking the desired atoms using the tools in the Constraints tab of the MacroModel panels within Maestro, or by using the `FXAT` command in a *jobname*`.com` file.

Atoms are fixed or frozen in their original position by using tools in the Constraints tab within the various MacroModel energy panels in Maestro. The desired atom is selected through on-screen picking, and is then assigned values for how much it should be allowed to move (half-width, in Å) and the penalty for moving outside the defined limits (force constant, in kJ/mol-Å^2). This process is repeated for all atoms to be constrained. See Section 4.4 on page 37 for more information on specifying constraints using Maestro.

For additional information about using the `FXAT` command when manually constructing MacroModel command files (*jobname*`.com`), refer to the pages describing the `FXAT` opcode in the *MacroModel Reference Manual*.

All atoms not defined by a `FXAT` command remain fully flexible, free to sample the potential energy surface defined by the force field.

4.3.3 The Substructure Facility

You can define substructures by manually placing the appropriate lines in the MacroModel command file (*jobname*`.com`), or by using the Substructure tab in the MacroModel energy panels to define groups of flexible, fixed, and frozen atoms. The settings are saved in a substructure file consisting of `SUBS` commands for the flexible atoms and `FXAT` commands for fixed/frozen atoms.

Note that atoms not defined by either `SUBS` or `FXAT` commands are completely ignored (as if they had been deleted) during the calculation.

After defining the flexible part of the system (typically the ligand and atoms in its near vicinity), you can define a combination of fixed/frozen atom shells (having different force constants) around the movable part. These shells are defined based on a user-selected radius (in Ångstroms) from the flexible part of the system. To avoid cutting amino acid residues in two by the radius definition, Maestro offers a Fill residues option that ensures that if any atom of a residue is within the defined radius, the full residue is included in the shell. Additionally, there is an option for adding isolated atoms to any shell. Each shell can be assigned a different force constant (for a fixed shell), or can be completely frozen (force constant takes a negative value).

Looking at a protein-ligand complex as an example, the flexible part is most easily defined by picking. By choosing Molecules from the Pick menu in the Freely moving atoms (substructure) section, you can select an atom in the ligand to define the whole ligand as part of the substructure. Atoms belonging to the substructure are identified by white markers on screen. If parts of the active site are to be included in the substructure as well, selected residues can easily be picked by choosing Residues from the Pick menu and clicking on an atom in each residue to be included. Alternatively, a shell of flexible residues can be defined using the Atom Specification Language (ASL) in the ASL text box or in the Atom Selection dialog box. Assuming that the ligand is molecule number one and the protein is molecule number two, the following command in the substructure definition field places the ligand and a 3.5 Å shell of filled out residues in the substructure part:

```
m. 1 or fillres within 3.5 m. 1
```

The different shells defined around the substructure have markers of different colors (orange, purple, etc.), in order to easily visualize the various selections on screen.

After the substructure and shell selections are made, the substructure (*filename*`.sbc`) file can be written to disk. The name of the substructure file must match the input file. The substructure file has the following format:

```
Command    arg1 arg2 arg3 arg4  arg5 arg6 arg7 arg8
   SUBS    1    2    3    4       0. 0.   0.   0.
   SUBS    5    6    7    8       0. 0.   0.   0.
   [...]
   FXAT    123  0    0    0     200. 0.   0.   0.
   [...]
   FXAT    124  0    0    0      -1. 0.   0.   0.
   [...]
```

For the `SUBS` part of the substructure file:

arg1-4: Atom numbers of atoms included in the substructure (flexible) part of the system.

arg5-8: Not normally used.

For the `FXAT` part of the substructure file:

arg1: Atom number of atom to be included in fixed/frozen part of the system.

arg2-4: Not used.

arg5: Force constant (kJ/mol Å^2) used for atom listed in arg1. If the force constant is negative, the atom is frozen.

arg6-8: Desired X, Y, Z coordinates of atom defined in arg1. If all are zero (the default), the starting coordinates are used.

Note that the `SUBS` command in the MacroModel command file takes on two meanings, depending on whether or not a substructure file is to be used. If arg1 of `SUBS` is zero in the command file, MacroModel looks for a substructure file to obtain information on flexible and fixed/frozen parts of the system. There are other possible uses of the `SUBS` command: in particular it may be useful to combine the use of substructures both in a *filename*`.sbc` file and separate `SUBS` lines in the MacroModel command file. Please refer to the `SUBS` command in the *MacroModel Reference Manual*.

The substructure file also supports the use of ASL expressions, with the commands `ASL1`, `ASL2`, and `ASL3`. The syntax of their use is described in the *MacroModel Reference Manual*. `ASL1` performs the same function as `SUBS` to specify atoms for a substructure; `ASL2` performs the same function as `FXAT`, to specify atoms to be fixed or frozen, and `ASL3` supports restraining atoms in the ligand for Embrace calculations.

For more information on using Maestro's substructure facility, see Section 4.5 on page 40.

4.4 The Constraints Tab

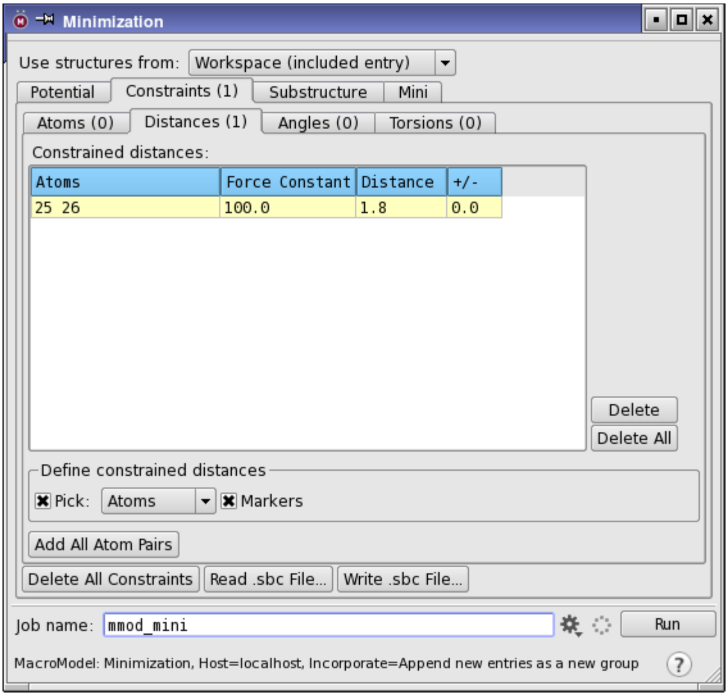
In MacroModel, subsets of atoms can be constrained in almost all energetic calculations, and the constraints can be of two basic types. The atoms can be completely frozen and not allowed to move, or atoms can be fixed, which means that the atoms in question are tethered in place using a constraint. Fixed atoms have the ability to move, frozen atoms do not. The following two sections describe how to prepare MacroModel computations from Maestro that include fixed and frozen atoms. For more information, see the *MacroModel Reference Manual* sections on of `FXAT`, `FXDI`, `FXBA`, `FXTA`, and `SUBS`.

Making selections in the Constraints tab adds the appropriate opcode entries to a separate input substructure file, usually named *jobname*`.sbc`. This mechanism allows large numbers of constraints to be shared between different energetic jobs for the same structural system. You can also read and write substructure files containing the constraints with the Read and Write buttons in the Constraints tab. For more information about the Substructure Facility, see Section 4.3 on page 33. Structural constraints can be added in both the Constraints tab and the Substructure tab for a single computation.

Note: Constraints can be violated if you use automatic setup (see Section 8.2.2 on page 79). If so, MacroModel will fail.

4.4.1 Constraining Atoms, Distances, Angles, and Torsions

Individual atom positions, bond distances, bond angles, and torsion angles can all be constrained from the Constraints tab. This tab has subtabs for each of the structural elements you can constrain, which you can use to pick the desired elements and define the relevant settings. The tools in the tabs for defining constrained atoms, distances, angles, and torsions are generally the same. Exceptions are noted below.


Minimization
Potential
Constraints (1)
Substructure
Mini
Atoms (0)
Distances (1)
Angles (0)
Torsions (0)
Delete
Delete All
Define constrained distances
Add All Atom Pairs
Delete All Constraints
Read .sbc File...
Write .sbc File...
Run

Use structures from: Workspace (included entry)

Constrained distances:

Atoms	Force Constant	Distance	+/-
25 26	100.0	1.8	0.0

☒ Pick: Atoms ☒ Markers

Job name: mmod_mini

MacroModel: Minimization, Host=localhost, Incorporate=Append new entries as a new group

Figure 4.2. The Constraints tab.

Specifying atoms to be constrained: To constrain an atom, distance, angle, or torsion, display the appropriate tab: Atoms, Distances, Angles, or Torsions. Choose a structural unit from the Pick menu and click on the desired atoms in the Workspace. The picked atoms appear in the list in the upper portion of the tab.

To constrain many atoms simultaneously, you can use an ASL expression. ASL expressions can be built using the Atom Selection dialog box, which you open by clicking the Atom Selection button and choosing Select.

For more information on ASL, see the online help or the *Maestro Command Reference Manual*.

If you want to constrain all of the distances, angles, or dihedrals, click the Add All button at the bottom of the relevant tab. You can also use this approach to constrain most of these parameters, by constraining all of them and then deleting constraints for the parameters that you want to vary, by selecting them and clicking Delete.

Specifying the force constant and atom flexibility: By default the constrained structural elements have harmonic potentials with a force constant of 100 kJ/mol-$Å^2$ with the current value of the element used for the potential minimum. The force constant can be modified. To change a single value, edit the table cell. To change the value for multiple constraints, select the table rows, then edit one of the table cells and press CTRL+ENTER (⌘ENTER).

The value of the distance, angle, or torsion can also be set in the relevant table column. This allows you to force the parameter to a value other than its input value.

For fixed distances, angles, and torsional angles, the position of the potential minimum can be adjusted. A flat-bottom potential, essentially a harmonic potential with the two halves separated by a region of zero potential, can be requested by specifying the half width for the flat region in the +/- column.

Specifying a nondefault force constant and half width prior to selecting atoms: If atom-centered constraints are being applied, constraint parameters can be set up beforehand with the Maestro command `constrainedset`. For example, to set the force constant to 50 and the width to 0.8, enter the following command in the Commands text box of the main window:

```
constrainedset constant=50 width=0.8
```

Constrained atoms that are subsequently selected use these constraint parameters.

Removing constraints: To remove one constraint, select it from the list at the top of the specific constraints panel and click Delete. To remove all constraints of a particular type, click Delete All in the corresponding constraints panel. To remove all constraints of all types, click Delete All Constraints.

Reading and writing constraints: To write out constraints to a substructure file, click Write .sbc File, and enter the file name in the file selector that opens. To read in constraints that were defined previously, click Read .sbc File and select the substructure file that contains the constraints.

4.4.2 Freezing Atoms

“Frozen” atoms have no forces acting on them and do not move from their initial position in the structure. Frozen atoms do influence other, non-frozen atoms in the structure, mostly through nonbonded interactions. Frozen atoms differ from constrained atoms in that constrained atoms may move slightly.

To freeze atoms, select them in the Atoms tab, then select the check boxes in the Frozen column of the table. To freeze multiple atoms simultaneously, select the rows in the table, and then control-click the check box in the Frozen column for one of the rows. To unfreeze an atom, click its check box again (or use control-click for multiple atoms).

4.5 The Substructure Tab

The Substructure tab is used to set the definition of a substructure and any shells of constrained or frozen atoms that surround it.

If any substructure or shell atoms are specified, an input substructure file, *jobname*`.sbc`, is created and used. All instructions relating to the fixed and frozen atoms are entered in the input substructure file, and read from this file when the calculation is launched.

4.5.1 Defining a Substructure

You can select the atoms you want to include in a substructure definition by picking the desired atoms, chains, residues, molecules, or entries from the Workspace structure. To do this, select the structural unit you want to use from the Pick menu in the Freely moving atoms (substructure) section, and then pick atoms in the Workspace. If Markers is selected, the atoms in the substructure are marked with white markers.

In addition to defining substructure atoms by picking them from the structure, you can specify many atoms simultaneously by entering an appropriate ASL expression in the ASL text box in the Freely moving atoms (substructure) section. For example, the expression:

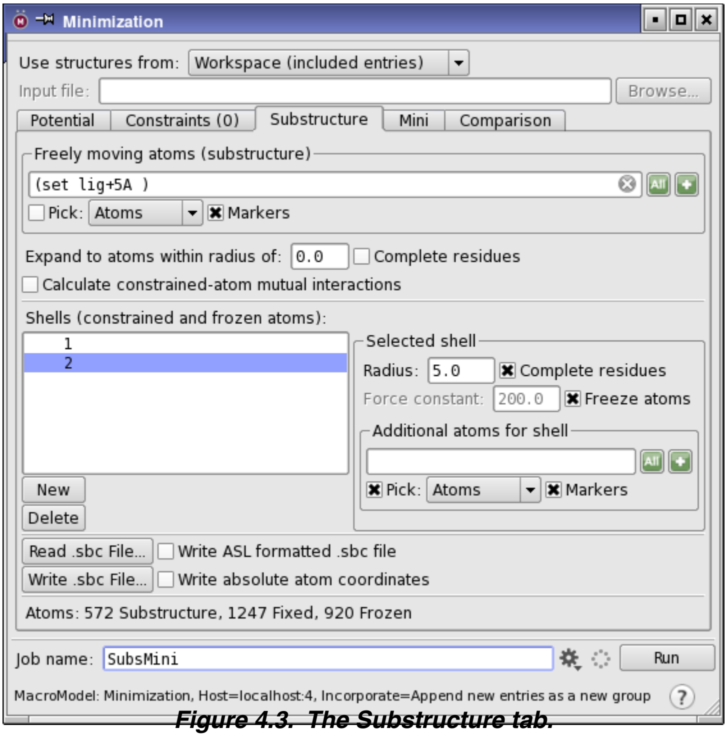
```
fillres within 6.0 mol.n 3
```

picks all the atoms within 6 Å of molecule number 3, and also adds to the selection the remainder of any residues partially selected using the 6 Å proximity criterion. You can also construct ASL expressions using the Atom Selection dialog box. To open this dialog box, click the Atom Selection button and choose Select.

Once you have picked atoms from the Workspace structure or have specified substructure atoms using ASL, you can expand the “shell” of atoms to a specified distance from the defined substructure atoms. To expand the shell, enter the desired radius in the Expand to atoms within radius of text box.

To ensure that the substructure consists of only complete residues, select Complete residues. The defined substructures then include all the atoms in the residues to which the defined atoms belong.

Interactions between constrained atoms (both fixed and frozen) are normally not calculated, which saves time but excludes these interactions from the final energy. You can include these interactions in the calculation by selecting Calculate constrained-atom mutual interactions.



Use structures from: Workspace (included entries)

Input file:

Freely moving atoms (substructure)

(set lig+5A)

☐ Pick: Atoms ☒ Markers

Expand to atoms within radius of: 0.0 ☐ Complete residues

☐ Calculate constrained-atom mutual interactions

Shells (constrained and frozen atoms):

1
2

Selected shell

Radius: 5.0 ☒ Complete residues

Force constant: 200.0 ☒ Freeze atoms

Additional atoms for shell

☒ Pick: Atoms ☒ Markers

☐ Write ASL formatted .sbc file

☐ Write absolute atom coordinates

Atoms: 572 Substructure, 1247 Fixed, 920 Frozen

Job name: SubsMini

MacroModel: Minimization, Host=localhost:4, Incorporate=Append new entries as a new group

Figure 4.3. The Substructure tab.

4.5.2 Creating a Shell of Atoms

Defined substructures indicate which atoms may freely move during the energy calculations. Often, subsequent sets of atoms surrounding the substructure are intended to be fixed and/or frozen during the computation. This is accomplished by defining shells around the substructure in the Shells (constrained and frozen atoms) portion of the Substructure tab. To create a shell around a defined substructure, click New in the Substructure tab of the relevant MacroModel panel, under the Shells list. If Markers is selected, the shell atoms are indicated in the Workspace by colored markers: orange, purple, yellow, green, and so on.

Once you have created one or more shells, you can define and modify their contents in the following ways, by selecting a shell from the Shells list and using the controls in the Selected shell section:

Specifying a Shell Radius: You can assign a radius by entering a value into the Radius text box. The shell includes all atoms within the given radius of the previous shell, or the substructure if it is the first shell.

Picking Additional Shell Atoms: To add individual atoms to a shell, select a structural unit from the Pick menu, and pick Workspace atoms. You can also add shell atoms by entering an ASL expression in the Additional atoms for shell section, or by constructing an ASL expression in the Atom Selection dialog box, which you open with the Select button. Atoms may be added even if a substructure has not been defined.

Setting the Force Constant for a Shell: Shell atoms that are to be constrained rather than frozen are allowed to move a finite distance based on the force constant value you supply. Enter the desired force constant in the Force constant text box.

Freezing Atoms in a Shell: To freeze atoms in a shell so that the atoms cannot move at all, select Freeze atoms.

Filling Out Residues in a Shell: You can include in the shell all atoms in residues having any members in the original shell definition by selecting Complete residues.

Deleting a Shell: To delete an existing shell, select the shell number in the Shells list and click Delete.

4.5.3 Reading and Writing Substructures

You may want to write your substructure definitions to a separate file so that they can be used in subsequent calculations. To write a substructure file, click Write .sbc File, specify the desired file name, and click OK. Include the `.sbc` suffix in the name. You can choose to include the ASL expressions in the substructure file by selecting Write ASL formatted .sbc file. For more information on use of ASL in substructure files, see page 68 of the *MacroModel Reference Manual*.

The atom coordinates written to the substructure file can be absolute coordinates or relative coordinates. If you select Write absolute atom coordinates, writes the absolute atom coordinates from the current structure are written to the substructure file, and these coordinates will be used whenever you use this substructure file. If you want to define the substructure for use with more than one structure, leave this option unselected. In this case, zeroes are written to the substructure file for the coordinates. These zeroes are interpreted as relative coordinates: that is, the coordinates are taken from whichever structure the constraints in the substructure file are applied to.

To read an existing substructure file, click Read .sbc File, navigate to the desired file and select it, then click OK. If you want, you can then edit the substructure to create a new substructure, using the controls in the Substructure tab.

4.6 Setting Up, Running, And Monitoring Jobs

When you have finished with all the settings for the calculation, click the Settings button to open the Job Settings dialog box and choose options for how and where the job is run. This dialog box provides a standard interface to the Job Control facility, which manages all jobs submitted from Maestro. It has a standard set of controls, but is configured to include application-specific or task-specific controls when needed. For example, for distributed jobs, the Host menu is replaced by a table that allows maximum flexibility in selecting hosts and processors. Details of the Settings button and the Job toolbar are given in Section 1.7 on page 5. For more information on the Job Settings dialog box, see Section 2.2 of the *Job Control Guide*.

Distributed MacroModel calculations can be started from Maestro for the following job types:

- Multiple minimization of non-conformers
- Redundant conformer elimination
- Ligand torsional search
- Serial conformational searches using MCMM, LMOD, or MCMM/LMOD

You may want to set up a MacroModel calculation and save the generated input files without actually submitting the job. This may be the case, for example, if you want to use MacroModel from the command line. To create the structure file and the command (`.com`) file needed to launch the job at a later time, specify the desired values using the settings on the desired MacroModel panel, and then choose Write from the Settings button menu.

After you click the Run button in the Job Settings dialog box, or the Run button on the panel for the calculation, the Monitor panel is displayed if you have the appropriate preference set, and the contents of the *jobname*`.log` file are displayed as it is written. While jobs are running, the Detach, Pause, Resume, Stop, Kill, and Update buttons are active. When there are no jobs currently running, only the Monitor and Delete buttons are active. You can choose to monitor or delete the currently selected job by clicking the corresponding buttons. To monitor or delete another job in the database, first choose the job in the list and then click the appropriate button. You can also monitor jobs from other projects. For more information on the Monitor panel, see Section 3.2 of the *Job Control Guide*.

Current Energy Calculations

Determining the current energy for a structure is a basic type of calculation in molecular modeling. Setting up a current energy calculation in MacroModel involves many of the same steps as other calculations within MacroModel, so be sure to read this chapter carefully.

In addition to calculating the total energy for the structure, MacroModel can be instructed to produce an `.mmo` file during a current energy calculation. An `.mmo` file contains detailed information on the various contributions to the potential energy and can lead to a number of insights. Maestro's Force Field Viewer is an excellent tool for interactively examining this information.

5.1 The Current Energy Panel

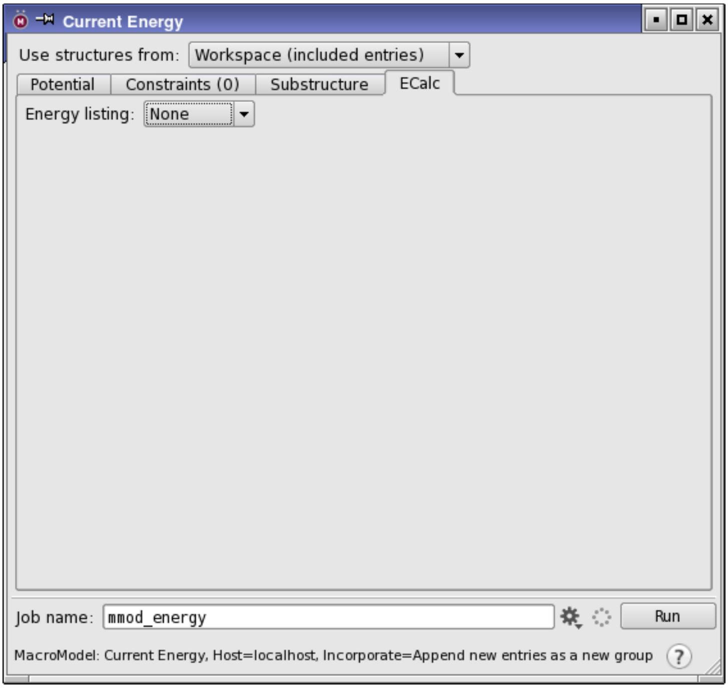
You can use the Maestro Current Energy panel to prepare and submit MacroModel Current Energy calculations. The Current Energy panel contains 4 tabs, each of which contains tools and settings related to a different aspect of a Current Energy calculation—Potential, Constraints, Substructure, and ECalc. The first three tabs were described in Chapter 4, so only the ECalc tab is described in this chapter.

To open the Current Energy panel, choose Current Energy from the MacroModel submenu of the Applications menu in the main menu bar.

You can calculate the current energy for multiple structures in a single job, by selecting multiple structures in the Project Table or including multiple structures in the Workspace, and choosing the appropriate option from the Use structures from option menu.

5.2 The ECalc Tab

The Energy Listing option menu allows you to select the amount of information in the energy listing for the current job. Selecting the None setting results in the total molecular mechanics energy being listed in the *jobname*`.log` file. Selecting Minimal prints in addition a minimal summary in the Energy Summary file (*jobname*`.mmo`). Selecting Complete lists the complete energy output to the Energy Summary file, including all internal coordinate components. A "complete" energy listing is required to use Maestro's force field viewing tool. You should note, however, that performing calculations with a complete energy listing can create large output files.



Use structures from: Workspace (included entries)

Energy listing: None

Job name: mmod_energy

MacroModel: Current Energy, Host=localhost, Incorporate=Append new entries as a new group

Figure 5.1. The ECalc tab of the Current Energy panel.

5.3 Command File Examples

To run MacroModel calculations, a molecular structure file and a command file are required. The molecular structure file contains the structures to be used as input in the calculation. The command file contains the name of the input structure file, the name of the output structure file, and an ordered list of operation codes (*opcodes)* for the calculations.

When you start a job from Maestro or write out the input files, Maestro writes out a molecular structure file and a command file. For many types of jobs, command files written this way are complete and adequate, but for some types of jobs you may need to adjust the Maestro-generated command file.

The command files and the log files for the examples given in this section can be found in `$SCHRODINGER/macromodel-v`*version*`/samples/Examples`.

Below is an example of a command file for an energy calculation with solvation. A description of the opcodes and their arguments follows.

```
ecalc.mae
ecalc-out.mae
 MMOD       0      1      0      0     0.0000     0.0000     0.0000    0.0000
 FFLD      10      1      0      1     1.0000     0.0000     0.0000    0.0000
 EXNB       0      0      0      0     0.0000     0.0000     0.0000    0.0000
 BDCO       0      0      0      0    41.5692 99999.0000     0.0000    0.0000
 SOLV       3      1      0      0     0.0000     0.0000     0.0000    0.0000
 READ       0      0      0      0     0.0000     0.0000     0.0000    0.0000
 ELST       1      0      0      0     0.0000     0.0000     0.0000    0.0000
```

`MMOD`: Creates and updates an intermediate structure file so that structures can be displayed in Maestro as the job progresses.

`FFLD`: Force field selection. Arg1 denotes the actual force field used in the calculation (in this case MMFF94). Arg2 defines the electrostatic treatment for the calculation. Default (arg=0) is to use the dielectric treatment encoded in the force field, however, in this case a constant dielectric is used due to the use of solvation model 3 (see `SOLV` below). Arg4 is MMFF94 specific. Arg4=1 defines the MMFF94s version of the force field, which ensures planarity around delocalized sp2 nitrogens.

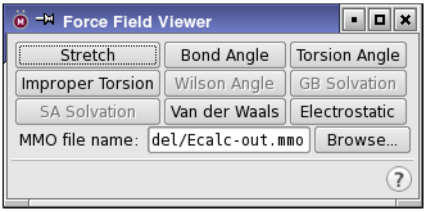
`EXNB`: Extended non-bonded interaction cut-off. This is set by default when solvation is used in a calculation. The default values for extended cut-off are: 8 Å vdW (arg5); 20 Å charge-charge electrostatic (arg6); 4 Å hydrogen bonding (arg7).

`BDCO`: Use the Bond Dipole CutOff (BDCO) method for truncating electrostatic interactions. Arg5 and arg6 are used to specify the cutoffs used for charge-dipole and charge-charge interactions, respectively.

`SOLV`: Specify the implicit solvation treatment to be used in the calculation. Arg1 defines the type of solvation model to be used. Arg1=3 means that the GB/SA solvation model will be used. Arg2 defines what type of solvent is used. Arg2=1 selects water as the solvent.

`READ`: Directs MacroModel to read the input file.

`ELST`: Calculates the single point energy of the input structure(s). Arg1 determines the extent of output listing, and to which files the output will be written. Arg1=1 gives a listing of the total molecular mechanics energy to the log file and a complete listing with all internal coordinates to the `.mmo` file. Arg2 allows for switching between kJ/mol and kcal/mol units for the energies listed.



MMO file name: del/Ecalc-out.mmo

Figure 5.2. The Force Field Viewer panel.

5.4 The Force Field Viewer

If you want to view details of the force field used in a MacroModel calculation, you can use the Force Field Viewer panel in Maestro. You first ensure that Complete is selected from the Energy Listing option menu before you run the job. The data displayed by the Force Field Viewer is stored in `.mmo` files, which are created when energy operations are performed using Complete energy listings. To read an `.mmo` file, click Browse and navigate to the desired file.

The Force Field Viewer panel has a number of buttons that represent the various types of interactions that can be present in a calculation (stretch, bond angle etc.). After an `.mmo` file has been read, buttons that correspond to types of interactions present in that particular file become active. Each button, when selected, displays a corresponding panel. In each panel, the interactions are listed, and selection of an interaction in the list marks the interaction in the displayed structure with a magnifying glass icon and a line or an asterisk. By default the first item in the list is selected. These panels are described in detail in the online help. The structure is also read and displayed as a scratch entry in the Workspace. When you have finished viewing force fields, click the Clear toolbar button to remove this scratch entry.

The Wilson Angle button and the Improper Torsion buttons are never active simultaneously, since they perform similar operations, but are associated with different force fields. Wilson angles are used in MMFF and MMFFs force field calculations, and improper torsions are used in the others.

A brief summary of each button and its corresponding panel is given below.

Stretch

The Stretch panel is used to display the stretching interactions for a given calculation. The selected interaction is marked with a dashed yellow line.

Bond Angle

The Bend panel, opened by clicking Bond Angle, is used to display the bending interactions for a given calculation. The selected interaction is marked with a dashed red line.

Torsion Angle

The Torsion Angle panel displays the torsion interactions for a calculation. The selected interaction is marked with a pale green dashed line.

Improper Torsion

In the Improper Torsion panel, the improper torsion interactions associated with a particular calculation can be viewed. The selected improper torsion in the displayed structure is marked with a gold-colored asterisk.

Wilson Angle

The Wilson Angle panel displays the Wilson angle interactions for a given calculation. The selected Wilson angle is marked with a peach-colored asterisk.

GB Solvation

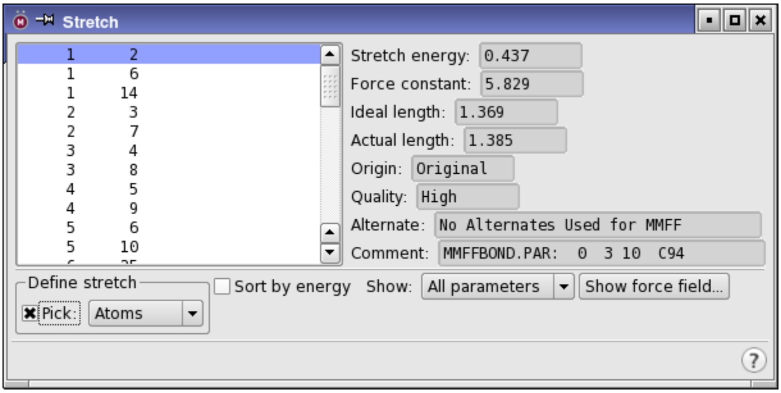
The GB Solvation panel can be used to view the Generalized Born (GB) portion of the solvation interactions for a given calculation. The selected atom in the displayed structure is marked with a yellow asterisk.

SA Solvation

The SA Solvation panel can be used to view the Surface Area (SA) solvation interaction components for a given calculation. The selected entry in the SA Solvation List is marked with a peach-colored line.

Van der Waals

The Van der Waals panel can be used to view all van der Waals interactions for a given calculation. The selected interaction in the displayed structure is marked with a dashed orange line.



1	2
1	6
1	14
2	3
2	7
3	4
3	8
4	5
4	9
5	6
5	10

Stretch energy: 0.437
Force constant: 5.829
Ideal length: 1.369
Actual length: 1.385
Origin: Original
Quality: High
Alternate: No Alternates Used for MMFF
Comment: MMFFBOND.PAR: 0 3 10 C94

☒ Pick: Atoms
☐ Sort by energy
Show: All parameters

Figure 5.3. The Stretch panel.

Electrostatic

The Electrostatic panel can be used to view all electrostatic interactions for a given calculation. The selected interaction is marked with a dashed purple line.

5.5 Checking and Interpreting Results

The energy estimate for an energy calculation depends critically on the nature of the structure it is applied to. It is often crucial that the particular structure was produced using a calculation, such as a minimization or a dynamics simulation, employing the same force field. Otherwise, the energy may contain significant contributions due to strain resulting from differences in the equilibrium structures for the different force fields.

You should also bear in mind the points raised in Section 2.2 on page 11.

Detailed energy listings can be useful, particularly in conjunction with Maestro's Force Field Viewer tool, for the following reasons:

- To gain insight into what interactions are key within the system
- To understand how important parameters of limited quality are for a particular calculation
- To examine problematic structures

Minimizations

Energy minimization is a key type of calculation in molecular modeling, in part because it gives a distinct structure that is often related to a subset of conformers found under thermal conditions. It also plays an important role in many compound calculations, such as conformational searching, and thus this chapter is a useful one to review carefully. MacroModel provides a number of well-tested and efficient minimization methods.

Minimization can be performed on a single structure or on a collection of structures. The collection of structures can be different molecules or they can be conformations of the same molecule. If the structures are conformers, then it is possible to do the minimization in conjunction with the elimination of redundant conformers.

As well as minimizing multiple structures, you can make use of multiple solvents and estimate the partition coefficient (logP) of the distribution of the minimized structures between the solvents.

A utility, `premin`, is available for preminimization of ligands for use with Glide and other applications. This utility provides particular choices of minimization parameters.

6.1 The Minimization Panel

The MacroModel Minimization panel is used to set up and submit minimization calculations from Maestro. The Minimization panel contains five tabs, the first three of which also appear in the other MacroModel panels. The controls in the first three tabs and the structure input controls are discussed in detail in Section 4.1 on page 31 through Section 4.5 on page 40. The remaining two tabs are described in the sections below.

To open the Minimization panel, choose Applications → MacroModel → Minimization or Tasks → Minimization → Force Field in the main window.

To set up a minimization calculation:

1. Select the entry or entries that you want to use as input, or display the structures you want to use in the Workspace.
2. Open the Minimization panel.
3. Set the controls in the first three tabs—Potential, Constraints, and Substructure.

4. In the Mini tab, select a minimization method, the maximum number of iterations, and the convergence criterion and threshold.
5. In the Comparison tab, select the type of structure (conformers or non-conformers) and the method, and make any related settings.

6.2 Setting Minimization Parameters

The features of the Mini tab for setting the minimization parameters are described in detail in the next few subsections.

6.2.1 Minimization Methods

From the Method option menu, select a minimization method for your calculation. Choose from the following supported methods:

- PRCG (Polak-Ribiere Conjugate Gradient)

  This is a conjugate gradient minimization scheme that uses the Polak-Ribiere first derivative method with restarts every 3N iterations [17]. This is the best general method for energy minimization, but it should not be used to find transition states.

- TNCG (Truncated Newton Conjugate Gradient)

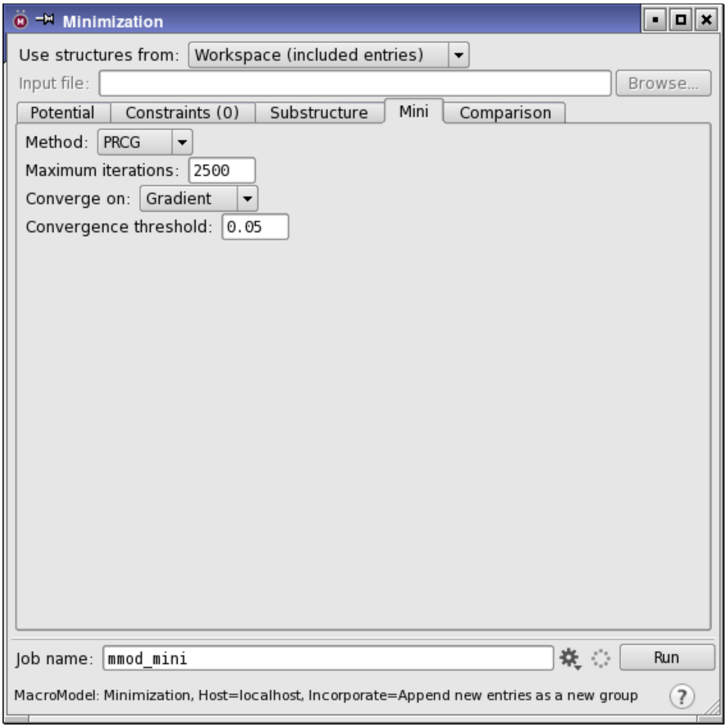
  TNCG uses second derivatives and line searching and is highly efficient for producing very low gradient structures. It generally converges in one tenth the number of iterations necessary for a PRCG, but each iteration takes more time. Often FMNR re-minimization of TNCG structures gives the lowest final gradients [18].

- OSVM (Oren-Spedicato Variable Metric)

  This is a variable metric minimization that uses the Oren-Spedicato modification [19] of the Fletcher-Powell quasi-Newton method. Convergence to saddle points is possible. Typical convergence occurs in 3N – 6N iterations, but note that iteration speed is relatively slow. OSVM is not recommended for structures with poor starting geometries.

- SD (Steepest Descent)

  This is a steepest descent minimization method. The “SD” should not be used to find saddle points, and convergence is poor towards the end of minimization. This is a good method for starting geometries that are far from the minimum, but switching to another method is recommended when derivatives fall below 10 kJ/mol-Å or so. It is generally not optimal to use SD minimization. PRCG is usually a better choice.



Use structures from: Workspace (included entries)

Input file:

Method: PRCG

Maximum iterations: 2500

Converge on: Gradient

Convergence threshold: 0.05

Job name: mmod_mini

MacroModel: Minimization, Host=localhost, Incorporate=Append new entries as a new group

Figure 6.1. The Mini tab of the Minimization panel.

- FMNR (Full Matrix Newton Raphson)

  With this method, convergence to saddle points is not uncommon. Use FMNR with preminimized structures having RMS gradients of less than 0.1 kJ/mol-Å. Use line searching (select after prompt) for problematic cases, or if the RMS First Derivative is greater than 0.1. The preminimization requirement derives from the Newton Raphson assumption of a quadratic potential surface. The method works only if the assumption is valid. FMNR is the most effective method for fully converging structures, but the computational resources required are significant for large structures.

  To find saddle point structures, you will need to start very close to the saddle point, and you will have to disable line searching.

  FMNR has excellent convergence properties, and typically converges in two to ten iterations.

- LBFGS (Low-memory Broyden-Fletcher-Goldfarb-Shanno)

  A method that performs well with large structures.

- Optimal

  This choice uses the TNCG method if the number of unfixed atoms is less than 1000 and solvation is employed, otherwise it uses the PRCG method. (Sets `MINI` arg1 to 20).

6.2.2 Convergence Parameters

You can specify convergence criteria for your calculation using the **Converge on** option menu in the **Mini** tab. Choose from **Gradient**, **Energy**, **Movement**, or **Nothing**.

Gradient is the default value. With the **Energy** setting, Macromodel determines when to halt the calculation based on the energy difference between iterations. With the **Movement** setting, MacroModel determines when to stop the calculation based on the maximum atomic movement at each iteration. When the **Nothing** option is selected, the calculation runs for the maximum number of iterations as specified in the **Max # Iterations** text field.

The value in the **Maximum iterations** text box determines when MacroModel should end the calculation if the specified convergence criterion hasn't been met. The default value is `50` for SD and FMNR minimization methods, and `500` for the remaining methods. To specify a different number of iterations, enter it into the **Maximum iterations** text box.

The value entered in the **Convergence threshold** text box determines the threshold applied to the **Converge on** method. The default convergence setting is convergence on **Gradient** with a threshold of `0.05`. These settings will suffice in most cases.

6.2.3 Using Several Different Minimization Stages During a Minimization

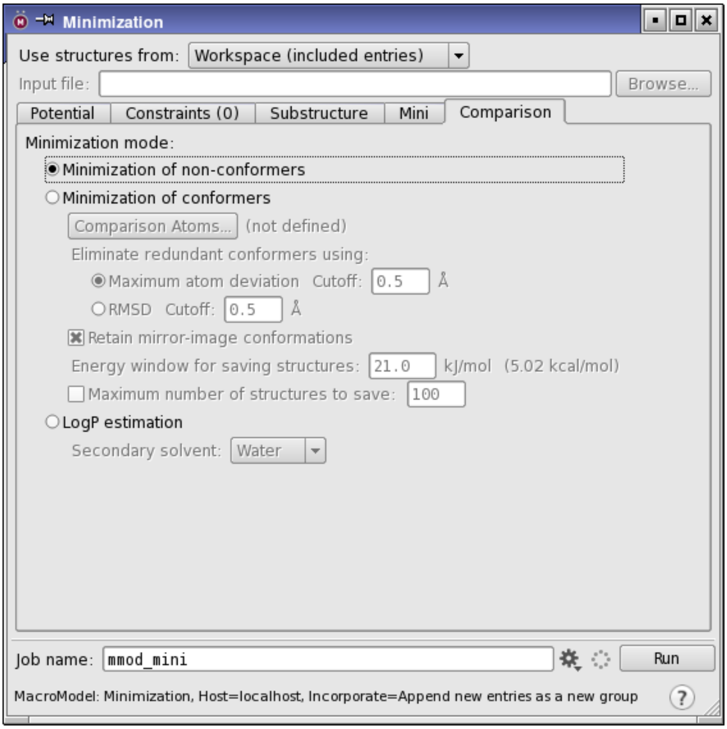
The MWRT feature of MacroModel enables different minimizer methods to be automatically applied during the minimization stage of a MacroModel computation. Each phase of an MWRT minimization can use a different minimizer method. For instance, an initial steepest descent minimization could be followed by an all-purpose method, with the process concluded by a minimizer method expected to result in final convergence, like FMNR. The minimization sequence ceases when the convergence criteria (as defined by the CONV opcode) is reached. MWRT can be applied to minimizations and conformational searches, including multi-structure computations. This feature is supported with a particular set of choices as the **Optimal** minimization method in Maestro. To make your own choices, use the `MWRT`: opcode in the command file. Section 6.7.2 on page 61 contains examples of MWRT minimizations, and section 9.3.4 contains examples of MWRT conformational searches.

6.3 Setting Up Minimization of Conformers with Redundant Conformer Elimination

By default, all structures are treated as separate molecules. However, if your structures are conformers, you can perform the minimization and eliminate any conformers that are found to be redundant after the minimization, and you can also filter out conformers based on energetic criteria. To detect redundancies, you have to choose which atoms in the structure are to be used to compare the structures, and how close they have to be when the structures are superimposed for the structures to be considered redundant. These choices can be made in the Comparison tab. To indicate that your structures are conformers, select Minimization of conformers, then make settings as described below.

Comparison Atoms

During multiple minimizations of conformers, redundant structures are eliminated if comparison atoms are specified. The structures are compared against other low-energy structures that have already been minimized. The comparison is performed by rigid superposition, comparing



Use structures from: Workspace (included entries)

Input file:

Minimization mode:

● Minimization of non-conformers

○ Minimization of conformers

Comparison Atoms... (not defined)

Eliminate redundant conformers using:

● Maximum atom deviation Cutoff: 0.5 Å

○ RMSD Cutoff: 0.5 Å

☒ Retain mirror-image conformations

Energy window for saving structures: 21.0 kJ/mol (5.02 kcal/mol)

☐ Maximum number of structures to save: 100

○ LogP estimation

Secondary solvent: Water

Job name: mmod_mini

MacroModel: Minimization, Host=localhost, Incorporate=Append new entries as a new group

Figure 6.2. The Comparison tab of the Minimization panel.

only those atoms specified as "comparison atoms" in the setup, taking into account the topological symmetry of the molecule.

To define comparison atoms, click the Comparison Atoms button to open the Comparison Atoms panel, and use one or more of the following options:

- Click Heavy Atoms + O-H,S-H

  This adds all the non-hydrogen atoms and the hydrogen atoms attached to oxygen and sulfur to the list of comparison atoms.

- Click Heavy Atom

  This adds only the non-hydrogen atoms to the list of comparison atoms.

- Pick atoms in the Workspace

  Choose a structural unit from the Pick menu, and pick atoms in the Workspace to add the atoms in the structural unit to the list of comparison atoms.

- Select predefined atom sets

  Click the Atom Selections button and choose an item from the list to add the atoms in that category to the list of comparison atoms.

- Select atoms using the Atom Selection dialog box

  For more complex combinations of comparison atoms, you can select atoms using the Atom Selection dialog box to build an Atom Specification Language (ASL) expression for the required atoms. See Section 6.5 of the *Maestro User Manual* for more information on the Atom Selection dialog box. Click the Atom Selections button and choose Select to open the Atom Selection dialog box.

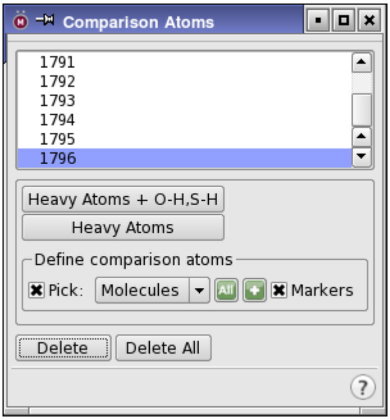


Figure 6.3. The Comparison Atoms panel.

When comparison atoms are picked, Maestro places light green “=” labels on them. To distinguish the currently selected atom, its label is colored turquoise. To hide these markers, clear the Markers option.

To delete a defined comparison atom, select it in the list of comparison atoms, then click Delete. To redefine a comparison atom, pick a new atom while the comparison atom is selected in the list. To delete all the defined comparison atoms, click Delete All.

Eliminate redundant conformers using

There are two options for the distance criterion for eliminating redundant conformers: Maximum atom deviation, and RMSD. The threshold for determining whether structures should be considered to be equivalent is set for both by default to 0.5 Å. You can change the cutoff in the text boxes. When the structures are compared, the maximum distance or the RMSD between pairs of corresponding atoms for all such pairs must be less than this threshold for the structures to be considered equivalent. These options are only available when comparison atoms have been defined.

Retain mirror-image conformations

If this option is selected, mirror-image conformations are treated as separate conformers, and are not eliminated as redundant. This option is only available when comparison atoms have been defined.

Energy window for saving structures

If a structure is minimized and found to be above the global minimum by more than the energy window value (in kJ/mol), it is rejected and not included in the output file.

Maximum number of structures to save

You can limit the number of structures returned in the output structure file by selecting this option and entering a value in this text box.

6.4 Partition Coefficient Estimation

Minimization calculations can also be used to obtain estimates of the partition coefficients of a set of solutes between two solvents. After you have set up other parameters for a multiple minimization, including a primary solvent, select LogP estimation in the Comparison tab and choose the secondary solvent from the Secondary solvent menu.

Each molecule in the input file is minimized twice, once in each solvent. The results of each minimization are used to evaluate the free energy difference for the two solvents. Hence the logarithm of the partition coefficient, log $P_{\text{solv1,solv2}}$, which is defined by the relationship

$$\log P_{\mathrm{solv1,solv2}} = (\Delta G_{\mathrm{solv2}} - \Delta G_{\mathrm{solv1}}) / (2.30RT)$$

where $P_{\mathrm{solv1,solv2}}$ = [Solute]$_{\mathrm{solv1}}$ / [Solute]$_{\mathrm{solv2}}$, $\Delta G_{\mathrm{solv}N}$ is the free energy of solvation of the molecule in solvent N, R is the gas constant, and T is the temperature in kelvin. The results are collected in a table at the end of the log file and are added as properties to the output structure file.

The solvation models are parametrized for ambient conditions. If the temperature deviates significantly from these conditions the solvation energy estimates and hence the calculated log $P_{\mathrm{solv1,solv2}}$ become less reliable.

6.5 Automatic Setup (AUTO)

You can use automatic setup for minimizations of a collection of conformers with redundant conformer elimination. Automatic setup is enabled by adding the `AUTO` opcode to the command file for a calculation. You should place the `AUTO` opcode, with appropriate arguments, just before the `MINI` opcode in the command file. You must also add a `MULT` opcode to the command file, as seen in the example in Section 6.7.4 on page 63. `AUTO` is available from Maestro only for conformational searches. Automatic setup should not be used with minimizations of single structures or minimizations with multiple non-conformers as input.

`AUTO` selects comparison atoms (`COMP`), so that redundant conformers can be identified and removed without explicitly designating comparison atoms in the command file or from Maestro. In addition, chiral atoms (`CHIG`) and torsion checks (`TORC`) are automatically identified by `AUTO` . `AUTO` applies the desired structural criteria to the minimized conformers as if the `CHIG` and `TORC` opcodes were present in the command file. `AUTO` has options to adjust or turn off these features. See the `AUTO` opcode in the *MacroModel Reference Manual* for more information.

In addition, `AUTO` uses the information in any substructure specified for the computation, including information on fixed or frozen atoms. The substructure must be specified in a substructure file and read in with a `SUBS 0` command, and cannot be specified in the command file. Likewise, fixed or frozen atoms must be specified in the substructure and cannot be specified independently.

Finally, if only automatic comparison atom setup is required for a multi-conformer minimization with redundant conformer elimination, then `COMP` arg1=0 may also be used in the command file. Examples are given in Section 6.7.4 on page 63.

`AUTO` is not currently compatible with `LOOP`, and must be used carefully with Embrace. For more information on the parameters that `AUTO` is assembling, use `DEBG 520` or `521`.

6.6 Preparation of Multi-Ligand Structure Files With premin

MacroModel's minimization capabilities are used in a script named `premin`, which is designed to prepare multi-ligand structure files for use in Glide and other applications. Database files sometimes contain problematic ligand structures that are either chemically incorrect or have species that are not covered by the force field parameters. This script culls out such problematic structures and minimizes the energy of all other structures. The problematic structures and the minimized structures are saved in separate files. You can also filter out problematic structures without doing a preminimization.

The preminimization employs a 4*r* distance-dependent dielectric model and uses MacroModel's efficient truncated-Newton minimizer. As well as the recommended default force field, MMFFs, `premin` also supports the OPLS_2001 and OPLS_2005 force fields.

The syntax for `premin` is shown below:

```
$SCHRODINGER/utilities/premin [options] input.mae
```

where *input*`.mae` is the input file that contains the ligand structures. The options are described in Table 6.1.

Table 6.1. Options for the premin script.

Option	Description
`-h`	Print help text and exit.
`-lic ligprep`	Use LigPrep license instead of MacroModel license
`-s` *output*	Output file for successfully processed structures. Default is *input*`-min.mae` for minimizations, and *input*`-filtered.mae` for filtering.
`-u` *bad*	Output file of structures that could not be minimized. Default is *input*`-bad.mae`.
`-v`	Print version number and exit.
`-doc`	Print a brief guide to using `premin`.
`-f 11`	Use OPLS_2001 instead of the default force field, MMFFs.
`-f 14`	Use OPLS_2005 instead of the default force field, MMFFs.
`-custom`	Use custom `filter.com` and `goodmin.com` files from the current working directory.
`-filter|` `-m filter`	Filtering only: skip minimization.

By default, successfully minimized structures are saved in *input*`-min.mae`, and unsuccessful structures are saved in *input*`-bad.mae`. If you choose to filter out problematic structures without minimization (`-filter` or `-m filter`), successfully filtered structures are saved in the file *input*`-filtered.mae`, and problematic structures are again saved in *input*`-bad.mae`.

The script creates two MacroModel command files, `filter.com` and `goodmin.com`, and writes them to the current working directory. You can then customize these files and use them in place of the default files with the `-custom` option. If you perform filtering only, the file `goodmin.com` is not used.

This script uses the `SPAT` opcode to filter out problematic structures that contain metal ions or generalized atom types.

6.7 Command File Examples

To run MacroModel calculations, a molecular structure file and a command file are required. The molecular structure file contains the structures to be used as input in the calculation. The command file contains the name of the input structure file, the name of the output structure file, and an ordered list of operation codes (*opcodes)* for the calculations.

When you start a job from Maestro or write out the input files, Maestro writes out a molecular structure file and a command file. For many types of jobs, command files written this way are complete and adequate, but for some types of jobs you may need to edit the command file.

The command files and the log files for the examples given in this section can be found in `$SCHRODINGER/macromodel-v`*version*`/samples/Examples`.

6.7.1 Constrained Minimization

Below is an example command file for a minimization calculation. Explanations of the opcodes that appear in the file follow.

```
mini-constr.mae
mini-constr-out.mae
 MMOD       0      1      0      0      0.0000      0.0000     0.0000     0.0000
 FFLD      10      1      0      1      1.0000      0.0000     0.0000     0.0000
 BDCO       0      0      0      0     41.5692  99999.0000     0.0000     0.0000
 READ       0      0      0      0      0.0000      0.0000     0.0000     0.0000
 FXTA      14     15     16     17    100.0000    -44.7270     5.0000     0.0000
 CONV       2      0      0      0      0.0500      0.0000     0.0000     0.0000
 MINI       9      0    500      0      0.0000      0.0000     0.0000     0.0000
```

`MMOD`: Creates and updates an intermediate structure file so that structures can be displayed in Maestro as the job progresses.

`FFLD`: Force field selection. Arg1 denotes the actual force field used in the calculation (in this case MMFF94). Arg2 defines the electrostatic treatment for the calculation. Default (arg=0) is to use the dielectric treatment encoded in the force field. However, in this case a constant dielectric is used due to the use of solvation model 3 (see `SOLV` below). Arg4 is MMFF94-specific. Arg4=1 defines the MMFF94s version of the force field, ensuring planarity around delocalized sp2 nitrogens.

`BDCO`: Use the Bond Dipole CutOff (`BDCO`) method for truncating electrostatic interactions. Arg5 and arg6 are used to specify the cutoffs used for charge-dipole and charge-charge interactions, respectively.

`READ`: Directs MacroModel to read the input file.

`FXTA`: Constrains a torsional angle. Arg1–arg4 give the atom numbers of the four atoms defining the dihedral angle that will be constrained. The angle is kept in place through a force constant, defined in arg5 (kJ/mol). Arg6 lists the desired value for the angle, and arg7 gives the half-width of the flat-bottom potential used (i.e., the “flexibility” of the constrained angle).

`CONV`: Defines convergence criteria. Arg1=2 signifies derivative convergence (if no `CONV` command is present, default criterion is 0.05 kJ/mol-Å; this value is set in arg5).

`MINI`: Starts the minimization. Arg1 defines the type of minimization algorithm to be used. Arg1=9 means that Truncated Newton-Raphson Conjugate Gradient will be used. In arg3, the number of minimization steps is defined. Arg3 can be set to a large number since the calculation stops automatically as soon as the convergence criterion has been reached.

6.7.2 Multiple-Method (MWRT) Minimizations

MWRT minimizations enable multiple minimizer methods to be applied to a structure during the minimization stage. This can enhance the performance of the minimization process.

The first example below demonstrates the efficient minimization of a small protein domain.

```
mwrt_mini.mae
mwrt_mini-out.mae
 FFLD      14      1      0      0     1.0000     0.0000     0.0000     0.0000
 SOLV       3      1
 EXNB       0      0      0      0     0.0000     0.0000     0.0000     0.0000
 BDCO       0      0      0      0    89.4427 99999.0000     0.0000     0.0000
 READ
 CONV       2      0      0      0     0.0020     0.0000     0.0000     0.0000
 MWRT       0
 MINI       0      1    100      0     0.0000     0.0000     0.0000    10.0000
 MINI       1      1    300      0     0.0000     0.0000     0.0000    10.0000
 MINI       9      1     50      0     0.0000     0.0000     0.0000    10.0000
 MWRT       1
 MINI       4      1     30      0     0.0000     0.0000     0.0000    10.0000
```

`MWRT`: arg1 value of zero stores the `MINI` minimization parameter data.

`MINI`: In order, the `MINI` commands request 100 SD iterations, 300 PRCG iterations, 50 TN iterations, and 30 FMNR iterations. The last `MINI` command is after the second `MWRT` command, and actually initiates the minimization.

`MWRT`: arg1 value of 1 instructs computaton to run sequential minimizations after the next `MINI` command.

The second example performs MWRT minimization on multiple drug candidate structures.

```
mwrt_multmin.mae
mwrt_multmin-out.mae
 FFLD      14      1      0      0     1.0000     0.0000     0.0000     0.0000
 SOLV       3      1
 EXNB       0      0      0      0     0.0000     0.0000     0.0000     0.0000
 BDCO       0      0      0      0    89.4427 99999.0000     0.0000     0.0000
 BGIN
 READ
 CONV       2      0      0      0     0.0020     0.0000     0.0000     0.0000
 MWRT       0
 MINI       0      1    100      0     0.0000     0.0000     0.0000    10.0000
 MINI       1      1    300      0     0.0000     0.0000     0.0000    10.0000
 MINI       9      1     50      0     0.0000     0.0000     0.0000    10.0000
 MWRT       1
 MINI       4      1     30      0     0.0000     0.0000     0.0000    10.0000
 END
```

`BGIN`/`END`: Defines the start/end of a loop over the input structures. All commands between the `BGIN` and `END` lines are performed for each structure in the input file.

6.7.3 Energy Minimization of Multiple Non-Conformers

Below is an example of the command file for an energy minimization of multiple non-conformers and explanations of the opcodes used in the file.

```
mult-min.mae
mult-min-out.mae
 MMOD    0    1    0    0     0.0000     0.0000     0.0000     0.0000
 FFLD   10    1    0    1     1.0000     0.0000     0.0000     0.0000
 BDCO    0    0    0    0    41.5692 99999.0000     0.0000     0.0000
 DEMX    0    0    0    0    50.0000     0.0000     0.0000     0.0000
 BGIN    0    0    0    0     0.0000     0.0000     0.0000     0.0000
 READ    0    0    0    0     0.0000     0.0000     0.0000     0.0000
 CONV    2    0    0    0     0.0500     0.0000     0.0000     0.0000
 MINI    9    0  500    0     0.0000     0.0000     0.0000     0.0000
 END     0    0    0    0     0.0000     0.0000     0.0000     0.0000
```

`MMOD`: Creates and updates an intermediate structure file so that structures can be displayed in Maestro as the job progresses.

`FFLD`: Force field selection. Arg1 denotes the actual force field used in the calculation (in this case, MMFF94). Arg2 defines the electrostatic treatment for the calculation. The default (arg=0) is to use the dielectric treatment encoded in the force field. Arg4 is MMFF94-specific. Arg4=1 defines the MMFF94s version of the force field, ensuring planarity around delocalized sp^2 nitrogens.

`BDCO`: Use the Bond Dipole CutOff (BDCO) method for truncating electrostatic interactions. Arg5 and arg6 specify the cutoffs used for charge-dipole and charge-charge interactions, respectively.

`DEMX`: This command prevents saving of high-energy conformers during the search. Arg5 defines the allowed energy window above the current global minimum. New conformers that are not within arg5 kJ/mol are discarded. Additionally, a preliminary energy test can be performed during the energy minimization to ensure that a reasonable structure has been found. Arg2 sets the number of energy iterations to perform before the preliminary test (a good value is approximately 1/3 of the total number of energy iterations), while arg6 defines the energy above which conformers are discarded (a value of about 1.5–2 times arg5 is recommended).

`BGIN`/`END`: Defines the start/end of a loop. All commands between the `BGIN` and `END` lines are performed for each structure in the input file.

`READ`: Directs MacroModel to read the input file.

`CONV`: Defines convergence criteria. Arg1=2 signifies derivative convergence. The default criterion, if no `CONV` command is present, is 0.05 kJ/mol-Å; this value is set in arg5.

`MINI`: Starts the minimization. Arg1 defines the type of minimization algorithm to be used. Arg1=9 means that Truncated Newton-Raphson Conjugate Gradient will be used. In arg3, the number of minimization steps is defined. Arg3 can be set to a large number since the calculation will automatically stop as soon as the convergence criterion is reached.

6.7.4 Multiple Conformer Minimization With Automatic Redundant Conformer Elimination

Below are two examples of command files for energy minimization of multiple conformers with redundant conformer elimination. The first uses the `AUTO` opcode; the second uses the `COMP` opcode with arg1=0. Explanations of the opcodes used in the file that are essential to this example are given.

```
mult_conformers.mae
mult_auto-out.mae
 MMOD       0      1      0      0     0.0000     0.0000     0.0000     0.0000
 FFLD      10      1      0      1     1.0000     0.0000     0.0000     0.0000
 SOLV       3      1      0      0     0.0000     0.0000     0.0000     0.0000
 EXNB       0      0      0      0     0.0000     0.0000     0.0000     0.0000
 BDCO       0      0      0      0    89.4427 99999.0000     0.0000     0.0000
 CRMS       0      0      0      0     0.0000     0.2400     0.0000     0.0000
 DEMX       0      0      0      0    51.0000     0.0000     0.0000     0.0000
 MULT       0      0      0      0     0.0000     0.0000     0.0000     0.0000
 BGIN       0      0      0      0     0.0000     0.0000     0.0000     0.0000
 READ       0      0      0      0     0.0000     0.0000     0.0000     0.0000
 CONV       2      0      0      0     0.0100     0.0000     0.0000     0.0000
 MSYM       0      0      0      0     0.0000     0.0000     0.0000     0.0000
 AUTO       1      0      0      0    -1.0000     0.0000     0.0000     0.0000
 MINI       1      0   5000      0     0.0000     0.0000     0.0000     0.0000
 END        0      0      0      0     0.0000     0.0000     0.0000     0.0000
```

`MULT`: Mandatory for this computation, due to the presence of the `AUTO` opcode.

`AUTO`: Automatic parameter assignment. Arg1 turns on `AUTO` only for the first conformer. Arg5 turns off torsional parameters, which are not needed for this minimization.

```
mult_conformers.mae
mult_comp-out.mae
 MMOD       0      1      0      0     0.0000     0.0000     0.0000     0.0000
 FFLD      10      1      0      1     1.0000     0.0000     0.0000     0.0000
 SOLV       3      1      0      0     0.0000     0.0000     0.0000     0.0000
 EXNB       0      0      0      0     0.0000     0.0000     0.0000     0.0000
 BDCO       0      0      0      0    89.4427 99999.0000     0.0000     0.0000
 CRMS       0      0      0      0     0.0000     0.2400     0.0000     0.0000
 DEMX       0      0      0      0    51.0000     0.0000     0.0000     0.0000
 BGIN       0      0      0      0     0.0000     0.0000     0.0000     0.0000
 READ       0      0      0      0     0.0000     0.0000     0.0000     0.0000
 CONV       2      0      0      0     0.0100     0.0000     0.0000     0.0000
 MSYM       0      0      0      0     0.0000     0.0000     0.0000     0.0000
 COMP       0      0      0      0     0.0000     0.0000     0.0000     0.0000
 MINI       1      0   5000      0     0.0000     0.0000     0.0000     0.0000
 END        0      0      0      0     0.0000     0.0000     0.0000     0.0000
```

`COMP`: Arg1=0 specifies that all heavy atoms are to be used as the comparison atoms when redundant conformer elimination is performed.

6.7.5 Multiple Minimization with Substructures Defined by ASL

The following example illustrates the use of ASL in a multiple minimization calculation that constrains heavy atoms of the molecules. The file `asl_test.mae` file consists of two molecules, cyclopentane and cyclohexane.

```
asl_test.mae
asl_test-out.mae
 FFLD      14      1      0      0     1.0000     0.0000     0.0000     0.0000
 BDCO       0      0      0      0    41.5692 99999.0000     0.0000     0.0000
 BGIN       0      0      0      0     0.0000     0.0000     0.0000     0.0000
 SUBS       0      0      0      0     0.0000     0.0000     0.0000     0.0000
 READ      -1      0      0      0     0.0000     0.0000     0.0000     0.0000
 CRMS       0      0      0      0     0.0000     0.2500     0.0000     0.0000
 CONV       2      0      0      0     0.0500     0.0000     0.0000     0.0000
 MINI       1      0    500      0     0.0000     0.0000     0.0000     0.0000
 END        0      0      0      0     0.0000     0.0000     0.0000     0.0000
```

`READ`: Arg1 is set to –1 to clear all molecular information for the previously read molecules.

The substructure file used for this example is as follows:

```
 ASL1      -1        all
 FXAT       0      0      0      0     0.0000     0.0000     0.0000     0.0000
 FXAT       0      0      0      0    -1.0000     0.0000     0.0000     0.0000
 ASL1       0       atom.ele H
 ASL2    200.0   not (atom.ele H)
```

The first three lines clear all free, fixed atom and frozen atom definitions respectively. The fourth line adds all hydrogens to the substructure. The fifth line constrains all heavy atoms to their starting positions with a constraining force constant of 200 kJ mol^{-1} Å^{-2}.

6.7.6 Partition Coefficient Estimation

Below is an example of a command file for conducting log $P_{\text{octanol,water}}$ calculations on a series of molecules. It is very similar to that used for multiple minimization of non-conformers.

```
logP.mae
logP-out.mae
 FFLD      11      1      0      1     1.0000     0.0000     0.0000     0.0000
 BDCO       0      0      0      0    41.5692 99999.0000     0.0000     0.0000
 SOLV       3      9      0      0     0.0000     0.0000     0.0000     0.0000
 LOGP       1      1      1      0     0.0000     0.0000     0.0000     0.0000
 BGIN       0      0      0      0     0.0000     0.0000     0.0000     0.0000
 READ       0      0      0      0     0.0000     0.0000     0.0000     0.0000
 AUTO       0     -1      0      0     0.0000     1.0000     0.0000     0.0000
 CONV       2      0      0      0     0.0500     0.0000     0.0000     0.0000
 MINI       1      0   5000      0     0.0000     0.0000     0.0000     0.0000
 END        0      0      0      0     0.0000     0.0000     0.0000     0.0000
```

`SOLV`: Specifies the first, or primary, solvent. Arg2 = 9 corresponds to octanol.

`LOGP`: Turns on partition coefficient estimation and specifies the second solvent. Arg2 = 1 corresponds to water.

`AUTO`: Necessary for the proper processing of `LOGP` calculations. Arg2 = −1 instructs `AUTO` not to make a list of comparison atoms, and arg6 = 1.0000 indicates that this is a serial calculation.

In the `.log` file produced when such a calculation is run, a report appears for each ligand processed:

```
 =====================================
 LOGP(octanol,water) Calculation for ligand #  1
 Ligand Name:  Case74C
 LOGP(octanol,water) =   1.41024
 =====================================
```

and a table appears at the end of the `.log` file:

```
=====================================
  LOGP(octanol,water) Results

 Ligand     Solvation Free Energies    LOGP(octanol,water)   Converged
             octanol         water
             kJ/mol.      kJ/mol.
     1        -26.99        -18.94         1.41             T
     2        -20.93        -12.48         1.48             T
     3        -20.02        -11.02         1.58             T
     4        -29.41        -16.19         2.32             T
...
    12        -16.84        -15.40         0.25             T
 =====================================
```

Here, `T` or `F` in the `Converged` column signifies whether or not the minimizations (one in each solvent for each molecule) converged. If they did not converge, the results should not be considered reliable.

6.8 Checking and Interpreting Results

Any individual minimization is intended to minimize the structure in the local minimum which may not be the global minimum of the system. This fact can make it hard to put the minimized system in the proper context without additional calculations (e.g., conformational searches). Typical output for a MacroModel minimization looks like this:

```
 Starting conjugate gradient minimization.
 Minimization converged; gradient = 0.387E-01 .LT.  0.500E-01
 Iterations = 110  out of  500
 Conf    1 E =  -594.116 (  0.039) kJ/mol
 BatchMin normal termination
 Total number of structures processed = 1
BatchMin: normal termination                           20-Feb-2002
15:31:29
```

The line:

```
 Minimization converged; gradient = 0.387E-01 .LT.  0.500E-01
```

indicates that the minimization met the convergence conditions. In this case the gradient attained, 0.0387 kJ/mol-Å, was less than the level required, 0.05 kJ/mol-Å.

If the calculation had not converged, the minimization converged line would be missing and the number of iterations would match the total specified.

Unconverged results typically have little or no value. If the minimization does not converge within the specified number of iterations, you may need to repeat or continue the minimization with more iterations.

Most MacroModel calculations consider minimizations converged when the RMS gradient of the energy is less than 0.05 kJ/mol-Å. While this is adequate for most types of calculations, this value may be modified or another property, such as atomic motion, may be used to detect when a minimization is sufficiently converged.

The results of a minimization are force-field-dependent: minimizing the same initial structure using different force fields usually results in similar but not identical structures.

Coordinate Scans

Coordinate scans can be used to probe the energy of a molecule as a function of a small number of geometric parameters: distances, angles, or dihedral angles. The process involves stepping through the specified parameters, restraining the parameters and performing a minimization on the remaining geometric parameters. For example, dihedral angle scans (formerly known as dihedral drives) are used to map out the potential energy for one or two dihedral angles, such as a Ramachandran ϕ–ψ plot for amino acids, and can be very useful in understanding conformational behavior. Coordinate scans can be performed either from Maestro or from the command line. Section 4.7 of the *MacroModel Reference Manual* contains information on the commands required for a command-line coordinate scan.

7.1 Performing a Coordinate Scan in Maestro

You can use the MacroModel Coordinate Scan panel to set up and submit coordinate scan calculations on either one coordinate or two coordinates of the same type. The tools in the Scan tab allow you to define the coordinates for the calculation, to set the range through which the coordinates are to be scanned, and to specify the increment for each point in the scan.

All of the components of the Coordinate Scan panel, with the exception of the Scan tab, appear on other MacroModel energy panels. For information about the upper portion of the Coordinate Scan panel, the Potential, Constraints, and Substructure tabs, see Section 4.1 on page 31 through Section 4.5 on page 40. For information about the controls in the Mini tab, see Section 6.1 on page 51.

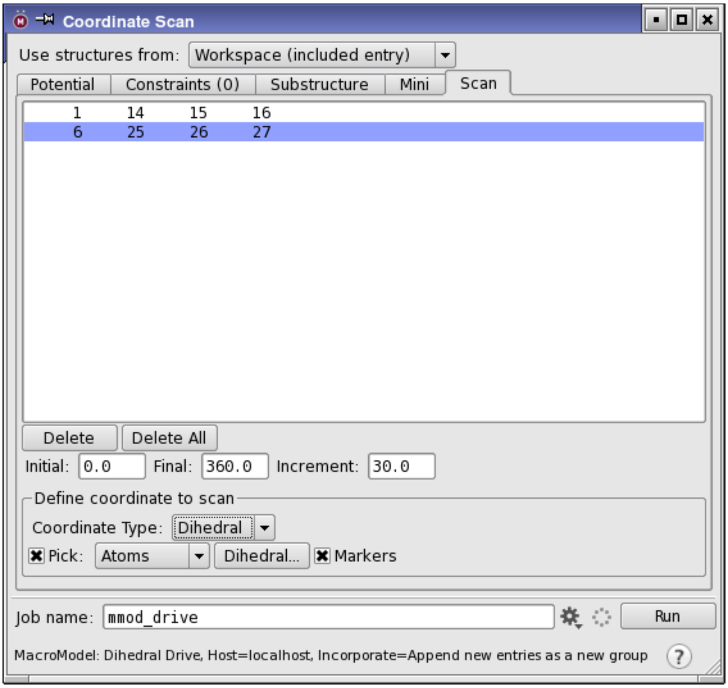
To open the Coordinate Scan panel, choose Coordinate Scan from the MacroModel submenu of the Applications menu on the main menu bar.

To set up a coordinate scan:

1. Choose a coordinate type from the Coordinate type option menu.

   You can choose Distance, Angle, or Dihedral. This choice fixes the coordinate type for the scan: if you scan two coordinates, they must be of the same type.

2. Choose Atoms or Bonds from the Pick menu.



Use structures from: Workspace (included entry)

1	14	15	16
6	25	26	27

Initial: 0.0 Final: 360.0 Increment: 30.0

Coordinate Type: Dihedral

☒ Pick: Atoms ☒ Markers

Job name: mmod_drive

MacroModel: Dihedral Drive, Host=localhost, Incorporate=Append new entries as a new group

Figure 7.1. The Scan tab of the Coordinate Scan panel.

3. Pick atoms or bonds in the Workspace to define the coordinate.

   You should pick two atoms or one bond for a distance, three atoms or two bonds for an angle, and four atoms or three bonds for a dihedral. The picked atoms are marked with purple cubes until the last pick is made. The cubes are then replaced with a line that marks the coordinate and a symbol for the scan type. Distances are marked in purple, angles in green, and dihedrals in aquamarine.

   You can also click the Dihedral button to select protein dihedrals by their standard names (psi, phi, omega, etc.).

4. Enter values for the range and step size in the Initial, Final, and Increment text boxes.

   An increment can be negative, but only if the initial value is larger than the final value. For distances, the default values are 1.0 and 10.0 angstroms for the initial and final values, and 1.0 angstroms for the increment. For angles and dihedrals, the default initial and final values are 0° and 360°, and the default increment is 30°. The minimum angle or dihedral increment is 1°, but an increment of 10° is usually sufficient.

5. (Optional) Repeat the picking operations to define a second coordinate.

To change the initial and final values and the increment for the scan calculation, select a coordinate entry from the list at the top of the tab, then enter values in the Initial, Final, and Increment text boxes.

To delete a defined coordinate, select it in the list of coordinates and click Delete. To delete all defined coordinates, click Delete All.

7.2 Command File Examples

To run MacroModel calculations, a molecular structure file and a command file are required. The molecular structure file contains the structures to be used as input in the calculation. The command file contains the name of the input structure file, the name of the output structure file, and an ordered list of operation codes (*opcodes)* for the calculations.

When you start a job from Maestro or write out the input files, Maestro writes out a molecular structure file and a command file. For many types of jobs, command files written this way are complete and adequate, but for some types of jobs you may need to adjust the Maestro-generated command file.

The command files and the log files for the examples given in this section can be found in `$SCHRODINGER/macromodel-v`*version*`/samples/Examples`.

Below is an example command file for a dihedral scan calculation and explanations of the opcodes used in the files.

```
dscan.mae
dscan-out.mae
 MMOD     0     1     0     0      0.0000      0.0000      0.0000      0.0000
 FFLD    10     1     0     1      1.0000      0.0000      0.0000      0.0000
 READ     0     0     0     0      0.0000      0.0000      0.0000      0.0000
 BGIN     0     0     0     0      0.0000      0.0000      0.0000      0.0000
 DRIV     1    14    15    16      0.0000    360.0000     30.0000      0.0000
 DRIV     6    25    26    27      0.0000    360.0000     30.0000      0.0000
 CONV     2     0     0     0      0.0500      0.0000      0.0000      0.0000
 MINI     9     0   500     0      0.0000      0.0000      0.0000      0.0000
 END      0     0     0     0      0.0000      0.0000      0.0000      0.0000
```

`MMOD`: Creates and updates an intermediate structure file so that structures can be displayed in Maestro as the job progresses.

`FFLD`: Force field selection. Arg1 denotes the actual force field used in the calculation (in this case MMFF94). Arg2 defines the electrostatic treatment for the calculation. The default (arg2=0) is to use the dielectric treatment encoded in the force field. In this case a constant dielectric (arg2=1), with a dielectric constant of 1 explicitly requested (arg5). Arg4 is

MMFF94-specific. Arg4=1 defines the MMFF94s version of the force field, ensuring planarity around delocalized *sp*^{2} nitrogens.

`READ`: Directs MacroModel to read the input file.

`BGIN`/`END`: Defines the start/end of a loop. All commands between the `BGIN` and `END` lines will be performed for each structure in the input file.

`DRIV`: Directs MacroModel to carry out a coordinate scan using the indicated dihedral angles (arg5=0° to arg6=360°) and step sizes (arg7=30°).

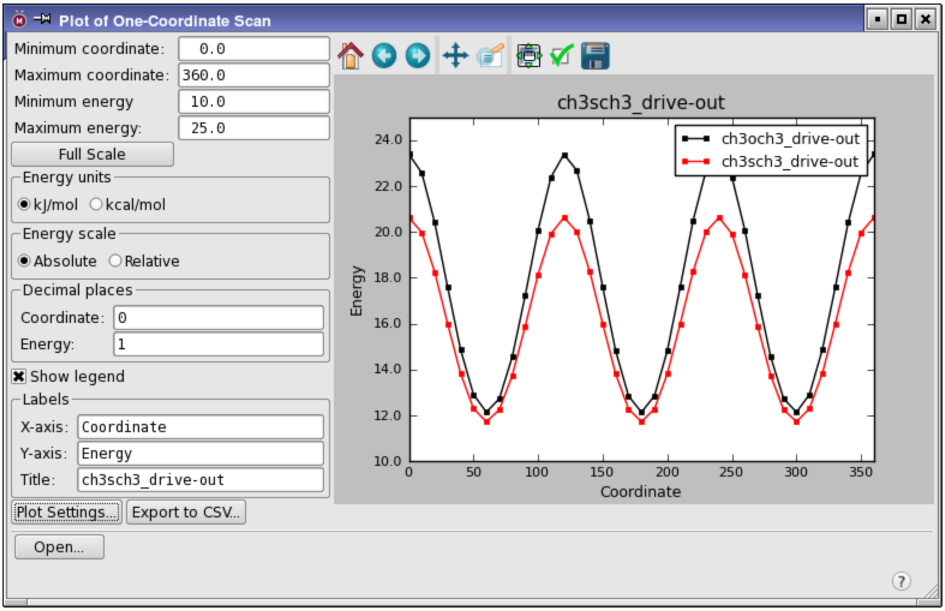
`CONV`: Defines convergence criteria. Arg1=2 signifies derivative convergence. The default criterion, if no `CONV` command is present, is 0.05kJ/mol-Å; this value is set in arg5.

`MINI`: Starts the minimization. Arg1 defines the type of minimization algorithm to be used. Arg1=9 means that Truncated Newton-Raphson Conjugate Gradient will be used. In arg3, the number of minimization steps is defined. Arg3 can be set to a large number since the calculation automatically stops as soon as the convergence criterion is reached.

7.3 Checking and Interpreting Results

Coordinate scan calculations involve minimizations, and you should check that minimizations for all coordinate values converge.

You can display the results of a scan in one or two coordinates in the Plot of One-Coordinate Scan panel or the Plot of Two-Coordinate Scan panel, which you open by choosing Tools → Plot Coordinate Scan → One Coordinate or Two Coordinates. For more information, see Section 11.2 of the *Maestro User Manual*. Examples of plots from coordinate scans are shown in Figure 7.2 and Figure 7.3.



Plot of One-Coordinate Scan

Minimum coordinate: 0.0
Maximum coordinate: 360.0
Minimum energy 10.0
Maximum energy: 25.0

Energy units
● kJ/mol ○ kcal/mol

Energy scale
● Absolute ○ Relative

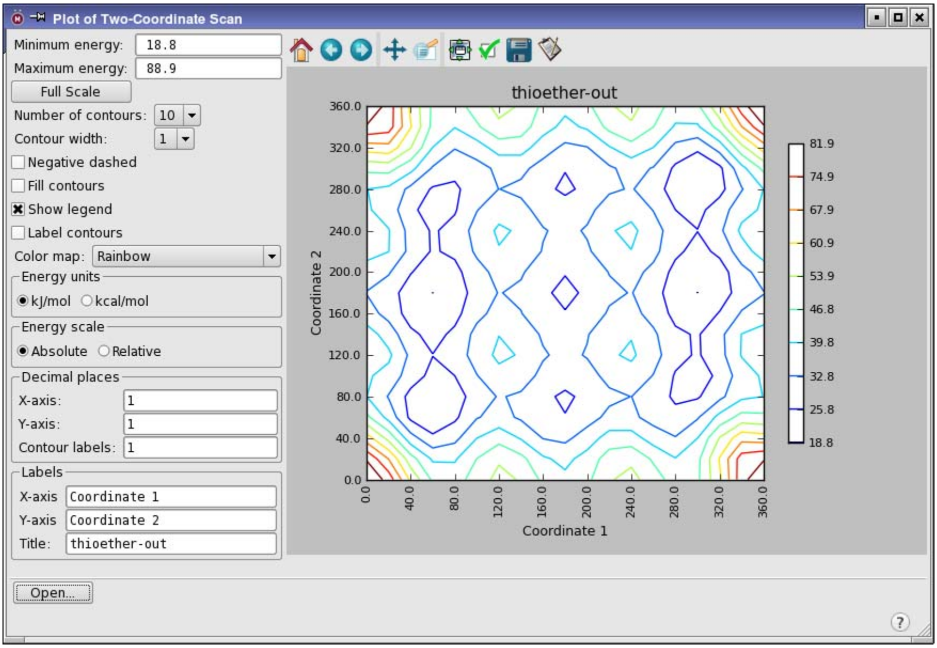
Decimal places
Coordinate: 0
Energy: 1

☒ Show legend

Labels
X-axis: Coordinate
Y-axis: Energy
Title: ch3sch3_drive-out

Full Scale
ch3sch3_drive-out
ch3och3_drive-out
ch3sch3_drive-out
Energy
24.0
22.0
20.0
18.0
16.0
14.0
12.0
10.0
0
50
100
150
200
250
300
350
Coordinate
Plot Settings...
Export to CSV...
Open...

Figure 7.2. The Plot of One-Coordinate Scan panel.



Plot of Two-Coordinate Scan

Minimum energy: 18.8
Maximum energy: 88.9
Number of contours: 10
Contour width: 1
☐ Negative dashed
☐ Fill contours
☒ Show legend
☐ Label contours
Color map: Rainbow

Energy units
● kJ/mol ○ kcal/mol

Energy scale
● Absolute ○ Relative

Decimal places
X-axis: 1
Y-axis: 1
Contour labels: 1

Labels
X-axis Coordinate 1
Y-axis Coordinate 2
Title: thioether-out

Full Scale
thioether-out
Coordinate 2
360.0
320.0
280.0
240.0
200.0
160.0
120.0
80.0
40.0
0.0
0.0
40.0
80.0
120.0
160.0
200.0
240.0
280.0
320.0
360.0
Coordinate 1
81.9
74.9
67.9
60.9
53.9
46.8
39.8
32.8
25.8
18.8
Open...

Figure 7.3. The Plot of Two-Coordinate Scan panel.

Conformational Searches

A frequent question in molecular modeling studies is, What conformations are important in this system? One way to address this question is to perform a conformational search in order to find a set of low energy conformers. MacroModel excels at conformational searching with a number of state-of-the-art conformational searching methods, described in the sections below. Conformational searches typically cycle through the process of generating a new structure, minimizing it, and then determining if the structure should be retained. Structure retention can be based upon energy relative to that of the lowest energy conformer found so far in the search and redundancy with other generated structures.

8.1 Conformational Search Methods

A variety of conformational search methods is available. The methods available in MacroModel are described below.

Monte Carlo Multiple Minimum (MCMM)

The Monte Carlo method (MCMM) implemented in MacroModel is highly efficient in performing global searching, exploring close as well as distant areas of the potential energy surface (PES). Random changes are made in torsion angles during the search. Although there is no limit to the number of variable torsions allowed in the search, more than about 10 or 15 flexible torsions greatly increases the complexity of the search. You should therefore expect increased searching times, and possibly also non-convergence of the search. See References 20 and 21 for more information.

Systematic Pseudo-Monte Carlo (SPMC)

If the task is to locate all possible conformational minima on the surface, MacroModel's systematic method (SPMC) ensures the exploration of unusual torsional values. SPMC allows for high efficiency throughout the search, even at the end where other methods tend to revisit already found conformers. The SPMC searching method is most efficient for smaller molecules. See Reference 22 for more information.

Low-Mode Conformational Search Methods

If you have little or no prior knowledge of the system to be searched, the Low-Mode Conformational Search (LMCS) [23], based on the principles behind saddle point searching, allows for automatic searching without the need to define parameters to be varied during the search. One important consequence of this is that LMCS allows for a serial search of multiple different molecules without user intervention.

A related method, Large Scale Low-Mode (LLMOD) [24, 25], has been developed for large-scale conformational searching, such as protein loop optimization, homology model refinement, and fully flexible docking for induced fit modeling. LLMOD is similar to LMOD, but computes low-mode eigenvectors of a Hessian matrix that is referenced only implicitly, through its product with a series of vectors. LLMOD is the first conformational search method that can be applied to fully flexible, unconstrained protein structures. Serial LLMOD calculations are not currently supported.

Mixed MCMM/Low-Mode Conformational Search Methods

A pure LMCS search is local in scope, but performs exceptionally well in global searches through hybridization with MCMM [26]. By defining key torsions to be varied through MCMM steps, the mixed MCMM/LMCS search has proved faster and more efficient than any other searching algorithm for a variety of systems. MCMM/LMCS has successfully been applied to difficult tasks, such as searching the conformational space of a ligand in the active site of a protein. Mixed MCMM/LMC2 searches are also supported.

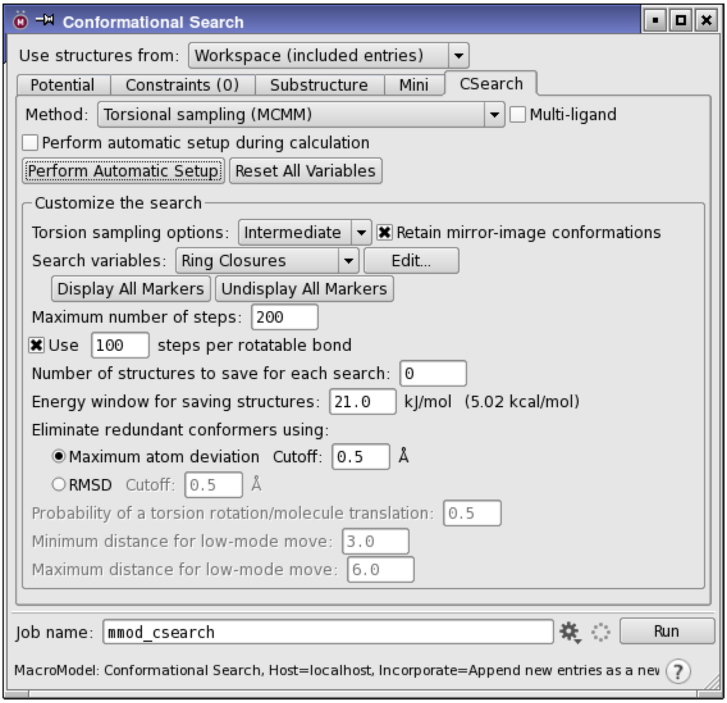
8.2 Performing Conformational Searches

You can use the Conformational Search panel to set up and submit MacroModel conformational search calculations. The panel has six parts, five of which are common to other MacroModel panels. For more information about the controls in the Mini tab, see Section 6.1 on page 51. For information about the upper portion of the Conformational Search panel and the Potential, Constraints, and Substructure tabs, see Section 4.1 on page 31 through Section 4.5 on page 40.

The CSearch tab is unique to the Conformational Search panel. Controls in this tab allow you to chose a search method; define an energy window for saving structures; set up torsion, chiral center, and distance checks; and define other search parameters.

To open the panel, select Conformational Search from the MacroModel menu in the main menu bar. If the MacroModel menu is not in the main menu bar, choose MacroModel from the Applications menu.

To set up a conformational search, you must first select any entries that you want to use as input from the Project Table, or display the structure you want to use in the Workspace. In the Conformational Search panel, set the options in the upper portion of the panel and in the Potential, Constraints, and Substructure tabs, then set the options in the Mini tab.



Use structures from: Workspace (included entries)

Method: Torsional sampling (MCMM) ☐ Multi-ligand

☐ Perform automatic setup during calculation

Customize the search

Torsion sampling options: Intermediate ☒ Retain mirror-image conformations

Search variables: Ring Closures

Maximum number of steps: 200

☒ Use 100 steps per rotatable bond

Number of structures to save for each search: 0

Energy window for saving structures: 21.0 kJ/mol (5.02 kcal/mol)

Eliminate redundant conformers using:

● Maximum atom deviation Cutoff: 0.5 Å

○ RMSD Cutoff: 0.5 Å

Probability of a torsion rotation/molecule translation: 0.5

Minimum distance for low-mode move: 3.0

Maximum distance for low-mode move: 6.0

Job name: mmod_csearch

MacroModel: Conformational Search, Host=localhost, Incorporate=Append new entries as a nev

Figure 8.1. The CSearch tab of the Conformational Search panel.

With these portions of the job setup complete, set the parameters of the conformational search in the CSearch tab. The controls in this tab are described below. When you have finished adjusting the settings in the CSearch tab, click Run to start the job with the current job settings, click the Settings button to set up and launch the job or choose Write from the Settings button menu to write the job commands to a file to be run or manually edited later.

8.2.1 Conformational Search Method

You can select the conformational search method to be used during a calculation from the Method option menu. The conformational search method options are described below.

Torsional sampling (MCMM)

This is the recommended conformational search method. The input structure is modified by random changes in torsion angles and/or molecular position as specified in the panels opened by the Torsion Rotations or Molecule Trans/Rot buttons. Ordinarily, whether a single structure or multiple structures appear in the input file, they will first be read in, minimized, and treated as if already found by the MCMM procedure. This allows a new search to be initialized from the output of a previous search, by using the output file of the old search as input for the new one. See page 75, and the `MCMM` opcode description in the *MacroModel Reference Manual*, for a detailed description of this method.

Serial torsional sampling (MCMM) is an automated procedure for performing a separate MCMM search on each structure in the input structure file, and is enabled by selecting Multi-ligand, next to the Method option menu. For this procedure, Perform automatic setup during calculation is selected and cannot be deselected.

Systematic torsional sampling (SPMC)

This method is similar to MCMM, but uses a systematic search instead of a random search. The search begins at low torsional resolution (120°), searches all angles without duplicating coverage, then doubles the resolution. This method has the advantage of not retracing its path and consequently converges the final stages of the conformational search more efficiently than MCMM. Like MCMM, the method is effectively open-ended: it will search conformational space until you stop it. See page 75, and the `SPMC` opcode command description in the *MacroModel Reference Manual*, for a detailed description of this method.

Low-mode sampling

This method, termed LMOD, is highly efficient and has the advantage that ring structures and variable torsion angles do not need to be specified. This conformational search method works by exploring the low-frequency eigenvectors of the system, which are expected to follow “soft” degrees of freedom, such as torsions. LMOD methods search conformational space aggressively enough to switch the chirality of atoms within the structures provided. Chirality checking should be used for chiral atoms for which such chirality switching is undesirable (see Section 8.2.2 on page 79). See page 75, and the `LMCS` opcode description in the *MacroModel Reference Manual*, for a detailed description of this method.

Serial low-mode sampling is an automated procedure for performing a separate low-mode search on each structure in the input structure file, and is enabled by selecting Multi-ligand, next to the Method option menu. For this procedure, Perform automatic setup during calculation is selected and cannot be deselected.

Mixed torsional/Low-mode sampling

This method uses a combination of the random changes in torsion angles and/or molecular position from the MCMM method, together with the low-mode steps from the LMOD method used in pure low-mode. See page 76, and the `MCMM` and `LMCS` opcode descriptions in the *MacroModel Reference Manual*, for a detailed description of this method.

You can use this method to perform a separate search on each structure in the input structure file by selecting Multi-ligand, next to the Method option menu. For this procedure, Perform automatic setup during calculation is selected and cannot be deselected.

Large scale low-mode sampling

Large Scale Low-Mode (LLMOD) is similar to Low-Mode (LMOD), except that it uses techniques to reduce the memory requirements so that it can be applied to much larger systems such as protein-ligand complexes. Like LMOD, LLMOD methods search conformational space aggressively enough to switch the chirality of atoms within the structures provided. Chirality checking should be used for chiral atoms for which such chirality switching is undesirable (see Section 8.2.2 on page 79). See the `LMC2` and `ARPK` opcode descriptions in the *MacroModel Reference Manual* for a detailed description of this method.

Large-scale low-mode sampling can use a large amount of memory. If memory issues are encountered, consider eliminating the use of solvation. In addition, reducing non-bonded cutoffs also reduces the memory required for the computation. You can do this in the Potential tab by selecting Normal from the Cutoff option menu, or by selecting User defined and entering the appropriate reduced values in the text boxes.

Mixed torsional/Large scale low-mode sampling

This method uses a combination of the random changes in torsion angles and/or molecular position from the MCMM method, together with the low-mode steps from the LLMOD method used in Large scale low-mode. See the `MCMM`, `LMC2`, and `ARPK` opcode descriptions in the *MacroModel Reference Manual* for detailed descriptions of this method.

8.2.2 Automatic Setup of Conformational Search Variables

Before you start a conformational search, the values for certain variables need to be set, such as which torsions are to be rotated and which rings are to be opened. There are three ways to set up conformational search variables from the CSearch tab: using Perform Automatic Setup, using Perform automatic setup during calculation, or setting variables manually. Setting conformational search variables manually is described in the next section.

The automatic setup procedures select comparison atoms and other conformational search parameters for you. For MCMM and mixed MCMM/LMOD calculations, comparison atoms, chiral atoms, torsion checks, molecule moves, variable torsions, and ring closures are automatically set up. For low-mode and large-scale low-mode calculations, comparison atoms, chiral atoms, and torsion checks are automatically set up.

Perform Automatic Setup

Use the Perform Automatic Setup button to automatically generate all the necessary variables for the search. The settings chosen by the automatic setup procedure can then be reviewed by choosing a variable type from the Search variables option menu and clicking Edit. The corresponding panel opens, in which you can define and edit the conformational search variables. These panels are described in Section 8.2.5 on page 82. Automatic setup is not available for all conformational search methods; if it is not, the Perform Automatic Setup button is dimmed. Perform Automatic Setup applies only to the structures in the Workspace. To set up conformational search variables for other structures, use Perform automatic setup during calculation.

If you perform an automatic setup on an entry that contains a substructure, only the atoms in the substructure are automatically set up with MCMM parameters and comparison atoms. Atoms in fixed and frozen regions are not included in automatic setup.

To clear any previously set conformational search variables before performing an automatic setup, click Reset All Variables.

Perform automatic setup during calculation

Perform automatic setup during calculation performs the same tasks as Perform Automatic Setup, but it permits you to set up parameters for all the structures that are processed when the search is performed. This option disables the controls used with Perform Automatic Setup. You can use this option with any of the conformational search methods, but you must use it with serial MCMM searches. By default, it is selected. This option inserts `AUTO` opcodes into the command file. See Section 7.4 on page 63 and the *MacroModel Reference Manual* for more information on `AUTO`.

Note: Automatic setup should not be used with constraints. If you set constraints and use automatic setup, MacroModel can fail because the automatic setup can request movement of atoms that have constraints.

Torsion sampling options

You can use the options on this menu, in the Customize the search section, to determine which of the torsions around amide and ester linkages and other planar groups (azo and imido groups) are selected for sampling during Automatic Setup. The options are as follows:

- Restricted—Do not sample amide and ester derivatives or other planar groups. Torsional constraints are applied to ensure that the relative conformation of these groups is maintained during the MCMM and minimization process. (Sets `AUTO` arg8=0.)
- Intermediate—Sample the torsions of amide or ester linkages of non-standard groups like anhydrides, carbamates, hydrazones, and so on. Normal ester and amide linkages are not sampled, nor are azo and imido linkages. (Sets `AUTO` arg8=1.)

- Enhanced—Sample all amide-like and ester-like linkages, including standard amides and esters. Azo and imido linkages are not sampled. (Sets `AUTO` arg8=2.)
- Extended—Sample rotations around C=N and N=N bonds, in addition to all amide and ester derivatives. (Sets `AUTO` arg8=3.)

8.2.3 Global Search Restrictions

In the Customize the search section, you can set restrictions on the scope of the search and its output. In addition, you can specify the criterion for equivalent structures in the lower portion of the panel.

Retain mirror-image conformations

By default, the search algorithm compares each trial structure and its mirror image with the list of unique structures to determine if the trial structure is unique. If you select this option, mirror images are considered to be separate structures and both structures are retained.

Maximum number of steps

For each search method, specify the number of steps to be performed in the Maximum number of Steps text box. When the number of generated trial structures matches the value in field, the conformational search is terminated.

Use *N* steps per rotatable bond

Use this text box to specify the number of search steps per rotatable bond. The total number of steps is still limited by the Maximum number of steps setting. This option is useful for serial searches, where the structures might have different numbers of rotatable bonds. The default is 100.

Number of structures to save for each search

Use this text box to specify the number of unique structures to save when the search is complete, counting from the structure that is lowest in energy. If the value is zero, all unique structures are saved.

Energy window for saving structures

The default value is set to 21 kJ/mol. The value in this text box is used to compare trial structures. Any new structures generated and minimized are saved as results of the search only if they are within the energy window value above the current global minimum. Lowering this value results in the search saving fewer structures.

Eliminate redundant conformers using

If comparison atoms are chosen or Perform automatic setup during calculation is selected, structures produced during a conformational search are compared to see if they are unique. Two options are available for the conditions for structures to be considered different:

- Maximum atom deviation—consider structures to be different if the maximum atom deviation for any atom exceeds the threshold given in the Cutoff text box.
- RMSD—consider structures to be different if the RMS deviation for all compared atoms exceeds the threshold given in the Cutoff text box.

The default cutoff is 0.5 Å for both options. Unless there are fixed or frozen atoms, the structures are superimposed prior to calculating the distances.

8.2.4 Low-Mode Parameters

In the lower portion of the Customize the search section are three text areas that are relevant only to low-mode searches. These are active only when any of the four methods involving Low-Mode Conformational Searching is the current search method.

Probability of a torsion rotation/molecule translation

This control is available only with the mixed methods. This text box allows you to set a probability that any defined torsion rotations and molecule translations will be made during the search. The value should be a number from 0.0 to 1.0.

Minimum distance for low-mode move
Maximum distance for low-mode move

Used for setting the minimum and maximum distance for a low-mode move. During a search, the fastest moving atom is moved at randomly generated distances that are between the minimum and maximum values specified in these text boxes.

8.2.5 Setting Conformational Search Variables Manually

Automatic setup of search variables can be used for most purposes. There are occasions in which you may want to set conformational search variables manually, or to examine them. To do so, select the search variable from the Search variables option menu, and click Edit. This button opens a panel that can be used to set the chosen kind of variable for a conformational search.

A description of each of these panels is given below. Each panel has some common features: a text box at the top that lists the defined features; a Markers option, which controls display of the defined features; a Delete button, which deletes the selected definition from the list; and a

Delete All button, which deletes all the defined features in the list. In any of the subpanels, you can edit a defined feature by selecting it in the list and picking Workspace atoms.

Markers can be displayed in the Workspace to identify the conformational search variables. Most of the panels have a Markers option to control the display of markers for the variables set in that panel. When you close the panel, the markers are undisplayed. You can also display or undisplay markers for all variables by clicking Display All Markers or Undisplay All Markers.

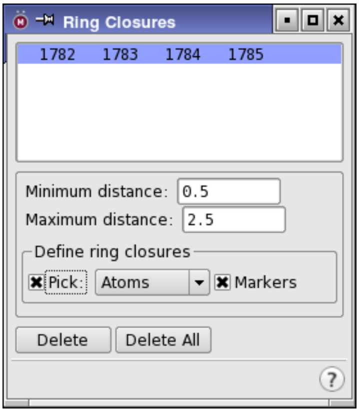
To clear any previously set conformational search variables, click Reset All Variables.

8.2.5.1 Ring Closures

During a Monte Carlo-based conformational search (MCMM or SPMC), rings must be opened before varying their torsions. Then, after random torsion variations have been evaluated, the ring must be re-closed. The appropriate location for ring closures must be defined before a calculation can be started. “Ring Closures” consist of four atoms that define the bond to be broken (and then re-closed) during an MCMM or SPMC step.

The simplest way to create ring closures is to use the Perform Automatic Setup button. MacroModel makes reasonable decisions about where a ring should be opened. The list of atoms in the ring that will be opened appears in the text box at the top of the Ring Closures panel. Each ring closure is defined by four atoms. The opening is made between the second and third atoms. If Markers is selected, the ring closure is marked by a light green line, which is broken between the second and third atoms. A lightning bolt is also placed between the second and third atoms to show where the opening will be made.

The functions of the Ring Closures panel are described below.



1782 1783 1784 1785

Minimum distance: 0.5

Maximum distance: 2.5

Figure 8.2. The Ring Closures panel.

Minimum distance

You must specify the minimum acceptable distance between the second and third ring closure atoms. If the distance between the ring closure atoms is less than this minimum distance, then the ring is reopened and a different set of random variations is performed. The default value for this is 0.5 Å. This value should suffice for most searches.

Maximum distance

The maximum acceptable distance between the second and third atoms of a re-closed ring can be specified. If the distance between the ring closure atoms is greater than this maximum distance, the ring is reopened and a different set of random variations is performed. The default value for this is 2.5 Å. This value should suffice for most searches.

Define ring closures

To define the location of a ring closure manually, choose Atom or Bond from the Pick menu and pick four atoms or three bonds from the ring. Picked atoms are marked by purple boxes. The bond is broken between the second and third picked atoms. Once four atoms have been picked, a new entry is displayed in the list and the ring closure is marked by the green line and lightning bolt, as described above.

8.2.5.2 Torsion Rotations

During a Monte Carlo conformational search, random changes are performed on the structure of the molecule. Structure energies are then evaluated. To set up a Monte Carlo-based search using either the MCMM or SPMC method, you must specify which torsions in the molecule can be rotated. These torsions are defined by pairs of atoms (defining bonds around which the structures can be rotated) using the Torsion Rotations panel.

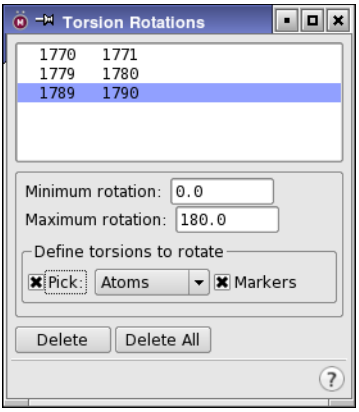
The simplest way to define torsion rotations is to use the Perform Automatic Setup button. MacroModel detects torsions that need to be rotated and generates a list of torsion rotation definitions. This list appears in the text box at the top of the Torsion Rotations panel.

If Markers is selected, the bonds defined as torsion rotations are indicated in the Workspace by a purple line and an icon (a purple horizontal line encircled by an arrow).

The functions of the Torsion Rotations panel are described below.

Minimum rotation

The minimum acceptable rotation of a torsion must be specified. Each time that a torsion must be rotated, an increment for rotation larger than this value is selected. The default minimum rotation is 0°.



1770 1771
1779 1780
1789 1790

Minimum rotation: 0.0
Maximum rotation: 180.0

Define torsions to rotate

☒ Pick: Atoms ☒ Markers

Figure 8.3. The Torsion Rotations panel.

If a torsion rotation is added for a double or amide bond (i.e., to search both E- and Z- isomers around these bonds), the minimum value for rotations should be set to 90°. This maximizes the probability that interconversion between the two isomers will occur.

Maximum rotation

This text field allows you to specify a value for the maximum acceptable torsion rotation increment. Each time that a torsion must be rotated, an increment for rotation smaller than this value is selected. The default maximum rotation is 180°.

Define torsions to rotate

To define torsion rotations manually, choose Atom or Bond from the Pick menu, and pick two atoms or one bond in the Workspace structure to define the torsion rotation. When you pick the first of a pair of torsion rotation atoms, Maestro places a crimson box around it. Once both the atoms have been picked, a new entry is displayed in the list at the top of the panel, and the bond is marked as described above if Markers is selected.

8.2.5.3 Molecule Trans/Rot

During a Monte Carlo conformational search, random changes are performed on the structure of the molecule. Structure energies are then evaluated to find the lowest-energy structure possible. To set up a Monte Carlo-based search using either the MCMM or SPMC method, you must specify which molecules will be translated and rotated. The Molecule Trans/Rot panel facilitates the specification of molecules to be rotated and translated during a Monte Carlo (MCMM or SPMC) conformational search.

The simplest way to define molecules for translation and rotation is to use the Perform Automatic Setup button. MacroModel locates molecules that need to be translated and rotated, and generates a list of molecule translation and rotation definitions. This list appears in the text box at the top of the Molecule Trans/Rot panel.

Maestro labels the selected molecules with peach-colored vertical arrows encircled by "rotation" arrows if Show markers is selected. The currently selected molecule is distinguished by a turquoise label.

The controls of the Molecule Trans/Rot panel are described below.

Minimum rotation

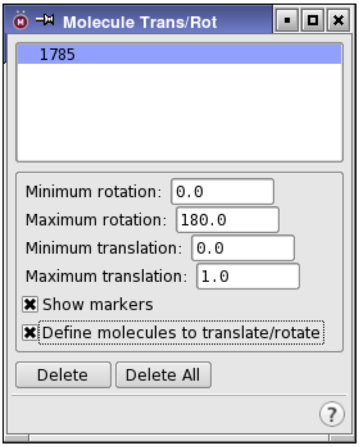
The minimum acceptable rotation of a molecule must be specified. Each time a molecule must be rotated, an increment for rotation that is larger than this value is selected. The default minimum rotation is 0°.

Maximum rotation

The Maximum rotation text field allows you to specify a value for the maximum acceptable molecule rotation increment. Each time a molecule must be rotated, an increment for rotation that is smaller than this value is selected. The default maximum rotation is 180°.

Minimum translation

The value in the Minimum translation text box sets the lower limit for molecule translation. Each time a molecule must be translated, a random increment larger than this value is used. The default value is 0.0.



1785

Minimum rotation: 0.0
Maximum rotation: 180.0
Minimum translation: 0.0
Maximum translation: 1.0
☒ Show markers
☒ Define molecules to translate/rotate

Figure 8.4. The Molecule Trans/Rot panel.

Maximum translation

This text box sets the upper limit for molecule translation. Each time a molecule must be translated, a random increment smaller than this value is used. The default value is 1.0.

Define molecules to translate/rotate

To define molecules to be translated or rotated manually, click on an atom in the Workspace structure. (Define molecules to translate/rotate is selected by default.) A new entry is displayed in the list at the top of the panel. The atom is colored peach and the vertical arrow icon is placed by the atom in the Workspace. Although only the picked atom is marked, the entire molecule is selected. You should not pick more than one atom in any molecule, even though this operation is allowed.

8.2.5.4 Comparison Atoms

During conformational searches, new structures are generated and minimized. The structures are compared against other low-energy structures that have already been found in the search. The comparison is performed by rigid superposition, comparing only those atoms specified as “comparison atoms” in the setup.

The simplest way to define comparison atoms is to click Perform Automatic Setup. MacroModel locates comparison atoms and generates a list of the atoms. This list appears in the text box at the top of the Comparison Atoms panel.

The features of the Comparison Atoms panel work in the same way as for multiple minimization (see page 61).

To define comparison atoms, use one or more of the following options:

- Click Heavy Atoms + O-H,S-H.

  This adds all non-hydrogen atoms and the hydrogen atoms attached to oxygen and sulfur to the list of comparison atoms.

- Click Heavy Atoms.

  This adds only the non-hydrogen atoms to the list of comparison atoms.

- Pick atoms in the Workspace.

  Choose an object from the Pick menu, and click atoms in the Workspace to add the atoms to the list of comparison atoms.

- Select atoms using the Atom Selection dialog box.

  For more complex combinations of comparison atoms, click the Atom Selections button and choose Select to select atoms using the Atom Selection dialog box.

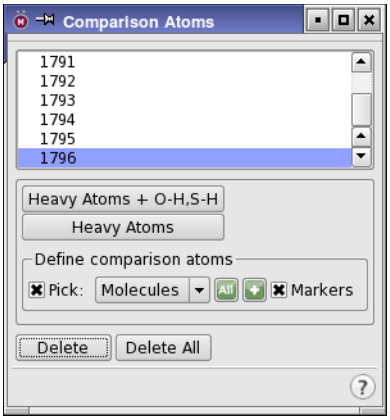

Comparison Atoms
1791
1792
1793
1794
1795
1796
Heavy Atoms + O-H,S-H
Heavy Atoms
Define comparison atoms
☒ Pick: Molecules
All
☒ Markers
Delete
Delete All
?

Figure 8.5. The Comparison Atoms panel.

- Select all atoms.

  Click All to add all atoms to the list of comparison atoms.

Comparison atoms are marked in light green in the Workspace with an "=" icon beside them. The currently selected comparison atom is marked in aquamarine.

8.2.5.5 Chiral Atoms

Because Monte Carlo conformational searches can generate and then minimize highly strained structures, chiral atoms in a molecule might be inverted. To prevent structures with inverted centers from appearing on the final list of optimized structures, you must identify a molecule's chiral atoms before beginning a search. Once defined, the chirality of each center is compared against that in the starting structure. If inversion has occurred, the search result is rejected.

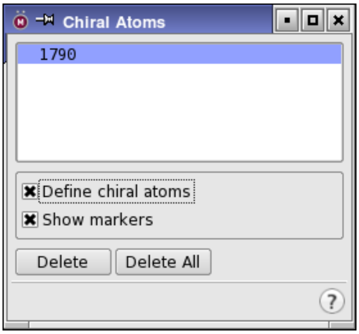

Chiral Atoms
1790
☒ Define chiral atoms
☒ Show markers
Delete
Delete All
?

Figure 8.6. The Chiral Atoms panel.

The simplest way to define chiral atoms is to click Perform Automatic Setup. MacroModel locates chiral atoms and generates a list of the atoms. This list appears in the text box at the top of the Chiral Atoms panel.

If it is necessary to define chiral atoms manually, check that Define chiral atoms is selected (the default), and pick the atoms in the Workspace to add them to the list.

Chiral atoms are marked in peach with “R|S” labels beside them. The selected atom is colored turquoise. To hide these markers, deselect Show markers.

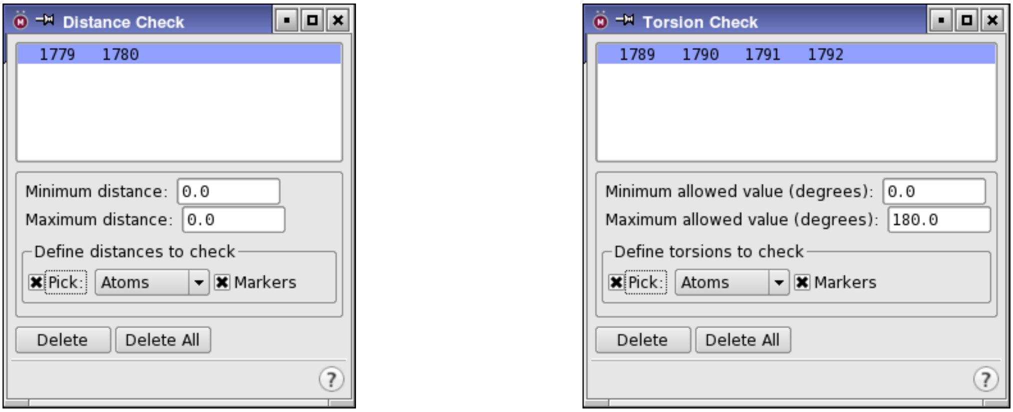
8.2.5.6 Distance Check

Occasionally you might want to restrict the scope of a conformational search to generate only structures that are consistent with certain geometric constraints—for example, when experimentally obtained results such as NOE constraints are available. Distance checks are used to reject structures in which certain distances do not meet the specified criteria.

Distance checks cannot be set using the automatic setup process. They must be added by picking atoms from the structure in the Workspace.

To define a distance for checking, choose either Atom or Bond from the Pick menu, then pick two atoms or one bond in the Workspace. A new entry is added to the list at the top of the panel. Maestro marks the defined distance check with a purple dotted line and a “check” icon. The selected distance check is distinguished by two solid lines on either side of the dotted line.

For each pair of atoms that you select, you must define the minimum and maximum distances in the Minimum distance and Maximum distance text boxes. Structures with distances that lie outside this range are rejected.



Distance Check

1779 1780

Minimum distance: 0.0

Maximum distance: 0.0

Define distances to check

☒ Pick: Atoms ☒ Markers

Torsion Check

1789 1790 1791 1792

Minimum allowed value (degrees): 0.0

Maximum allowed value (degrees): 180.0

Define torsions to check

☒ Pick: Atoms ☒ Markers

Figure 8.7. The Distance Check and Torsion Check panels.

8.2.5.7 Torsion Check

Because highly strained structures may be generated during Monte Carlo conformational searches, the geometry around double or amide bonds may be changed (from E- to Z- isomers, for example). You might want to restrict the scope of the conformational search to structures that retain the original torsions around these bonds. Torsion checks can be used to reject structures that do not retain the original geometry for these special cases.

The simplest way to define torsion checks is to click Perform Automatic Setup. MacroModel generates a list of the amides and double bonds in a structure. This list appears in the text box at the top of the Torsion Check panel.

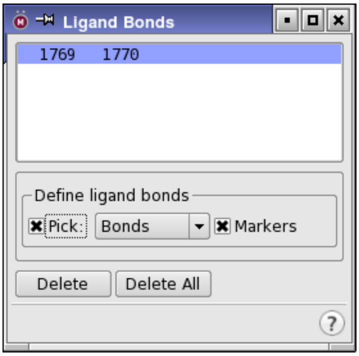
Torsion checks can be added by picking atoms in the Workspace. To define a torsion for checking, choose either Atom or Bond from the Pick menu, then pick four atoms or three bonds in the Workspace. A new entry is added to the list at the top of the Torsion Check panel, and the torsion check is marked with a solid red line and a check mark. The currently selected torsion check is marked by solid lines on either side of the red line.

For each torsion check, the minimum and maximum torsional angle must be defined. Any search structures in which the checked torsional angle is not between the minimum and maximum angle is rejected and not included in search results.

8.2.5.8 Ligand Bonds

Conformational searches of inorganic complexes produce structures in which ligand positions vary with respect to the metal center. Maestro permits the specific definition of the ligand bonds around which the reorientation takes place.

To use this feature, choose Atom or Bond from the Pick menu and click on the desired bonds or atom pairs that define these bonds in the Workspace. A new entry appears in the list at the top of the panel, and Maestro marks the bond with a purple dotted line and a scissor icon. The selected bond is distinguished by purple lines on either side of the dotted line.



1769 1770

☒ Pick: Bonds ☒ Markers

Figure 8.9. The Ligand Bonds panel.

After the bonds have been selected, click Perform Automatic Setup in the Conformational Search panel. Maestro generates molecular rotation and translation commands for each fragment created by the defined ligand bonds. In addition, it produces torsion rotations, torsion checks, and chiral atom definitions. To view or edit the automatically generated settings, open the Torsion Rotations, Torsion Check, or Chiral Atoms panels.

8.3 Command File Examples

To run MacroModel calculations, a molecular structure file and a command file are required. The molecular structure file contains the structures to be used as input in the calculation. The command file contains the name of the input structure file, the name of the output structure file, and an ordered list of operation codes (*opcodes)* for the calculations.

When you start a job from Maestro or write out the input files, Maestro writes out a molecular structure file and a command file. For many types of jobs, command files written this way are complete and adequate, but for some types of jobs, you may need to adjust the Maestro-generated command file.

The command files and the log files for the examples given in this section can be found in `$SCHRODINGER/macromodel-v`*version*`/samples/Examples`.

8.3.1 Conformational Search Using MCMM

An example command file appears below for a conformational search calculation that uses the MCMM search method. Descriptions of the opcodes in the file follow.

```
mcmm.mae
mcmm-out.mae
 MMOD      0      1      0      0     0.0000     0.0000     0.0000     0.0000
 FFLD     10      1      0      1     1.0000     0.0000     0.0000     0.0000
 BDCO      0      0      0      0    41.5692 99999.0000     0.0000     0.0000
 READ      0      0      0      0     0.0000     0.0000     0.0000     0.0000
 MCMM    100      0      0      0     0.0000     0.0000     0.0000     0.0000
 MCNV      2      4      0      0     0.0000     0.0000     0.0000     0.0000
 MCSS      2      0      0      0    50.0000     0.0000     0.0000     0.0000
 MCOP      1      0      0      0     0.0000     0.0000     0.0000     0.0000
 DEMX      0    166      0      0    50.0000   100.0000     0.0000     0.0000
 MSYM      0      0      0      0     0.0000     0.0000     0.0000     0.0000
 COMP      1      2      3      4     0.0000     0.0000     0.0000     0.0000
 COMP      5      6      7      9     0.0000     0.0000     0.0000     0.0000
 COMP     10     14     15     16     0.0000     0.0000     0.0000     0.0000
 COMP     17     25     26     27     0.0000     0.0000     0.0000     0.0000
 COMP     28     29     30     31     0.0000     0.0000     0.0000     0.0000
 TORS      1     14      0      0     0.0000   180.0000     0.0000     0.0000
 TORS      6     25      0      0     0.0000   180.0000     0.0000     0.0000
 TORS      14      15       0       0      0.0000    180.0000      0.0000      0.0000
 TORS      15      16       0       0      0.0000    180.0000      0.0000      0.0000
 TORS      25      26       0       0      0.0000    180.0000      0.0000      0.0000
 CONV       2       0       0       0      0.0500      0.0000      0.0000      0.0000
 MINI       9       0     500       0      0.0000      0.0000      0.0000      0.0000
```

`MMOD`: Creates and updates an intermediate structure file so that structures can be displayed in Maestro as the job progresses.

`FFLD`: Force field selection. Arg1 defines the force field used in the calculation (in this case MMFF94). Arg2 defines the electrostatic treatment for the calculation. In this case a constant dielectric is used. Arg4 is MMFF94 specific: arg4=1 defines the MMFF94s version of the force field, ensuring planarity around delocalized *sp*^{2} nitrogens.

`BDCO`: Use the Bond Dipole CutOff (`BDCO`) method for truncating electrostatic interactions. Arg5 and arg6 are used to specify the cutoffs used for charge-dipole and charge-charge interactions, respectively.

`READ`: Read the input file.

`MCMM`: Use Monte Carlo Multiple Minimum searching. Arg1 defines the number of MC steps for the search.

`MCNV`: Sets the number of degrees of freedom to be varied at each MC step. If arg1 and arg2 differ, the search varies a random number of degrees of freedom between the numbers defined in arg1 and arg2. We recommend setting arg1=2 and arg2=maximum number of degrees of freedom.

`MCSS`: Select starting structure for the search steps. Arg1=2 defines use-directed selection of starting structures, where the least used structures will be used as starting geometries, as long as they are low enough in energy (as defined in arg5). This is more efficient in exploring new areas of the potential energy surface than, for instance, a random-walk starting geometry scheme. Arg5 gives the energy window for selecting a new starting structure. The new starting structure must be within arg5 kJ/mol of the lowest energy conformer found in the search.

`MCOP`: Monte Carlo options determine what and how often data is written to the log file. Arg1=1 ensures printing to the log file at every search step.

`DEMX`: This command is used to prevent saving of high-energy conformers during the search. Arg5 defines the allowed energy window above the currently found global minimum. New conformers that are not within arg5 kJ/mol will be discarded. Additionally, a preliminary energy test can be performed during the energy minimization, to ensure that a reasonable structure has been found. Arg2 sets the number of energy iterations to be performed before the preliminary test (a good value is approximately 1/3 of the total number of energy iterations), while arg6 defines the energy above which conformers will be discarded (a value of about 1.5–2 times arg5 is recommended).

`MSYM`: Invokes the numbering symmetry library `mmsym`, which automatically and more generally identifies a suitable numbering order for use in comparing molecular conformations.

`COMP`: Arg1-arg4 list the atom numbers of atoms to be used in structural comparison with all previously found conformers. A maximum of 200 atoms can be used in the comparisons. For arg1=0, all heavy atoms are compared. By default, structures that have equivalent atoms separated by more than 0.25 Å upon superposition are considered different.

`TORS`: Defines the variable torsions in the molecule. Arg1 and arg2 are the atom numbers of the two central atoms defining a variable torsion. Arg5 and arg6 define the minimum and maximum dihedral angle variations (in both directions).

`CONV`: Defines convergence criteria. Arg1=2 specifies derivative convergence (default criterion is 0.05 kJ/mol-Å, and this value is set in arg5).

`MINI`: Starts the minimization. Arg1 defines the type of minimization algorithm to be used. Arg1=9 means that Truncated Newton-Raphson Conjugate Gradient will be used. In arg3, the number of minimization steps is defined. Arg3 can be set to a large number since the calculation automatically stops as soon as the convergence criterion has been reached.

8.3.2 Multi-Structure Conformational Search Using LMOD

An example command file appears below for a conformational search calculation using low-mode conformational searching. Descriptions of the opcodes in the file follow. However, opcodes that are also in the MCMM command file example are not repeated. See the explanations in Section 8.3.1.

```
serial-lmcs.mae
serial-lmcs-out.mae
 FFLD      10      1      0      1     1.0000     0.0000     0.0000     0.0000
 BDCO       0      0      0      0    41.5692 99999.0000     0.0000     0.0000
 READ       0      0      0      0     0.0000     0.0000     0.0000     0.0000
 LMCS     100      0      0      0     0.0000     0.0000     3.0000     6.0000
 MCSS       2      0      0      0    50.0000     0.0000     0.0000     0.0000
 MCOP       1      0      0      1     0.0000     0.0000     0.0000     0.0000
 DEMX       0    166      0      0    50.0000   100.0000     0.0000     0.0000
 MSYM       0      0      0      0     0.0000     0.0000     0.0000     0.0000
 COMP       0      0      0      0     0.0000     0.0000     0.0000     0.0000
 CONV       2      0      0      0     0.0500     0.0000     0.0000     0.0000
 MINI       1      0    500      0     0.0000     0.0000     0.0000     0.0000
```

`LMCS`: Use the Low-Mode Conformational Search method. Arg1=100 means that 100 Monte Carlo steps will be carried out before the calculation stops.

`MCSS`: Select starting geometries for Monte-Carlo search steps. Arg1=2 tells MacroModel to use as starting geometries structures whose energies are allowed by arg2 and arg5, and are used the fewest times as starting structures.

`MCOP`: Arg1=1 prints search results every step. Arg4=1 signifies that this is a low-mode serial search. A separate conformational search will be performed on all molecules in the input file. Arg5=0 implies a pure low-mode conformational search, rather than a mixed-mode, as in a previous example.

`COMP`: Setting arg1=0 allows for all heavy atoms to be compared for each structure in the input file. This removes the need to use atom numbers for the individual structures.

8.3.3 Mixed MCMM/Low-Mode Search Using a Substructure File

This example uses the CDK2 structure 1e1v with the co-crystallized ligand cmg. Alternate binding modes of the ligand are sampled using a mixture of torsional moves, low-mode moves, and rotation and translation of the ligand, in a conformational search. This example was prepared with Maestro. A substructure is used, in which the freely moving region includes the ligand and any residues with atoms within 3.0 Å of the ligand. A fixed and frozen region was also set up in the **Substructure** tab. The conformational search parameters were initially set by using the **Perform Automatic Setup** button in the **CSearch** tab. The parameters were then modified using the individual parameter panels in the **CSearch** tab to arrive at the sample instruction file below. Only torsions in the ligand are being explicitly sampled in this example.

Descriptions of the opcodes in the file follow. However, opcodes in the previous command files in the chapter are not repeated. See the previous examples for those explanations.

```
subs-mcmm-lmod.mae
subs-mcmm-lmod-out.mae
 MMOD       0      1      0      0     0.0000     0.0000     0.0000     0.0000
 FFLD      11      1      0      0     1.0000     0.0000     0.0000     0.0000
 EXNB       0      0      0      0     0.0000     0.0000     0.0000     0.0000
 BDCO       0      0      0      0    89.4427 99999.0000     0.0000     0.0000
 SUBS       0      0      0      0     0.0000     0.0000     0.0000     0.0000
 READ       0      0      0      0     0.0000     0.0000     0.0000     0.0000
 CRMS       0      0      0      0    10.0000     0.5000     0.0000     0.0000
 LMCS      10      0      0      0     0.0000     0.0000     3.0000     6.0000
 MCNV       2      4      0      0     0.0000     0.0000     0.0000     0.0000
 MCSS       2      0      0      0    50.0000     0.0000     0.0000     0.0000
 MCOP       1      0      0      0     0.5000     0.0000     0.0000     0.0000
 DEMX       0    666      0      0    50.0000   100.0000     0.0000     0.0000
 COMP     117    121    128    133     0.0000     0.0000     0.0000     0.0000
 COMP     136    140    158    162     0.0000     0.0000     0.0000     0.0000
 COMP     165    169    174    176     0.0000     0.0000     0.0000     0.0000
 COMP     223    225    231    236     0.0000     0.0000     0.0000     0.0000
 COMP     439    445    466    470     0.0000     0.0000     0.0000     0.0000
COMP     473     476     482     487      0.0000      0.0000      0.0000      0.0000
COMP    1291    1296    1299    1303      0.0000      0.0000      0.0000      0.0000
COMP    1307    1309    1313    1318      0.0000      0.0000      0.0000      0.0000
COMP    1326    1362    1366    1370      0.0000      0.0000      0.0000      0.0000
COMP    1372    2105    2109    2115      0.0000      0.0000      0.0000      0.0000
COMP    2120    2155    2159    2164      0.0000      0.0000      0.0000      0.0000
COMP    2172    2323    2327    2331      0.0000      0.0000      0.0000      0.0000
COMP    2333    4599    4600    4601      0.0000      0.0000      0.0000      0.0000
COMP    4602    4603    4604    4605      0.0000      0.0000      0.0000      0.0000
COMP    4606    4607    4608    4609      0.0000      0.0000      0.0000      0.0000
COMP    4610    4611    4612    4613      0.0000      0.0000      0.0000      0.0000
COMP    4614    4615    4616       0      0.0000      0.0000      0.0000      0.0000
MSYM       0       0       0       0      0.0000      0.0000      0.0000      0.0000
CHIG     117     119     165     167      0.0000      0.0000      0.0000      0.0000
CHIG     223     439     468    1291      0.0000      0.0000      0.0000      0.0000
CHIG    1311    1364    2107    2157      0.0000      0.0000      0.0000      0.0000
CHIG    2325       0       0       0      0.0000      0.0000      0.0000      0.0000
TORS    4600    4610       0       0      0.0000    180.0000      0.0000      0.0000
TORS    4604    4605       0       0      0.0000    180.0000      0.0000      0.0000
TORS    4605    4606       0       0      0.0000    180.0000      0.0000      0.0000
TORS    4606    4611       0       0      0.0000    180.0000      0.0000      0.0000
MOLS    4633       0       0       0      0.0000    180.0000      0.0000      1.0000
TORC     135     134     132     133     90.0000    180.0000      0.0000      0.0000
TORC     164     163     161     162     90.0000    180.0000      0.0000      0.0000
TORC    1310    1309    1307    1308     90.0000    180.0000      0.0000      0.0000
TORC    2118    2117    2115    2116     90.0000    180.0000      0.0000      0.0000
CONV       2       0       0       0      0.1000      0.0000      0.0000      0.0000
MINI       9       0    1000       0      0.0000      0.0000      0.0000      0.0000
```

`SUBS`: With arg1=0 look for an `.sbc` file that contains information on how the substructure is set up in the system.

`MCOP`: Arg1=1 print search results every step. Arg5=0.5 specifies that this mixed Monte Carlo/ Low-Mode search will attempt Monte Carlo torsional and/or molecular translation moves half the time. The rest of the attempted moves will be generated using low-mode moves.

`MOLS`: Arg1-4 sepecifies one atom from each molecule to be moved using molecular translation/rotation moves. Arg5 and arg6 specify the minimum and maximum rotation angles while arg7 and arg8 specify the minimum and maximum translation distances.

8.3.4 Conformational Searches Using Multiple Minimization Methods (MWRT)

MWRT minimizations enable multiple minimizer methods to be applied to a structure during the minimization stage, which can enhance the performance of the minimization process. This method can be used in a conformational search. Examples are given in the examples directory,

`$SCHRODINGER/macromodel-v`*`version`*`/samples/Examples`

The example `mwrt_csearch` demonstrates an efficient small-molecule conformational search using MWRT. The example `mwrt_serialsearch` is a multi-structure example. In this second example, `AUTO` arg6=1 indicates a multi-structure conformation search. More information about MWRT is available in Section 6.2.3 on page 54.

8.4 Checking and Interpreting Results

Like other methods that involve minimizations, it is important to check that all of the minimizations for the conformers generated are converged. If some of the results are not converged, they can be minimized further using multiple minimization—see Chapter 7 for details.

You can also use Jaguar for further (quantum mechanical) refinement of the conformers. After you have optimized the structures with Jaguar, you should perform a redundant conformer elimination using the Jaguar energy as the source of energy, because some conformers that are considered separate by MacroModel might optimize to the same minimum in Jaguar. See Chapter 4 of the *Jaguar User Manual* for more information on Jaguar optimizations, and Chapter 15 of this manual for information on redundant conformer elimination. A procedure for a combined MacroModel/Jaguar conformational search is described in Section 7.6 of the *Jaguar User Manual*, and a tutorial is given in Chapter 4 of the *Jaguar Quick Start Guide*.

Often conformational searches yield large numbers of conformers. Clustering the conformers into families of similar structures can lead to useful insights. To do this, you can use the `conformer_cluster.py` script, which is installed by default in the distribution. There is a panel for setting up the job in Maestro which you open with Scripts → Cheminformatics → Clustering of Conformers.

Dynamics Calculations

Molecular dynamics simulations in MacroModel use classical mechanics (Newton's equations of motion) to mimic how the system would behave as a function of time, typically at or close to the temperature of interest. You can use molecular dynamics studies to learn about the thermal variations within a system or to permit it to relax out of a local minimum structure into related but more probable structures.

9.1 The Dynamics Panel

Using the Dynamics panel, you can set up dynamics calculations and either submit the calculation or write the job files for later use. The Dynamics panel consists of the upper, general portion that is common to other MacroModel panels. This section is discussed in Section 4.1 on page 31. The panel contains six tabs: Potential, Constraints, Substructure, Mini, Monitor, and Dynamics. The Monitor and Dynamics tabs are unique to the Dynamics panel. The controls in these tabs are explained below. For information on the other tabs, see Section 4.2 on page 32 through Section 4.5 on page 40.

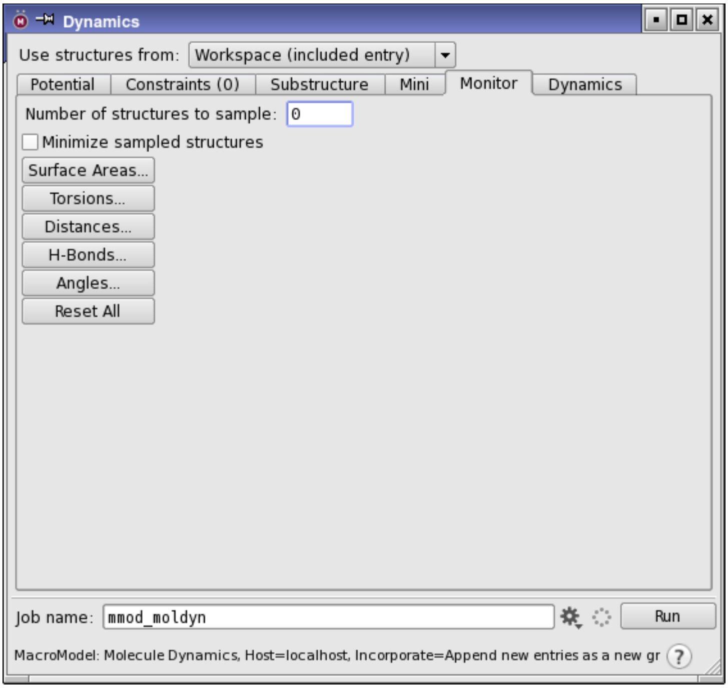
To open the Dynamics panel, select Dynamics from the MacroModel submenu of the Applications menu on the main menu bar.

9.2 Performing a Dynamics Calculation

To set up parameters for a dynamics calculation, first set values in the upper part of the Dynamics panel and in the Potential, Constraints, Substructure, and Mini tabs, then set values in the Monitor and Dynamics tabs, which are described below.

9.2.1 The Monitor Tab

During a molecular dynamics simulation, you can monitor a number of geometrical parameters. These are angles, inter-atomic distances, dihedral angles, the surface areas of individual atoms, and the population of hydrogen-bonds. The Monitor tab of the Dynamics panel contains a series of buttons that open panels that allow you to define these parameters. The results of the monitoring appear in the *jobname*`.mmo` file. Information is also recorded in the *jobname*`.log` and the *jobname*`-out.mae` files. The latter is used to convey the property values to Maestro's project table.



Use structures from: Workspace (included entry)

Number of structures to sample: 0

☐ Minimize sampled structures

Job name: mmod_moldyn

MacroModel: Molecule Dynamics, Host=localhost, Incorporate=Append new entries as a new gr

Figure 9.1. The Monitor tab of the Dynamics panel.

You can also sample structures during a simulation. To do this, simply enter the total number of structures you want to sample in the Number of structures to sample: text box. This number of structures is selected at regular intervals throughout the simulation and written to the output (*jobname*`-out.mae`) file.

If you want to minimize the sampled structures, select Minimize sampled structures. This option is incompatible with the monitoring of structural parameters, so the buttons are not available when this option is selected.

Five of the buttons in the Monitor tab open panels that allow you to choose what to monitor. Each of these panels is described in a section below. Each panel has a text area in which the atoms that define each geometrical parameter are listed, an atom selection tool, and Delete and Delete All buttons to delete one or all items from the list.

You can redefine an item by selecting it and then picking atoms in the Workspace. When parameters are defined, the atoms defining them are marked in the Workspace if Markers is selected. When you close the panel, the markers are automatically cleared.

The remaining button, Reset All, removes all entries from all dynamics monitors.

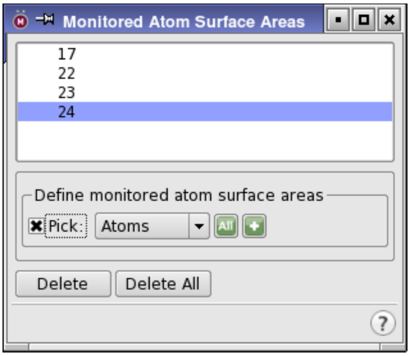

Monitored Atom Surface Areas
17
22
23
24
Define monitored atom surface areas
☒ Pick:
Atoms
All
Delete
Delete All
?

Figure 9.2. The Monitored Atoms Surface Area panel.

9.2.1.1 Surface Areas

Use the Monitored Atom Surface Areas panel to specify the atoms whose surface areas are to be monitored during the simulation. To monitor the surface area of specific atoms, use the selection tool to pick atoms. The selection tool allows you to specify atoms in the following ways:

- Choose a structural unit from a Pick menu and pick atoms in the Workspace belonging to structural units of the selected type.
- Choose all atoms by clicking the All button.
- Choose atoms using the Atom Selection dialog box, which creates ASL expressions that define the selected atoms. Open the Atom Selection dialog box by clicking the Atom Selection button and choosing Select.

➕

After each atom is picked, a new entry appears in the list of atoms in the Monitored Atoms List located at the top of the panel. When an atom is picked, Maestro marks it and places an eye icon beside it. The selected atom is colored teal, the other monitored atoms are colored green.

9.2.1.2 Distances

Use the Monitored Distances panel to specify interatomic distances for monitoring. To define a distance to be monitored, choose Atom or Bond from the Pick menu, and click on two atoms or one bond in the Workspace. When the distance is defined, the two atoms appear in the list at the top of the panel. The defined distances are marked with a purple dotted line and eye icon. The selected distance is distinguished by a thicker line.

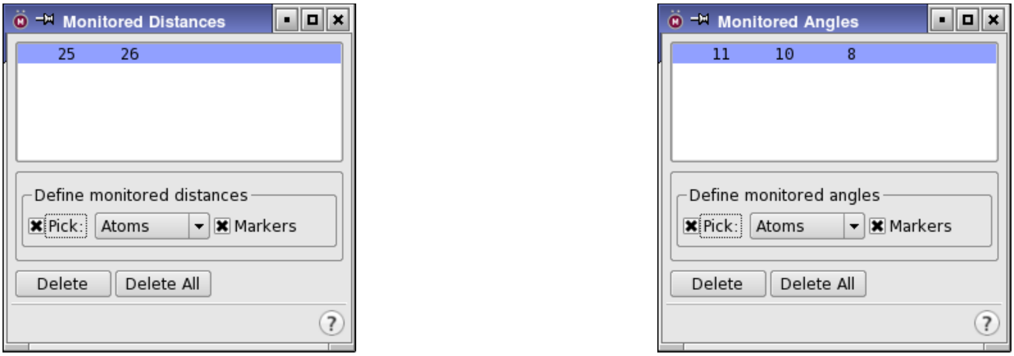

Monitored Distances
25 26
Define monitored distances
☒ Pick: Atoms
☒ Markers
Delete
Delete All
?
Monitored Angles
11 10 8
Define monitored angles
☒ Pick: Atoms
☒ Markers
Delete
Delete All
?

Figure 9.3. The Monitored Distances panel (left) and the Monitored Angles panel (right).

9.2.1.3 Angles

Use the Monitored Angles panel to specify angles for monitoring. To define an angle to be monitored, choose Atom or Bond from the Pick menu, and click on three atoms or two bonds in the Workspace that define an angle. When the angle is defined, the three atoms appear in the list at the top of the panel. The defined angles are marked with a green solid line, a dotted line through the angle, and an eye icon. The selected angle is distinguished by a thicker line.

9.2.1.4 Torsions

Use the Monitored Torsions panel to select dihedral angles for monitoring. To define a dihedral angle to be monitored, choose Atom or Bond from the Pick menu, and click on four atoms or three bonds in the Workspace that define a dihedral angle. When the dihedral angle is defined, the four atoms appear in the list at the top of the panel. The defined dihedral angles are marked with a red solid line, a dotted line through the dihedral angle, and an eye icon. The selected angle has a thicker line.

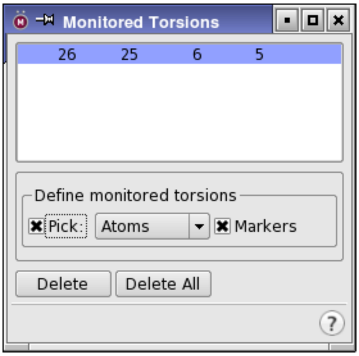
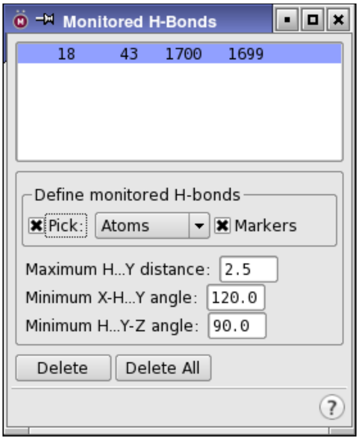

Monitored Torsions
26 25 6 5
Define monitored torsions
☒ Pick: Atoms
☒ Markers
Delete
Delete All
?

Figure 9.4. The Monitored Torsions panel.



18 43 1700 1699

Define monitored H-bonds

☒ Pick: Atoms ☒ Markers

Maximum H...Y distance: 2.5

Minimum X-H...Y angle: 120.0

Minimum H...Y-Z angle: 90.0

Figure 9.5. The Monitored H-Bonds panel.

9.2.1.5 H-Bonds

During a dynamics simulation, MacroModel periodically examines the geometry around each monitored H-bond. If the bond meets the three user-specified H-bond criteria, it is counted in the H-bond population survey.

To select H-bonds from the structure in the Workspace, click the four atoms that define the bond and the associated angles and distances. The picking order of the atoms is important. Start with the heavy atom of the donor pair (designated X), followed by the hydrogen atom (designated H), the acceptor (designated Y), and an atom attached to the acceptor (designated Z) to define an angle.

For example, in a typical N–H to O=C hydrogen bond, the picking order should be N, H, O, then C. After the fourth atom has been selected, a new entry appears in list at the top of the panel. The H-bond is marked with solid yellow lines between X and H and between Y and Z, and a dotted yellow line between H and Y. The currently selected H-bond has solid lines on either side of the dotted line.

You can specify the monitoring criteria for the selected list entry by entering a maximum H...Y distance, a minimum X–H...Y angle, and a minimum H...Y–Z angle in the appropriate text boxes. If you do not specify values, the default values are used. The default value for the H...Y distance is 2.5 Å, the minimum X-H...Y angle has a default value of 120°, and the minimum H...Y-Z angle is 90°. If the distance is greater than the maximum or either of the angles is smaller than the minimum, the H-bond is not counted in the current population survey.

9.2.1.6 Monitoring Output

The values associated with the monitored output of structures, energies, and temperatures from a dynamics calculation correspond to

- The instantaneous value.
- The block average since the last monitored structure.
- The average of all steps since the start of the simulation.

The distance output also includes, for distance r,

- Block average of $1/r^6$.
- The average over all steps of $1/r^6$.

9.2.2 The Dynamics Tab

The Dynamics tab of the Dynamics panel contains settings for defining the dynamics calculation method and the calculation conditions.

The controls in the Dynamics tab are described below.

Method

This menu contains two options:

- Stochastic dynamics (default): The most common method because it includes random forces that simulate the buffeting of a system by solvent molecules. In combination with damping forces, the random forces provide a simple, robust method for controlling temperature in a simulation. The random forces can also assist in sampling the potential energy surface. See Reference 27 for more information.

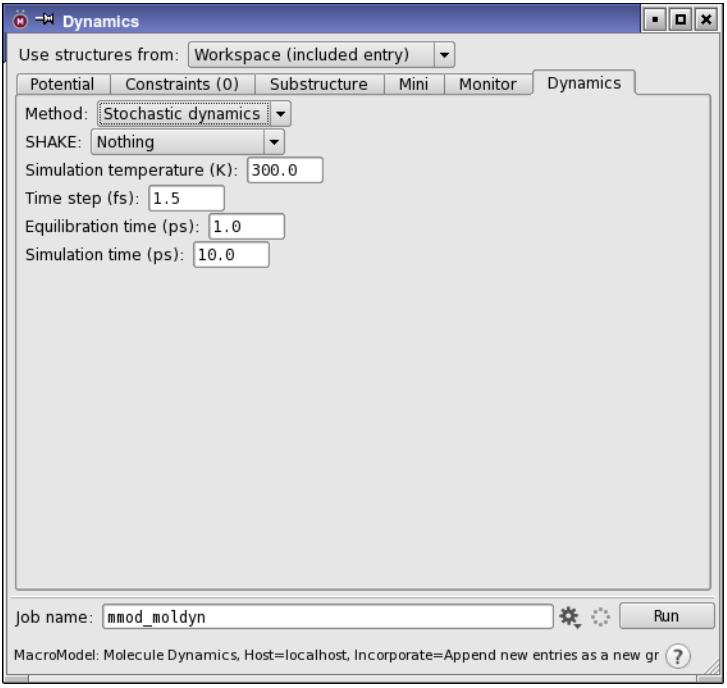
- Molecular dynamics: Uses a standard constant temperature velocity-Verlet algorithm.

SHAKE

The SHAKE procedure [28] constrains selected bond lengths to their original values. This procedure allows the use of larger time steps than unconstrained simulations. For most simulations, the Bonds to Hydrogens choice from the SHAKE option menu should be sufficient. This choice allows time steps up to 2 fs.

Simulation temperature (K)

The temperature at which a simulation is carried out. The default value is 300.0 K.



Use structures from: Workspace (included entry)

Method: Stochastic dynamics

SHAKE: Nothing

Simulation temperature (K): 300.0

Time step (fs): 1.5

Equilibration time (ps): 1.0

Simulation time (ps): 10.0

Job name: mmod_moldyn

MacroModel: Molecule Dynamics, Host=localhost, Incorporate=Append new entries as a new gr

Figure 9.6. The Dynamics tab of the Dynamics panel.

Time step (fs)

Determines the time step used in the integration of the equations of motion during the simulation. Smaller values lead to more accurate but computationally intensive simulations for a given amount of time simulated. The default value is 1.5 fs.

Equilibration time (ps)

Determines the length of the “settling down” period at the start of the simulation. The equilibration period is needed to allow initial velocities to stabilize before any monitoring data or sampled structures are written to output files.

Simulation time (ps)

The total time allowed for a given simulation. The default is 10 ps. However, if you want converged results, specify a much longer time, even for small to medium-sized systems.

9.3 Command File Examples

To run MacroModel calculations, a molecular structure file and a command file are required. The molecular structure file contains the structures to be used as input in the calculation. The command file contains the name of the input structure file, the name of the output structure file, and an ordered list of operation codes (*opcodes)* for the calculations.

When you start a job from Maestro or write out the input files, Maestro writes out a structure file and a command file. For many jobs, command files written this way are complete and adequate, but for some jobs you may need to adjust the Maestro-generated command file.

The command files and the log files for the examples given in this section can be found in `$SCHRODINGER/macromodel-v`*version*`/samples/Examples`.

9.3.1 Stochastic Dynamics

Below is an example of the command file for a stochastic dynamics simulation and explanations of the opcodes that appear in the file.

```
sdyn.mae
sdyn-out.mae
 MMOD       0      1      0      0      0.0000      0.0000      0.0000      0.0000
 FFLD      10      1      0      1      1.0000      0.0000      0.0000      0.0000
 BDCO       0      0      0      0     41.5692  99999.0000      0.0000      0.0000
 READ       0      0      0      0      0.0000      0.0000      0.0000      0.0000
 CONV       2      0      0      0      0.0500      0.0000      0.0000      0.0000
 MINI       9      0    500      0      0.0000      0.0000      0.0000      0.0000
 MDIT       0      0      0      0    300.0000      0.0000      0.0000      0.0000
 MDYN       0      1      1      0      1.5000      1.0000    300.0000      0.0000
 MDSA      10      0      0      0      0.0000      0.0000      1.0000      0.0000
 MDDA      14     15     16     17      0.0000      0.0000      0.0000      0.0000
 MDYN       1      1      1      0      1.5000     10.0000    300.0000      0.0000
 WRIT       0      0      0      0      0.0000      0.0000      0.0000      0.0000
```

`MMOD`: Creates and updates an intermediate structure file so that structures can be displayed in Maestro as the job progresses.

`FFLD`: Force field selection. Arg1 denotes the actual force field used in the calculation (in this case MMFF94). Arg2 defines the electrostatic treatment for the calculation. The default (arg=0) is to use the dielectric treatment encoded in the force field, however, in this case a constant dielectric is used. Arg4 is MMFF94-specific. Arg4=1 defines the MMFF94s version of the force field, ensuring planarity around delocalized sp^2 nitrogens.

`BDCO`: Use the Bond Dipole CutOff (BDCO) method for truncating electrostatic interactions. Arg5 and arg6 are used to specify the cutoffs used for charge-dipole and charge-charge interactions, respectively.

`READ`: Read the input file.

`CONV`: Defines convergence criteria. Arg1=2 signifies derivative convergence (default, if no `CONV` command is present, criterion is 0.05 kJ/mol-Å; this value is set in arg5).

`MINI`: An energy minimization precedes the dynamics calculation in order to eliminate excess potential energy. Arg1=9 indicates that the TNCG minimization method should be used for arg3 minization iterations.

`MDIT`: Apply random initial velocities corresponding to 300K (arg5) to all atoms.

`MDYN`: Perform the dynamics simulation. Arg2=1 selects the use of the SHAKE protocol to constrain hydrogen bonds to their natural values. Arg3=1 sets up a stochastic dynamics run. The time step (in fs) is set in arg5, the length of the simulation (in ps) in arg6, and the temperature (in K) of the simulation in arg7.

Two `MDYN` command lines appear in this command file. The first performs a short equilibration run, while the second is where the actual sampling occurs. Arg1 defines printing of energy listing. In the first `MDYN` line, arg1=0, meaning that a summary of the energies is printed to the log file only. In the second `MDYN` line, arg1=1, signifying that energies as well as monitoring results are written to the `.mmo` file.

`MDSA`: Perform structure sampling during the stochastic dynamics simulation. Intermediate structures generated in the dynamics simulation are saved to the output structure file at regular intervals during the calculation. Arg1 defines the total number of structures to sample (number of "snapshots" to take) during the simulation. Structures can also be written out based on time-intervals (in ps) using arg5. Arg7=1 forces deletion of all structures stored in the `-out.mae` file prior to each simulation.

`MDDA`: Monitor a dihedral angle during the simulation. Arg1-arg4 defines the atom numbers of the angle to be monitored. The default arg5 divides the reported results into 10-degree increments. Average values are reported in the `.log` file while more detailed information (including the distribution of angles) is reported in the `.mmo` file.

`WRIT`: Write the final structure to the output file.

9.3.2 Simulated Annealing

Simulated annealing is often used to relax structures into a lower energy state a dynamics technique by systematically lowering the temperature used in the simulations. Below is an example command file for a simulated annealing calculation, where the target temperature for the system is changed in a step-wise manner. The temperature can also be changed in a continuous manner. See the description of the `MDFT` opcode in the *MacroModel Reference Manual* for more information.

```
sim-ann.mae
sim-ann-out.mae
 FFLD      10      1      0      1    1.0000      0.0000      0.0000      0.0000
 BDCO       0      0      0      0   41.5692 99999.0000      0.0000      0.0000
 READ       0      0      0      0    0.0000      0.0000      0.0000      0.0000
 CONV       2      0      0      0    0.0500      0.0000      0.0000      0.0000
 MINI       9      0    500      0    0.0000      0.0000      0.0000      0.0000
 MDIT       0      0      0      0  300.0000      0.0000      0.0000      0.0000
 MDYN       0      1      1      0    1.5000     10.0000    300.0000      0.0000
 MDYN       0      1      1      0    1.5000     20.0000    150.0000      5.0000
 MDYN       0      1      1      0    2.0000     20.0000     50.0000      5.0000
 MINI       9      0    500      0    0.0000      0.0000      0.0000      0.0000
```

`MDYN`: Note that the three separate `MDYN` command lines perform different tasks. First, a 10 ps equilibrium run at 300 K (initialized by the `MDIT` line) is performed, followed by a 20 ps simulation coupled to a thermal bath of 150 K. Arg8 gives the time constant of the bath (in ps). This second `MDYN` command slowly cools the system to 150 K, while the last `MDYN` command cools the system to 50 K.

A `WRIT` command is not necessary because the final `MINI` command ensures that the minimized structure is written to the output file.

See Section 9.3.1 on page 104 for a description of the other opcodes in this file.

9.4 Checking and Interpreting Results

It is often difficult to know when you have sampled enough in a molecular dynamics simulation. It usually is helpful to have some prior knowledge of the time-scales for crucial processes within the system to know if convergence may have been achieved.

While molecular dynamics simulations provide a wealth of structural and temporal information, the information is often hard to interpret. If the goal is to thermally sample the local conformational minimum of a small molecule, then you might not need to examine the results carefully provided that you have simulated sufficiently long (50 to 100 ps may be enough). However, if the system is large and the conformational variation is large, simulations likely will not adequately sample the conformations available, and careful, problem-specific consideration of the results may be needed to learn from such studies. In such circumstances, it almost always helps to examine the trajectory visually with a tool such as the ePlayer. In addition, clustering tools may help you identify when key events occurred during the simulation.

MC/SD Calculations

Monte Carlo/Stochastic Dynamics (MC/SD) performs constant temperature calculations that take advantage of the strengths of Monte Carlo methods for quickly introducing large changes in a few degrees of freedom, and stochastic dynamics for its effective local sampling of collective motions. MC/SD is a good general choice for studying a system at constant temperature.

10.1 The MC/SD Panel

You can prepare, write job files for, and submit a Monte Carlo/Stochastic Dynamics calculation from the Maestro MC/SD panel. The MC/SD panel has a general setting section and a Potential tab, a Constraints tab, and a Substructure tab like other MacroModel energy panels. These portions of the panels are described in detail in Section 4.1 on page 31 through Section 4.5 on page 40. The MC/SD panel also has Monitor and Dynamics tabs. These tabs also appear on the Dynamics panel, and are discussed in Section 9.2 on page 97. The MCSD tab is unique to the MC/SD panel. This tab contains reaction condition settings.

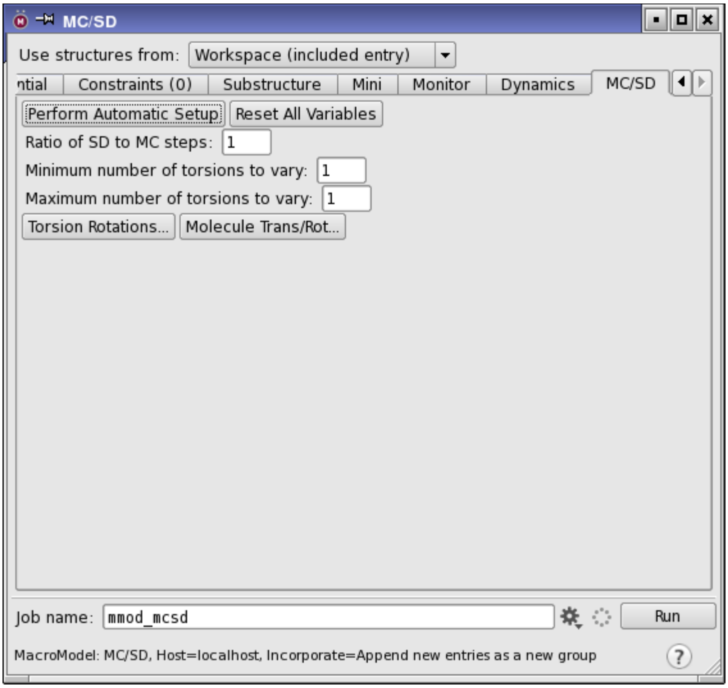
To open the MC/SD panel, choose MC/SD from the MacroModel submenu of the Applications menu in the main menu bar.

10.2 Setting Up MC/SD Calculations

The MC/SD procedure [29] differs from a normal dynamics simulation in that it uses a mixture of Metropolis Monte Carlo and dynamics steps in order to greatly increase the rate at which a simulation explores conformational space. For MC/SD simulations, you must specify torsions to be rotated and, if there is more than one molecule in the system, molecules to be translated and rotated.

The controls for MC/SD simulations are in the MCSD tab, with the exception of the temperature setting, which is in the Dynamics tab.

To define the torsions to be rotated and the molecule translation and rotation selections automatically, click the Perform Automatic Setup button. To view and edit the automatically generated settings, or to specify the settings manually, open the Torsion Rotations and Molecule Trans/Rot panels by clicking the corresponding buttons. These panels are the same as for conformational searches. For a more detailed description of these panels, see Section 8.2.5.2 on page 84 and Section 8.2.5.3 on page 85.


MC/SD
ntial
Constraints (0)
Substructure
Mini
Monitor
Dynamics
MC/SD
Perform Automatic Setup
Reset All Variables
Torsion Rotations...
Molecule Trans/Rot...
Run
?

Use structures from: Workspace (included entry)

Ratio of SD to MC steps: 1

Minimum number of torsions to vary: 1

Maximum number of torsions to vary: 1

Job name: mmod_mcsd

MacroModel: MC/SD, Host=localhost, Incorporate=Append new entries as a new group

Figure 10.1. The MC/SD tab of the MC/SD panel.

The value entered in the Ratio of SD to MC steps text box determines how many Monte Carlo trials will be performed for each dynamics time step. The default value is 1, but the number can be increased to give more dynamics steps per Monte Carlo step.

You can specify the minimum and maximum number of torsions to vary in an MC step in the corresponding text boxes. The number of torsions varied is randomly selected from the range defined by the minimum and maximum values. The default is to vary only one torsion.

We recommend against the use of SHAKE during MC/SD simulations, as it can result in less than optimal temperature control.

10.3 File Examples

To run MacroModel calculations, a molecular structure file and a command file are required. The molecular structure file contains the structures to be used as input in the calculation. The command file contains the name of the input structure file, the name of the output structure file, and an ordered list of operation codes (*opcodes)* for the calculations.

When you start a job from Maestro or write out the input files, Maestro writes out a molecular structure file and a command file. For many types of jobs, command files written this way are complete and adequate, but for some types of jobs you may need to adjust the Maestro-generated command file.

This section contains an example of an MC/SD computation using a small organic molecule. The command files and the log files for the examples given in this section can be found in `$SCHRODINGER/macromodel-v`*version*`/samples/Examples`.

```
mcsd.mae
mcsd-out.mae
 MMOD      0      1      0      0     0.0000     0.0000     0.0000     0.0000
 FFLD     10      1      0      1     1.0000     0.0000     0.0000     0.0000
 BDCO      0      0      0      0    41.5692 99999.0000     0.0000     0.0000
 READ      0      0      0      0     0.0000     0.0000     0.0000     0.0000
 CONV      2      0      0      0     0.0500     0.0000     0.0000     0.0000
 MINI      1      0    500      0     0.0000     0.0000     0.0000     0.0000
 MCNV      1      3      0      0     0.0000     0.0000     0.0000     0.0000
 TORS      1     14      0      0     0.0000   180.0000     0.0000     0.0000
 TORS      6     26      0      0     0.0000   180.0000     0.0000     0.0000
 TORS     14     15      0      0     0.0000   180.0000     0.0000     0.0000
 TORS     15     16      0      0     0.0000   180.0000     0.0000     0.0000
 TORS     16     17      0      0     0.0000   180.0000     0.0000     0.0000
 TORS     17     18      0      0     0.0000   180.0000     0.0000     0.0000
 TORS     26     27      0      0     0.0000   180.0000     0.0000     0.0000
 MCSD      1      0      0      0     0.0000     0.0000   300.0000     0.0000
 MDIT      0      0      0      0   300.0000     0.0000     0.0000     0.0000
 MDYN      0      0      1      0     1.5000     1.0000   300.0000     0.0000
 MDSA     20      0      0      0     0.0000     0.0000     1.0000     0.0000
 MDYN      1      0      1      0     1.5000    10.0000   300.0000     0.0000
 WRIT      0      0      0      0     0.0000     0.0000     0.0000     0.0000
```

Several opcodes have been discussed in previous chapters. Only the opcodes relevant to the MC/SD procedure are described below.

`MINI`: Minimize the structure before the dynamics simulation.

`MCNV`: The number of torsion angles to vary during the MC portion of the simulation. Here, between one and three torsion variations are specified.

`TORS`: Specify the torsion angles to vary by atom number.

`MCSD`: Specify MC/SD sampling at 300 K with an MC to SD step ratio of 1:1.

`MDIT`: Specify dynamics simulation at a temperature of 300 K.

`MDYN`: Perform equilibration dynamics run for 1 ps.

`MDSA`: Sample 20 structures during the following full dynamics run.

`MDYN`: Perform full MC/SD dynamics run for 10 ps at 1.5 fs intervals.

10.4 Checking and Interpreting Results

As with normal dynamics simulations, it is often difficult to know when you have sampled enough in a MC/SD simulation. It is usually helpful to have some prior knowledge of the time scales for crucial processes within the system to know if convergence may have been achieved. While MC/SD simulations provide a wealth of structural and temporal information, this information is often hard to interpret. If the goal is to thermally sample the local conformational minimum of a small molecule, then you may not need to examine the results carefully provided that you have simulated sufficiently long (50 to 100 ps may be enough). However, if the system is large and the conformational variation is large, simulations likely will not adequately sample the conformations available, and careful, problem-specific consideration of the results may be needed to learn from such studies. In such circumstances, it almost always helps to examine the monitored trajectory with a tool such as Maestro's ePlayer. In addition, clustering tools such as XCluster may help identify when key events occurred during the simulation.

If the MC/SD simulation is unstable and fails, consider turning off long-range constant derivatives, by setting `EXNB` arg1 to 1, and increasing the non-bonded cutoffs to a large number, such as 100 Å, by choosing User defined from the Cutoff option menu in the Potential tab and entering values in the text boxes.

Minta Calculations

MINTA can be used for fast computation of the conformational free energy of small- and medium-sized molecular systems *in vacuo* and in the presence of a continuum solvent model. MINTA is an excellent tool for calculating the binding free energy of molecular complexes composed of substrate molecules bound to small receptors used in the molecular recognition field or enzyme receptor models used in pharmaceutical research. Unlike available free energy simulation programs, MINTA calculations are user-friendly and are a simple tool for medicinal chemists familiar with conformational analysis. For more information on MINTA calculations, see Appendix H of the *MacroModel Reference Manual*.

11.1 The MINTA Panel

You can run the MINTA program from the MINTA panel using the Workspace contents, project table entries, or structures in a separate file as input. The MINTA panel has a general setting portion, which is described in Section 4.1 on page 31. The panel also contains four tabs: Potential, Substructure, Mini, and MINTA. For information on settings in the Potential tab, see Section 4.2 on page 32. For substructure-related material, see Section 4.5 on page 40. For the Mini tab description, see Section on page 51.

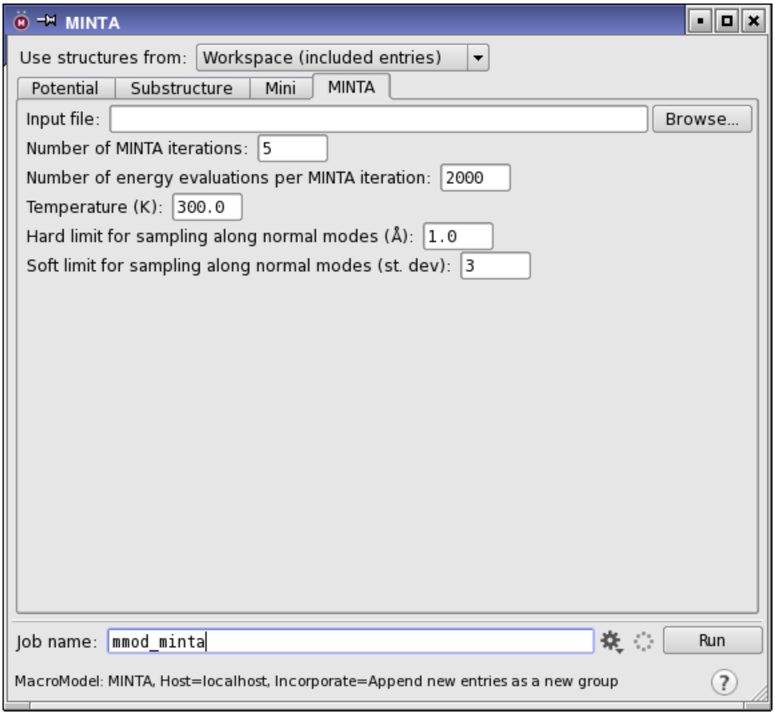
To open the MINTA panel, choose MINTA from the MacroModel submenu of the Applications menu in the main menu bar. The controls in the MINTA tab are described below.

Input File

Enter the name of the input file in the text box, or click Browse to navigate to the file. This must be a Maestro-formatted file that contains one or more valid conformations. These structures are usually the results of a previous conformational search. The Browse button opens a file selector for locating the desired input file. If no file is entered, the structural input is taken from the source chosen from the Use structures from option menu.

Number of MINTA iterations

The MINTA numerical integrals are calculated in statistical blocks to achieve better convergence. The number of blocks used is referred to as the number of MINTA iterations. The default value is 5, and the minimum value is 1.



Use structures from: Workspace (included entries)

Input file:

Number of MINTA iterations: 5

Number of energy evaluations per MINTA iteration: 2000

Temperature (K): 300.0

Hard limit for sampling along normal modes (Å): 1.0

Soft limit for sampling along normal modes (st. dev): 3

Job name: mmod_minta

MacroModel: MINTA, Host=localhost, Incorporate=Append new entries as a new group

Figure 11.1. The MINTA tab of the MINTA panel.

Number of energy evaluations per MINTA iteration

MINTA integration is based on single point energy evaluations. The default number of energy evaluations per MINTA iteration is 2000, and the minimum value is 1.

Temperature (K)

The value entered in this text box sets the temperature for the MINTA calculations. The default value is 300 K, and the minimum value is 0 K.

Hard limit for sampling along normal modes (Å)

This value determines a distance to which sampling is limited from the equilibrium geometry of the structure along any of the normal mode directions in 3*N*-6(5) dimensional normal mode space, where *N* is the number of atoms. The default value is 1.0 Å, the minimum value is 0.0 Å, and the maximum value is 3.0 Å.

Soft limit for sampling along normal modes (st. dev)

This value determines the units of standard deviation for sampling along normal modes. Sampling is limited to different distances from the equilibrium geometry along different normal mode directions. For a particular mode *i* sampling is limited to a particular distance equal to this value times the standard deviation of the multidimensional Gaussian function, along the particular normal mode direction *i*.

11.2 Command File Examples

To run MacroModel calculations, a molecular structure file and a command file are required. The molecular structure file contains the structures to be used as input in the calculation. The command file contains the name of the input structure file, the name of the output structure file, and an ordered list of operation codes (*opcodes)* for the calculations.

When you start a job from Maestro or write out the input files, Maestro writes out a molecular structure file and a command file. For many types of jobs, command files written this way are complete and adequate, but for some types of jobs you may need to adjust the Maestro-generated command file.

The command file and the log file for the example given in this section can be found in `$SCHRODINGER/macromodel-v`*version*`/samples/Examples`.

```
MintaMae.mae
MintaMae-out.mae
 MMOD       0      1      0      0     0.0000     0.0000     0.0000     0.0000
 FFLD      10      1      0      1     1.0000     0.0000     0.0000     0.0000
 BDCO       0      0      0      0    41.5692 99999.0000     0.0000     0.0000
 BGIN       0      0      0      0     0.0000     0.0000     0.0000     0.0000
 READ       0      0      0      0     0.0000     0.0000     0.0000     0.0000
 CONV       2      0      0      0     0.0500     0.0000     0.0000     0.0000
 MINI       9      0    500      0     0.0000     0.0000     0.0000     0.0000
 MNTA       5   2000      0      0   300.0000     1.0000     3.0000     0.0000
 END        0      0      0      0     0.0000     0.0000     0.0000     0.0000
```

`MMOD`: Creates and updates an intermediate structure file so that structures can be displayed in Maestro as the job progresses.

`FFLD`: Force field selection. Arg1 denotes the actual force field used in the calculation (in this case MMFF94). Arg2 defines the electrostatic treatment for the calculation. The default (arg2=0) is to use the dielectric treatment encoded in the force field. However, in this case a constant dielectric is used. Arg4 is MMFF94-specific. Arg4=1 defines the MMFF94s version of the force field, ensuring planarity around delocalized sp2 nitrogens.

`BDCO`: Use the Bond Dipole CutOff (`BDCO`) method for truncating electrostatic interactions. Arg5 and arg6 are used to specify the cutoffs used for charge-dipole and charge-charge interactions, respectively.

`BGIN/END`: The loop reads structures from a preceding conformational search. The input file should contain only conformers of the same molecule or molecular complex.

`READ`: Read the input file.

`CONV`: Defines convergence criteria. Arg1=2 signifies derivative convergence (default criterion, if no `CONV` command is present, is 0.05kJ/mol-Å. This value is set in arg5).

`MINI`: Starts the minimization. Arg1 defines the type of minimization algorithm to be used. Arg1=9 means that Truncated Newton-Raphson Conjugate Gradient will be used. In arg3, the number of minimization steps is defined. Arg3 can be set to a large number since the calculation will automatically stop as soon as the convergence criterion is reached.

`MNTA`: A Minta free energy calculation will be performed. Minta numerical integrations are performed in blocks in order to achieve better convergence, and arg1 defines the number of such blocks. Arg2 gives the number of energy evaluations per block, hence the total number of energy evaluations per structure is arg1*arg2. Minta can be run in an adaptive manner, which is slightly more accurate than the non-adaptive mode. Arg3=0 specifies the use of non-adaptive mode integration. Set arg3 not equal to 0 and arg4 to values from 1 to 10 for adaptive integration. Arg4 defines the number of “soft” (or “low-frequency” vibrational modes) degrees of freedom for which numerical integration should be applied. The default is to apply numerical integration to all degrees of freedom. However, this is recommended only for very small molecules (about 20 atoms or fewer). It is strongly recommended not to use values of arg4 greater than 50. Arg5 sets the simulation temperature, while arg6 and arg7 define hard and soft limits for sampling along normal modes.

11.3 Checking and Interpreting Results

For many MINTA calculations, the input set of conformations must be generated in a separate calculation such as a conformational search. The set of conformations needed for accurate MINTA calculations should be fairly extensive and include all low-lying conformers.

Protein Loop Construction

Before analyzing how a given protein-ligand pair will interact, you may want to refine the protein structure. This commonly involves examining the structure of loops—short sequences of amino acids which typically occur on the surface of the protein, but in the middle of the protein sequence.

The reasons for refining a protein structure include:

- The structure of this portion of the protein was not well-resolved.
- The conformation of the loop changed upon solvation.

In addition, when working with enzymes, you may want to examine how the structure of the loop changes as you change the sequence in the loop.

The MacroModel LOOP tool is a conformational search method for protein loops. It generates a variety of structures that are both diverse and geometrically appropriate for the protein loop, and minimizes the energy of these structures. LOOP functions in much the same manner as MCMM, and is used in conjunction with the `MINI`, `MCOP`, `MSYM`, and `LPOP` opcodes. LOOP is best used in conjunction with other methods, like LLMOD, to sample the loop structures more thoroughly. For more information on LOOP, see the *MacroModel Reference Manual*.

12.1 Performing LOOP Calculations

When performing a LOOP calculation, you can indicate the loop sequence you want to examine in one of two ways:

- Indicate that you want to use the loop sequence is that in the original protein.
- Create an auxiliary *filename*`.lsq` file, which lists the desired loop sequence.

12.1.1 Input Restrictions

For the current implementation of LOOP, input must meet the following criteria:

- Only one loop can be examined at a time, although successive calculations may focus on different loops.

- The protein loop and the residues it is immediately attached to must consist of amino acids containing the backbone atoms Np-Cα-Cp, where Np and Cp are nitrogen and carbon atoms involved in peptide bonds.

- Each loop must contain at least four residues.
- An all-atom representation of the protein must be used.
- No disulfide bonds are permitted within the loop or between the loop and the rest of the protein.
- No atoms in the loop can be frozen.
- LOOP will not work from a collection of input protein structures. It will work only from one such structure.

12.1.2 Structure Handling Considerations

When using LOOP, you should be aware of the following:

- Rings, such as the one in proline, are treated as rigid.
- If comparison atoms (`COMP`) are specified explicitly or implicitly (see arg3), then `MSYM` must be used.
- All LOOP runs renumber the atoms within the structure so that the loop atoms are the highest numbered atoms in the structure. During a run, MacroModel tracks the shifted atoms numbers. LOOP automatically produces a substructure file, *filename*`-out.sbc`, containing the shifted atom numbers for use in follow-up studies.

12.2 Example .lsq Input File

Consider the following contents of an example auxiliary file, *filename*`.lsq`.

```
GLY
ALA
SER
THR
ARG
GLU
SER
```

Based on the file contents, LOOP creates a new loop with the seven specified amino acids, starting with GLY at the N-terminus end of the loop.

The format and contents of the file meet the following criteria:

- Only alpha amino acids are used.
- The amino acids used all appear in the Maestro fragment tables, found in the Maestro Build panel.

- Standard three-letter abbreviations are used.
- Nonstandard (D) alpha amino acids from the Maestro fragment tables can also be used, but these require the four-letter abbreviation, e.g., DALA.

12.3 Command File Examples

To run MacroModel calculations, a molecular structure file and a command file are required. The molecular structure file contains the structures to be used as input in the calculation. The command file contains the name of the input structure file, the name of the output structure file, and an ordered list of operation codes (*opcodes)* for the calculations.

This section includes example command files for using LOOP, when using a LOOP sequence from an input protein structure and using the LOOP sequence from a file. The command files and the log files for the examples given in this section can be found in `$SCHRODINGER/macromodel-v`*version*`/samples/Examples`.

12.3.1 LOOP Job Using the Input Structure Sequence

An example command file for a LOOP sequence from an input protein structure appears below. Descriptions of the opcodes used in the file follow.

```
loop_input.mae
loop_input-out.mae
 SOLV      3      1      0      0     0.0000     0.0000     0.0000     0.0000
 FFLD     11      0      0      0     0.0000     0.0000     0.0000     0.0000
 EXNB      0      0      0      0     0.0000     0.0000     0.0000     0.0000
 SUBS      0      0      0      0     0.0000     0.0000     0.0000     0.0000
 READ      0      0      0      0     0.0000     0.0000     0.0000     0.0000
 LOOP    423    467      0      4     0.0        0.26       0.0000     0.0000
 DEMX      0      0      0      0  1000.0        0.0000     0.0000     0.0000
 MSYM      0      1      0      0     0.0000     0.0000     0.0000     0.0000
 MCOP      1      0      0      0     0.0000     0.0000     0.0000     0.0000
 CONV      2      1      0      0     0.0000     0.0000     0.0000     0.0000
 MINI      9      0   2000      0     0.0000     0.0000     0.0000     0.0000
```

`SOLV`: The GB/SA effective water model is being used.

`FFLD`: This example uses OPLS_2001 with constant dielectric electrostatics, which is appropriate if GB/SA solvation is being used.

`EXNB`: Extended non-bonded cut-offs should be used with GB/SA solvation.

`SUBS`: Read in a substructure from an `.sbc` file. Currently, best results are obtained if all the non-loop atoms in the system are frozen. A shell containing complete residues for all atoms within 6 Å is usually sufficient.

`READ`: Read in the protein structure.

`LOOP`: Turn on LOOP.

> arg1: N-terminus atom number for the loop. This is the atom number for the peptide Nitrogen atom that joins the loop to the protein at the N-terminus of the loop. Atom # 423 for this example.
>
> arg2: C-terminus atom number for the loop. This is the atom number for the peptide Carbon atom that joins the loop to the protein at the C-terminus of the loop. Atom # 467 for this example.
>
> arg3=0: Use the sequence for the loop from the protein structure supplied.
>
> arg4=4: Generate four candidate loop structures. Typically you would want to generate many more than this.
>
> arg5=0.0: Save up to the default number of structures (10,000).
>
> arg6=0.26: If a given pair of atoms is closer than arg6 times the sum of their van der Waals radii after loop generation just prior to minimization then the structure is rejected and another loop structure is generated. Default: 0.25.
>
> arg7=0.0: Automatically add all heavy atoms in the loop to the comparison list used in identifying conformations that are identical (`COMP`) entries. Also automatically add all chiral atoms in the loop to the list of chiral atoms that need to be checked (`CHIG` entries) before accepting a generated conformer.

`DEMX`: arg5=1000.0: Keep conformers that are up to 1000 kJ/mol higher in energy than the lowest energy conformers. Such a large value might be appropriate if a follow-up study using a conformational search method like LLMOD were going to be conducted.

`MSYM`: arg2=1: Use mmsym to compare conformers. The comparison is done in-place, that is, translations and rotations of the protein as a whole are not permitted when comparing structures.

`MCOP`: Print messages to the log file concerning every conformation generated.

`CONV`: Minimizations are converged when the RMS gradient is less than 0.1 kJ/mol-Å.

`MINI`: Minimize the structure using the TNCG minimizer.

This LOOP run also writes out `COMP` and `CHIG` command lines containing the shifted atom numbers to the `.log` file for explicitly and implicitly specified `COMP` atoms and `CHIG` atoms in the system. These lines could be inserted into the command files for subsequent studies.

12.3.2 LOOP Job Using the Sequence From a File

This example is identical to the previous command file except that arg3 of `LOOP` is 1, indicating that a *filename*`.lsq` is used to specify the sequence for the loop.

12.4 Example LOOP Job Output

12.4.1 Atom Renumbering Using Input Protein Structure

The excerpt below shows the atom renumbering formation from the `.log` file generated during a run in which the loop sequence from the actual protein structure was used as input.

```
LOOP being generated using sequence from input structure
Note Substructure information may have changed.
Recording Substructure information in: MCPC603-out.sbc

Beginning of new list of COMP and CHIG commands

COMP   6607   6608   6609   6610   0.0000   0.0000   0.0000   0.0000
COMP   6611   6612   6613   6614   0.0000   0.0000   0.0000   0.0000
COMP   6615   6616   6617   6618   0.0000   0.0000   0.0000   0.0000
COMP   6619   6620   6621   6622   0.0000   0.0000   0.0000   0.0000
COMP   6623   6624   6625   6626   0.0000   0.0000   0.0000   0.0000
COMP   6627   6628   6629   6630   0.0000   0.0000   0.0000   0.0000
COMP   6631   6632   6633   6634   0.0000   0.0000   0.0000   0.0000
COMP   6635   6636   6637   6638   0.0000   0.0000   0.0000   0.0000
COMP   6639   6640   6641   6642   0.0000   0.0000   0.0000   0.0000
COMP   6643   6644   6645   6646   0.0000   0.0000   0.0000   0.0000
COMP   6647   6648   6649   6650   0.0000   0.0000   0.0000   0.0000
COMP   6651   6652   6653   6654   0.0000   0.0000   0.0000   0.0000
CHIG   6612   6617   6623   6626   0.0000   0.0000   0.0000   0.0000
CHIG   6630   6641   6650      0   0.0000   0.0000   0.0000   0.0000

End of new list of COMP and CHIG commands
```

12.4.2 Conformational Search Using Input Protein Structure

The excerpt below is from the same `.log` file as the excerpt above. This `.log` file was generated using the loop sequence from the actual protein structure.

```
         Step     1 New global minimum.  E (kJ/mol) =    -3830.85
Conf    1 E = -3830.85 (  0.057) is unique and stored as structure 1
 Search initialized with  1  structures from the .dat file

Conf    2 E = -3668.40 (  0.077) is unique and stored as structure 2
Conf    3 E = -3652.38 (  0.043) is unique and stored as structure 3
Conf    4 E = -3630.91 (  0.096) is unique and stored as structure 4
```

```
Final report:
    4 unique conformations found
    4 minimized with good convergence
Found     1 confs within  1.00 kcal/mol ( 4.18 kJ/mol) of glob. min.
Global minimum E =  -3830.85 found     1 times.

 BatchMin normal termination
 Total number of structures processed = 4
 Conformations with poor convergence marked with a *
 Conformation       1 (   -3830.854     kJ/mol) was found     1 times
 Conformation       2 (   -3668.398     kJ/mol) was found     1 times
 Conformation       3 (   -3652.378     kJ/mol) was found     1 times
 Conformation       4 (   -3630.911     kJ/mol) was found     1 times
           *** MC Statistics ***
 Percent of minimized structures within energetic window: 100.0000000
 Average number of duplicates:   1.000000000
 Duplication standard deviation: 0.0000000000E+00
 5  structures generated
 0  rejected by ring closure
 2  rejected by van der Waals
 0  duplicate minimised structures
              Time in Monte Carlo generation loop:       5.8 CPU sec
                     Time in energy minimizations:     153.4 CPU sec
                    Time in geometry optimisation:       0.0 CPU sec
```

12.4.3 LOOP Run Using an .lsq File as Input

The `.log` file for LOOP runs that use auxiliary *filename*`.lsq` files looks essentially the same as those in the previous two examples, except that when an `.lsq` file is used, the following statement appears in the `.log` file output:

```
 LOOP being generated using sequence in .lsq file
```

Embrace

You can obtain a set of ligands that have been pre-positioned with respect to a receptor from various sources, including Schrödinger's docking program, Glide. To study the association of the ligands with the receptor further, you can use the automated mechanism of Multi-Ligand Bimolecular Association with Energetics (Embrace). With Embrace, complexes can be studied using simple minimizations or conformational searches. Embrace calculations can be distributed over multiple processors, either from Maestro or from the command line.

13.1 Minimizations With Embrace

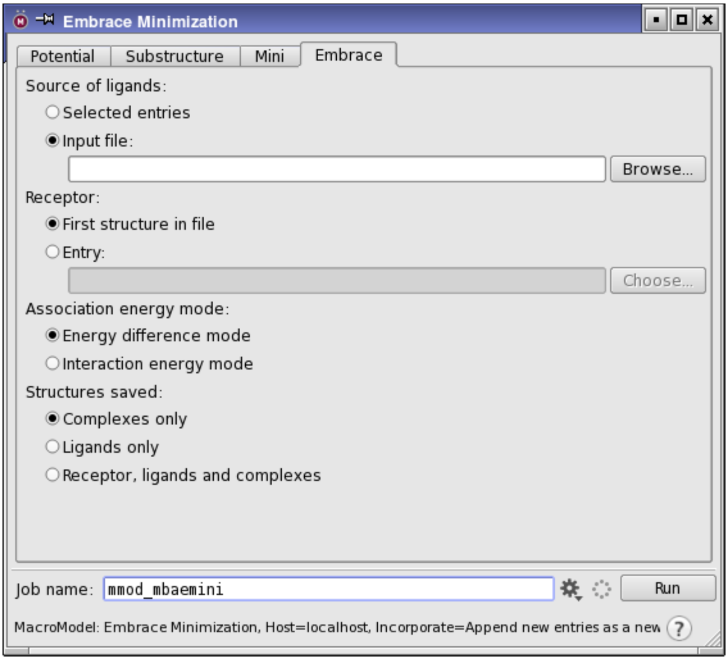
An Embrace minimization is a type of multiple minimization in which each of the specified pre-positioned ligands is minimized, in turn, with the receptor. You can perform Embrace minimization calculations in two modes: Interaction Mode, in which the interaction between each ligand and the substrate is studied, and Energy Difference Mode, in which energy changes upon association are estimated.

The Embrace Minimization panel is used to set up and submit Embrace minimization jobs. To open this panel, choose Embrace Minimization from the MacroModel submenu of the Applications menu in the main menu bar. The upper and lower parts of the panel and the Potential and Substructure tabs are common to all MacroModel panels. These components are described in detail in Section 4.1 on page 31 through Section 4.5 on page 40. The Mini tab is common to many of the MacroModel panels. For an explanation of the controls in this tab, see Section 6.1 on page 51. The controls for the Embrace minimization settings are located in the Embrace tab.

To perform an Embrace minimization, first configure the general job settings in the upper portion of the Embrace Minimization panel and the Potential and Substructure tabs as described in Chapter 4. Because the receptor is usually large, you should consider defining substructures and fixing or freezing atoms that are far from the active site, to speed up the calculations. Then, configure the Embrace settings in the Embrace tab, discussed below.

Source of ligands

You can select ligands for an Embrace calculation from the entries in the Project Table, or you can read the ligands from a file. To specify the file, you can enter the path to the file in the text box, or you can click Browse and navigate to the file.



Source of ligands:

○ Selected entries

● Input file:

Receptor:

● First structure in file

○ Entry:

Association energy mode:

● Energy difference mode

○ Interaction energy mode

Structures saved:

● Complexes only

○ Ligands only

○ Receptor, ligands and complexes

Job name: mmod_mbaemini

MacroModel: Embrace Minimization, Host=localhost, Incorporate=Append new entries as a new

Figure 13.1. The Embrace Minimization panel showing the Embrace tab.

Receptor

For an Embrace calculation to execute correctly, it must correctly identify which structure is to be treated as the receptor.

If you are reading the ligands from file, you can nominate the first structure in the file as the receptor, or you can specify it as an entry in the current project. If the receptor is not in the file, you must specify it as an entry in the current project. You can either type the entry name in the text box or click Choose and select an entry in the Choose Entry dialog box that is displayed. In this case, the receptor is written to a separate file for the job.

If the ligands are entries in the current project, you can nominate the first selected entry as the receptor, or you can choose another entry for the receptor. To choose an entry, type the entry name in the text box or click Choose and select an entry in the entry selection dialog box that is displayed.

Association energy mode

The Embrace calculation can function in two modes. The first mode is Energy difference mode. In this mode, the calculation is performed first on the receptor, then on the ligand, and finally on the complex. The energy difference is then calculated using the equation:

$$\Delta E = E_{complex} - E_{ligand} - E_{protein}$$

The full effects of relaxation and solvation are included in this mode.

The second is Interaction energy mode. In this mode, the atoms in the ligand and the receptor are separated into two sets, and the interaction energy between the two sets is calculated. Interaction energy mode deals with terms that can be considered pair-wise additive, so the surface energy term in the solvation energies is not included in the interaction energy. Interaction energy mode does not include the relaxation or the change in solvation of the ligand on binding.

To specify a preference for calculation of the association energy, select either Energy Difference Mode or Interaction Energy Mode.

Structures saved

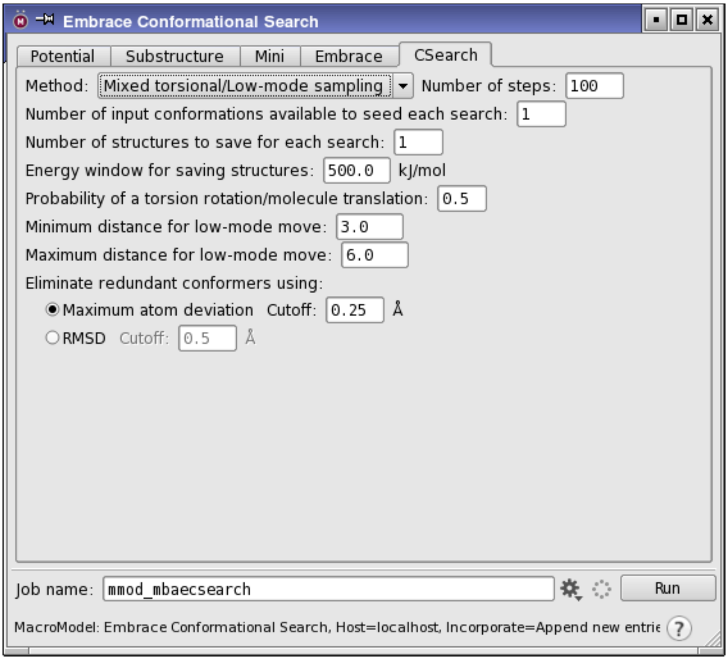
The three options for how output structures are written to the output structure file are: Complexes only (minimized complexes); Ligands only (ligand structures extracted from minimized complexes); and Receptor, ligands, and complexes, in which the receptor minimized without a ligand and ligands minimized without the receptor are written. For interaction mode, the last option is equivalent to Complexes only.

13.2 Conformational Searches With Embrace

Embrace can perform conformational searches in addition to minimizations. Energy difference mode is the only mode supported for conformational searches with Embrace. Searches are conducted on the receptor, each ligand, and each ligand-receptor complex. The energies used in the energy difference equation

$$\Delta E = E_{complex} - E_{receptor} - E_{ligand}$$

for the receptor and the ligand are the values from the lowest energy conformations found for those systems. However, multiple complex conformations may be retained and a separate $E_{complex}$ is used in the energy difference equation for each one (`MCOP` arg6 specifies the number of such conformations to keep).



Method: Mixed torsional/Low-mode sampling Number of steps: 100
Number of input conformations available to seed each search: 1
Number of structures to save for each search: 1
Energy window for saving structures: 500.0 kJ/mol
Probability of a torsion rotation/molecule translation: 0.5
Minimum distance for low-mode move: 3.0
Maximum distance for low-mode move: 6.0
Eliminate redundant conformers using:
● Maximum atom deviation Cutoff: 0.25 Å
○ RMSD Cutoff: 0.5 Å
Job name: mmod_mbaecsearch
MacroModel: Embrace Conformational Search, Host=localhost, Incorporate=Append new entrie

Figure 13.2. The Embrace Conformational Search panel showing the CSearch tab.

The Embrace Conformational Search panel is used to set up and submit Embrace conformational search jobs. To open this panel, choose Embrace Conformational Search from the MacroModel submenu of the Applications menu in the main menu bar. The upper and lower parts of the panel and the Potential and Substructure tabs are common to all MacroModel panels. These components are described in detail in Section 4.1 on page 31 through Section 4.5 on page 40. The Mini tab is common to many of the MacroModel panels. For an explanation of the controls in this tab, see Section 6.1 on page 51. The controls for the Embrace settings are located in the Embrace tab, which is described in Section 13.1 on page 121. The conformational search parameters are set up in the CSearch tab. This tab contains a subset of the controls found in the CSearch tab for regular conformational searches and is described below.

Embrace conformation searches automatically employ the `AUTO` Automatic Setup mechanism. `AUTO` selects the MCMM and comparison atom parameters for each individual ligand-receptor complex. Computations prepared from Maestro have the `AUTO` opcode added to the job's command file (*jobname*`.com`) automatically. In addition, `AUTO` is substructure-aware. Parameters are only indicated for the freely moving receptor and ligand atoms, but not for fixed or frozen regions of the receptor. Thus, only a proper receptor substructure needs to be indicated to prepare the appropriate MCMM parameters. Examples can be found in Section 13.3 on page 126. In addition, only the substructure facility can be used to indicate fixed or frozen atoms in an Embrace calculation, and ligand atoms cannot be fixed or frozen. Other than by adjusting the substructure, it is not possible to adjust the `AUTO` parameters for individual complexes in an Embrace automated calculation.

Conformational searches on protein-ligand complexes are computationally intensive, and the use of substructures is strongly recommended to reduce the CPU time and memory required. Even with fairly small substructures and very short searches, such searches require hours or days for each ligand processed.

To perform an Embrace conformational search, first configure the general job settings in the upper portion of the **Embrace Minimization** panel and the **Potential** and **Substructure** tabs. Then configure the Embrace settings in the **Embrace** tab and the conformational search settings in the **CSearch** tab, which are described below.

Method

There are only three methods available for performing an Embrace conformational search: **Torsional sampling (MCMM)**, **Low-mode sampling**, and **Mixed torsional/Low-mode sampling**. Large scale low-mode (`LMC2`) searches are not supported at this time. Low-mode calculations are limited to a few hundred atoms and should be used for systems with smaller ligand/substructure combinations. However, for such systems, low-mode searches can be comparatively efficient. MCMM conformational searches are less memory-intensive, and thus may be used on larger portions of the system provided that the number of degrees of freedom changed at any one time is limited and the substructure carefully selected.

Number of steps

The number of steps that will be performed in any search is determined by the value entered in this text box. When the number of generated trial structures matches the value given, the conformational search is terminated.

Number of input conformations available to seed each search

The value in this text box specifies the number of conformers available for each ligand to seed the search. The file containing the ligands must have exactly the number of conformers specified for each ligand, otherwise the search will fail when it reaches the ligand that does not have the specified number. Multiple conformers could, for example, come from the use of `COPY`/`ALGN`, from a separate conformational search in which a specified number of conformations is kept, or from Glide poses.

Number of structures to save for each search

Specify the number of structures to save for each search, counting from the lowest in energy. A zero value means “save all structures.”

Energy window for saving structures

This is the threshold value for comparison of trial structures. Any new structures generated and minimized are kept only if their energy is less than this value above the current global minimum. Lowering this value results in fewer structures saved. The default value is set to 500 kJ/mol.

Probability of a torsion rotation/molecule translation
Minimum distance for low-mode move
Maximum distance for low-mode move

These three text boxes are relevant only to the low-mode searches and are active only when a method involving low-mode conformational searching is selected.

The first text box is used only with the Mixed torsional/Low-mode sampling method and allows the setting of a probability that any defined torsion rotations and molecule translations are made at each step during the search. This should be a number from 0.0 to 1.0.

The other two text boxes are used for setting the minimum and maximum distance for a low-mode move. During a search, the fastest moving atom is moved randomly generated distances that are between these two limits.

Eliminate redundant conformers using

If comparison atoms are chosen or Perform automatic setup during calculation is selected, structures produced during a conformational search are compared to see if they are unique. Two options are available for the conditions for structures to be considered different:

- Maximum atom deviation—consider structures to be different if the maximum atom deviation for any atom exceeds the threshold given in the Cutoff text box.
- RMSD—consider structures to be different if the RMS deviation for all compared atoms exceeds the threshold given in the Cutoff text box.

The default cutoff is 0.5 Å for both options.

13.3 Specifying a Substructure for Embrace

The use of substructures can dramatically speed up Embrace calculations. Specifying substructures is described in Section 4.5 on page 40. Substructures used in Embrace must meet an additional requirement: the receptor atoms must be numbered starting from 1 when constructing the substructure. Below are two simple recipes for setting up substructures. In the first, you can define shells of constrained atoms near the association site in the receptor. In the second, all atoms in the receptor are either fixed or frozen. If you already have a suitable substructure, you can simply read it in by clicking Read .sbc File near the bottom of the Substructure tab.

13.3.1 Using a Ligand To Assist in Creating a Receptor Substructure

In this example, we will use a ligand from an entry called `our_lig` to set up a substructure for the receptor. The substructure will contain a shell of all receptor atoms from complete residues that lie within 6.0 Å of the ligand. Two additional shells, one fixed and one frozen, are also specified.

1. Include the receptor in the Workspace and then include the ligand.

   It is important to include the receptor first to ensure that its atoms are assigned atom numbers starting from 1. You can now create a substructure in the usual manner. Maestro's support for the Embrace utility automatically ignores atoms within the substructure that are numbered higher than the atoms in the receptor so the inclusion of ligand atoms is not a problem when you use the Maestro interface. For each ligand that Embrace is applied to, all atoms in the ligand are automatically added to the substructure without constraints.

2. In the Atoms for substructure section, select Molecule from the Pick menu and pick a ligand atom.

3. Select Complete residues and enter `6.0` in the Expand to atoms within radius of text box.

   Alternatively, enter the following ASL expression in the ASL text box:

   ```
   fillres within 6.0 entry.name our_lig
   ```

Now add a shell of fixed receptor atoms from complete residues within 5.0 Å of the substructure atoms:

4. Click New Shell to create a new shell.

5. In the Radius text box, enter `5.0` and select Complete Residues.

To add another shell containing frozen receptor atoms from complete residues within 4.0 Å of the previously defined shell, follow the same procedure and select Freeze atoms.

13.3.2 Creating a Substructure with All Receptor Atoms Fixed or Frozen

1. Include the receptor, and only the receptor, in the Workspace.

2. Click New Shell.

3. In the Additional atoms for shell section, click All. By default, all atoms are fixed, that is, constrained to their current positions using a harmonic potential.

4. To freeze the atoms in place, select Freeze atoms.

13.4 Command File Examples

To run MacroModel calculations, a molecular structure file and a command file are required. The molecular structure file contains the structures to be used as input in the calculation. The command file contains the name of the input structure file, the name of the output structure file, and an ordered list of operation codes (*opcodes)* for the calculations.

When you start a job from Maestro or write out the input files, Maestro writes out a molecular structure file and a command file. For many types of jobs, command files written this way are complete and adequate. For some types of jobs, however, including Embrace conformational searches, you will need to adjust the Maestro-generated command file.

The command files and the log files for the examples given in this section can be found in `$SCHRODINGER/macromodel-v`*version*`/samples/Examples`.

13.4.1 Embrace Minimization Calculations

You can set up the files for Embrace minimization calculations using Maestro. This is the recommend mechanism for setting up such calculations. This section explains the structures of Embrace minimization `.com` files if you want to customize them.

13.4.1.1 Interaction Mode

Interaction mode uses the `ASET` mechanism to define atom sets and to calculate interaction energies within and among all such sets. All results for sets are saved in the log file for the run. Only interaction energies between sets 1 and 2 are saved as project properties within the output structure file. Note that the surface energy contribution in GB/SA calculations is associated with the interaction of a set with itself.

Below is an example command file for an Embrace job run in interaction energy mode. The input file, `embrace_input.mae`, follows the Glide pose-viewer pattern, that is, the first structure in the file must be the receptor and the remaining structures should be ligands previously positioned appropriately relative to the receptor. A description of the opcodes used in the file follows, then an excerpt from the output.

```
embrace_input.mae
embrace_interaction-out.mae
 FFLD     11      1      0      0     1.0000     0.0000     0.0000     0.0000
 SOLV      3      1      0      0     0.0000     0.0000     0.0000     0.0000
 EXNB      0      0      0      0     0.0000     0.0000     0.0000     0.0000
 BDCO      0      0      0      0    89.4427 99999.0000     0.0000     0.0000
 MBAE      0      0      0      0     0.0000     0.0000     0.0000     0.0000
 SUBS      0      0      0      0     0.0000     0.0000     0.0000     0.0000
 READ      0      0      0      0     0.0000     0.0000     0.0000     0.0000
 BGIN      0      0      0      0     0.0000     0.0000     0.0000     0.0000
 READ      0     0     0     0    0.0000    0.0000    0.0000    0.0000
 ASET      0     0     0     0    0.0000    0.0000    0.0000    0.0000
 ASET      0     0     0     0    2.0000    0.0000    0.0000    0.0000
 ASET      1  4598     0     0    2.0000   -2.0000    0.0000    0.0000
 ASET      1  4598     0     0    1.0000    2.0000    0.0000    0.0000
 CONV      2     0     0     0    0.1000    0.0000    0.0000    0.0000
 MINI      1     0  2000     0    0.0000    0.0000    0.0000    0.0000
 ELST     -1     0     0     0    0.0000    0.0000    0.0000    0.0000
 END       0     0     0     0    0.0000    0.0000    0.0000    0.0000
 MBAE     -1     0     0     0    0.0000    0.0000    0.0000    0.0000
```

`SOLV`: The GB/SA effective water model is being used.

`FFLD`: Use OPLS_2001 with constant dielectric electrostatics, which is appropriate if GB/SA solvation is being used.

`EXNB`: Extended non-bonded cut-offs should be used with GB/SA solvation.

`BDCO`: Use the Bond Dipole CutOff (BDCO) method for truncating electrostatic interactions. Arg5 and arg6 are used to specify the cutoffs used for charge-dipole and charge-charge interactions, respectively.

`MBAE`: Turn on Embrace.

arg1 – 0: Using Interaction energy mode.

arg2 – 0: MINI calculations being performed.

arg3 – 0: Minimized complex structures are written out.

If arg3 were 1, only the ligand structures extracted from the minimized complex structure would be recorded.

`READ`: The first `READ` command obtains the structure of the receptor.

`SUBS`: Substructures may be used. All opcodes related to the substructure (`SUBS`, `FXAT`, `FXDI`, `FXBA`, and `FXTA`) should be defined in an `.sbc` file. The `.sbc` file should contain only lines referring to the receptor with the receptor atoms numbered starting from 1.

`BGIN`: Start the loop that processes each substructure in turn.

`READ`: Read in a ligand. This also produces a combined structure consisting of the receptor and the ligand with the receptor atoms being numbered lower than the ligand atoms. All ligand atoms are automatically added to the `SUBS` list at this stage.

`ASET`: This series of `ASET` opcodes places the substrate atoms in set 1 and all other atoms in set 2. If the ligand structure has multiple molecules in it, then this case would place all of the molecules for that structure in set 2. The first `ASET` line removes all of the atoms from the existing set (by placing them in set 0, which is a dummy set). The second `ASET` line places all atoms in set 2. The third and fourth `ASET` commands remove the receptor atoms (numbered from 1 to 4729) from set 2 and place them in set 1. Other `ASET` combinations may be useful depending on the system (e.g., you could leave out particular molecules, such as water molecules from the substrate CT). Note that while the interaction energies from all sets are recorded in the `.log` file, only those for sets 1 and 2 are recorded in the output structure file for eventual inclusion in Maestro's Project Table.

`MINI`: Minimize the energy of the structure using the PRCG minimizer.

`ELST`: Invokes the `ASET` mechanism for calculating interaction energies. Arg1 = 1 causes limited information to be stored in the `.log` file only. Other values may produce very large `.mmo` files.

`END`: End the loop over ligands.

`MBAE`: When arg1 = –1, `MBAE` is turned off. This command is not needed in this example.

Below is the data generated by Embrace for one ligand. The complete output file contains one such set for each ligand in the analysis set. Note that Conf 2 refers to the first ligand in the file, which is the second structure in the input structure file. Note also that only the energetic results between sets 1 and 2 are recorded in the output structure file.

A table is given at the end of the log file containing a summary of the results for all the ligands.

```
 Conf    5 E =  -11776.075 (   0.098) kJ/mol
Using numerical surfaces and analytical Born radii
Solvation GB set energies do not include constant
 contributions exclusively involving fixed/frozen atoms.
Energetic Interactions Within Atom Sets (with no. of interactions):
  Atom set   1:
    Total Energy (kJ/mol) =   -21703.7729 (   2063797 )
                  Stretch =       97.4944 (      2039 )
                     Bend =       95.2108 (       396 )
           Proper Torsion =      676.4688 (      4727 )
             Out-of-Plane =        4.9817 (        49 )
            Electrostatic =    -5172.1630 (    289798 ) (Part of nonbonded)
            Van der Waals =     -396.6492 (     40968 ) (Part of nonbonded)
             Solvation SA =       52.1844 (      2032 )
             Solvation GB =   -17216.1112 (   1723031 )
    Fixed-atom constraint =      154.8104 (       757 )
               Non-bonded =    -5568.8122 (    330766 ) (Elect + Hbnd + vdW)
  Atom set   2:
    Total Energy (kJ/mol) =     -362.1657 (      2814 )
                  Stretch =        6.1540 (        40 )
                     Bend =       25.9271 (        73 )
           Proper Torsion =       52.2179 (       106 )
```

```
             Out-of-Plane =         0.8049 (         7 )
            Electrostatic =      -451.4534 (       590 ) (Part of nonbonded)
            Van der Waals =        38.5829 (       516 ) (Part of nonbonded)
             Solvation SA =         1.2150 (        38 )
             Solvation GB =       -35.6141 (      1444 )
    Fixed-atom constraint =         0.0000 (         0 )
               Non-bonded =      -412.8705 (      1106 ) (Elect + Hbnd + vdW)
Energetic Interactions Between Atom Sets (with no. of interactions):
  Atom sets   1 and   2:
 Use these for Embrace interaction energies for ligand: Cmg_deriv
    Total Energy (kJ/mol) =      -227.2996 (    218888 )
                  Stretch =         0.0000 (         0 )
           Proper Torsion =         0.0000 (         0 )
             Out-of-Plane =         0.0000 (         0 )
            Electrostatic =      -190.7598 (     66953 ) (Part of nonbonded)
            Van der Waals =      -124.7660 (      9251 ) (Part of nonbonded)
             Solvation GB =        88.2263 (    142684 )
               Non-bonded =      -315.5259 (     76204 ) (Elect + Hbnd + vdW)
 Total Energy =  -11776.070 kJ/mol
 =====================================
Embrace Results

 Interaction Energy (ASET: sets 1 and 2) Results

Ligand ...Interaction Energy.............................................  Mini.
   #   Total Excl.  Valence    VDW    Electro-  Hydrogen     Sol-  Con-   Done
       Constraints                      static     Bonds    vation   straint
    1   -249.28      0.00   -167.62   -129.72      0.00     48.07      0.00 T
    2   -250.65      0.00   -168.11   -129.21      0.00     46.67      0.00 T
    3   -217.48      0.00   -116.60   -191.43      0.00     90.56      0.00 T
    4   -227.30      0.00   -124.77   -190.76      0.00     88.23      0.00 T
```

13.4.1.2 Energy Difference Mode

Energy difference mode first applies the type of calculation selected (currently only minimization is supported) to the receptor by itself. For each ligand in turn, it then applies the calculation to the ligand by itself and the complex of the ligand with the receptor, and reports the energy difference.

Below is an example command file for an Embrace job run in energy difference mode. A description of the opcodes used in the file follows.

```
embrace_input.mae
embrace_eDiff-out.mae
 FFLD      11      1      0      0     1.0000     0.0000     0.0000     0.0000
 SOLV       3      1      0      0     0.0000     0.0000     0.0000     0.0000
 EXNB       0      0      0      0     0.0000     0.0000     0.0000     0.0000
 BDCO       0      0      0      0    89.4427 99999.0000     0.0000     0.0000
 MBAE       1      0      0      0     0.0000     0.0000     0.0000     0.0000
 SUBS       0      0      0      0    0.0000     0.0000     0.0000     0.0000
 READ       0      0      0      0    0.0000     0.0000     0.0000     0.0000
 CONV       2      0      0      0    0.1000     0.0000     0.0000     0.0000
 MINI       1      0   2000      0    0.0000     0.0000     0.0000     0.0000
 BGIN       0      0      0      0    0.0000     0.0000     0.0000     0.0000
 READ       0      0      0      0    0.0000     0.0000     0.0000     0.0000
 CONV       2      0      0      0    0.1000     0.0000     0.0000     0.0000
 MINI       1      0   2000      0    0.0000     0.0000     0.0000     0.0000
 END        0      0      0      0    0.0000     0.0000     0.0000     0.0000
 MBAE      -1      0      0      0    0.0000     0.0000     0.0000     0.0000
```

`SOLV`: The GB/SA effective water model is being used.

`FFLD`: This example uses OPLS_2001 with constant dielectric electrostatics, which is appropriate if GB/SA solvation is being used.

`EXNB`: Extended non-bonded cut-offs should be used with GB/SA solvation.

`MBAE`: Turn on Embrace.

arg1 = 1: Using Energy Difference mode.

arg2 = 0: MINI calculations being performed.

arg3 = 0: Minimized complex structures are written to the output structure file.

`SUBS`: Substructures may be used. All opcodes related to the substructure should be defined in an `.sbc` file. The `.sbc` file should contain only lines referring to the receptor with the receptor atoms numbered starting from `1`.

`READ`: The first `READ` command obtains the structure of the receptor.

`MINI`: Minimize the energy of the receptor. Arg1=1 uses the PRCG minimization technique for up to arg3 (2000) iterations.

`BGIN`: Start the loop which processes each ligand in turn.

`READ`: Sequentially read the ligand structures.

`MINI`: Minimize first the current ligand structure, then minimize the receptor-ligand complex. Record the energy differences to the `*.log` file and to the output structure file.

`END`: End the loop for ligands.

`MBAE`: When arg1 = −1, Embrace is turned off. In this example, this command is not needed.

Below is the Embrace output from the `.log` file for one ligand in a job run in energy difference mode, showing the energy differences upon complexation.

```
 Embrace Energy Differences for Ligand #           1
 Ligand Name: Omo_4erkLig
      Total Energy(constraints) =      -43.3443 kJ/mol
   Total Energy(no constraints) =      -38.2964 kJ/mol
                Valence Energy =       18.3260 kJ/mol
                           VDW =     -161.5383 kJ/mol
                 Electrostatic =      -60.9017 kJ/mol
        Explicit Hydrogen Bonds =        0.0000 kJ/mol
                     Solvation =      165.8165 kJ/mol
                    Constraint =       -5.0478 kJ/mol
                     Converged =    T
```

In this example, `Converged = T` means that all minimization converged. If the value were `F` instead of `T`, then the energy differences would be suspect and the calculation may need to be repeated with more minimization steps.

At the end of the log file is a table summarizing the results for all the ligands.

Substructures can be specified using ASL. The substructure file has to be created by hand. An example of a substructure file that uses ASL is:

`$SCHRODINGER/macromodel-v`*`version`*`/samples/Examples/embrace_input_asl.sbc`

This file is used with the command file `embrace_eDiff_asl.com`, and is reproduced here:

```
 ASL1      -1 all
 FXAT       0       0       0       0 -1.0000     0.0000     0.0000     0.0000
 FXAT       0       0       0       0  0.0000     0.0000     0.0000     0.0000
 ASL1       0 not (mol.num 2) and fillres within 2.50 ( mol.num 2 )
 ASL1      -1 m. 2
 ASL2   100.0 (not(fillres within 2.5 m. 2)) and (fillres within 3.0 (fillres 
within 2.5 m. 2))
 ASL2    -1.0 (not(fillres within 3.0(fillres within 2.5 m. 2))) and (fillres 
within 3.0(fillres within 3.0(fillres within 2.5 m. 2)))
```

Substructure addition and removal are performed with the `ASL1` command. Fixed and frozen atom definitions are made with the `ASL2` command. See the *MacroModel Reference Manual* for information on the syntax of these commands.

In the above example:

The first three lines clear all free, fixed atom and frozen atom definitions respectively. Although this is not very useful in the context of Embrace, it is useful when a `SUBS` command is used within a `BGIN`/`END` loop.

The fourth line adds the atoms defined by the ASL string to the substructure. In this case `mol.num 2` (molecule number 2) is the ligand. Thus, the binding site, consisting of residues within 2.5 Å of the ligand, is added to the substructure.

The fifth line indicates that the ligand should not be part of this substructure. Embrace calculations handle ligands separately.

The sixth line specifies the atoms that are to be constrained with a force constant of 100.0 kJ mol^{-1} $\mathrm{Å}^{-2}$. The ASL expression selects atoms that are within 3.0 Å of the atoms defined by the fourth line, excluding atoms defined by the fourth line.

The seventh line specifies the atoms that are to be frozen. The ASL expression selects atoms that are within 3.0 Å of the atoms defined by the sixth line, excluding atoms defined by the sixth line.

Note: These two lines are wrapped in the example above, but should be a single line in the file.

In this case the receptor is only one molecule. In cases where receptor consists of more than one molecule, use `mol.num` *n* or `m.` *n* for the ligand definition, where *n* is the molecule number of the ligand.

13.4.2 Conformational Searches With Embrace

Below are example `.com` files for an `MCMM` and an `LMOD` conformational search with Embrace. After these an example is provided using `COPY` and `ALGN` in conjunction with an `MCMM`/`MBAE` conformational search.

Like Embrace minimization calculations, Embrace conformational searches include a table of results at the end of the `.log` file. Energetic properties are also included in this output structure file.

Embrace conformational searches can use very large amounts of computer time and memory. The memory requirements are similar to the equivalent non-Embrace search of the complex of the protein with the largest ligand. Because of these requirements, for nearly all receptors it will likely be necessary to specify a substructure to reduce the required resources to a practical level. If you want to perform only the Embrace search on the ligand within an essentially static receptor, then the substructure used should not include `SUBS` lines but only list the receptor atoms as fixed or frozen (see the `FXAT` opcode description for more information) and use only one step for the first `MCMM` or `LMOD` line (the one for the search of the receptor).

13.4.2.1 MCMM Conformational Search

This example uses Monte Carlo Multiple Minimum searching. A copy of this file is available at:

`$SCHRODINGER/macromodel-v`*version*`/samples/Examples/embrace_mcmm.com`

```
embrace_input.mae
embrace_mcmm-out.mae
 FFLD      11       1       0       0      1.0000      0.0000      0.0000      0.0000
 SOLV       3       1       0       0      0.0000      0.0000      0.0000      0.0000
 EXNB       0       0       0       0      0.0000      0.0000      0.0000      0.0000
 BDCO       0       0       0       0     89.4427  99999.0000      0.0000      0.0000
 MSYM       0       0       0       0      0.0000      0.0000      0.0000      0.0000
 MBAE       1       1       0       0      0.0000      0.0000      0.0000      0.0000
 MCMM      10       0       0       0      0.0000      0.0000      0.0000      0.0000
 MCNV       1       5       0       0      0.0000      0.0000      0.0000      0.0000
 MCSS       2       0       0       0    500.0000      0.0000      0.0000      0.0000
 MCOP       1       0   10000       0      0.0000      1.0000      0.0000      0.0000
 DEMX       0       0       0       0    500.0000      0.0000      0.0000      0.0000
 CRMS       0       0       0       0      0.0000      0.2500      0.0000      0.0000
 SUBS       0       0       0       0      0.0000      0.0000      0.0000      0.0000
 READ       0       0       0       0      0.0000      0.0000      0.0000      0.0000
 AUTO       0       2       1       1      1.0000      0.0000      5.0000      0.0000
 CONV       2       0       0       0      0.1000      0.0000      0.0000      0.0000
 MINI       1       0    2000       0      0.0000      0.0000      0.0000      0.0000
 AUTO       0       2       1       1      1.0000      1.0000      5.0000      0.0000
 MCMM      10       0       0       0      0.0000      0.0000      0.0000      0.0000
 MCOP       1       0   10000       0      0.0000      1.0000      0.0000      0.0000
 BGIN       0       0       0       0      0.0000      0.0000      0.0000      0.0000
 READ       0       0       0       0      0.0000      0.0000      0.0000      0.0000
 MINI       1       0    2000       0      0.0000      0.0000      0.0000      0.0000
 END        0       0       0       0      0.0000      0.0000      0.0000      0.0000
 MBAE      -1       0       0       0      0.0000      0.0000      0.0000      0.0000
```

`FFLD`: Use OPLS_2001 with constant dielectric electrostatics, which is appropriate if GB/SA solvation is being used.

`SOLV`: The GB/SA effective water model is being used.

`EXNB`: Extended non-bonded cut-offs should be used with GB/SA solvation.

`BDCO`: Use the Bond Dipole CutOff (`BDCO`) method for truncating electrostatic interactions. Arg5 and arg6 specify the cutoffs used for charge-dipole and charge-charge interactions, respectively.

`MSYM`: Invokes the numbering symmetry library `mmsym`, which automatically and more generally identifies a suitable numbering order for using in comparing conformations.

`MBAE`: Turn on Embrace

arg1 = 1: Using Energy Difference mode.

arg2 = 1: Conformation searches will be performed in Embrace calculations.

arg3 = 0: The lowest energy structures of the ligand-receptor complex are written to the output structure file.

`MCMM`: Use Monte Carlo Multiple Minimum searching. Arg1 defines the number of steps to use for the search of the receptor.

`MCNV`: Sets the number of degrees of freedom to be varied at each MC step. With different arg1 and arg2 values, the search varies a random number of degrees of freedom between the numbers defined in arg1 and arg2. For Embrace calculations, we recommend setting arg1 to a small number and arg2 to no more than 10.

`MCSS`: MC structure selection allows for the setting of selection of starting structure for the search steps. Arg1=2 defines use-directed selection of starting structures, where the least used structures will be used as starting geometries as long as they are low enough in energy (as defined in arg5). This is more efficient in exploring new areas of the potential energy surface than, for instance, a random-walk starting geometry scheme. Arg5 gives the energy window for selecting a new starting structure, which must be within arg5 kJ/mol of the lowest energy conformer found in the search.

`MCOP`: Monte Carlo options that determine what and how often data is written to the log file and how many structures are saved.

arg1 = 1: Print information for every search step to the log file.

arg3 = 10000: Use a large number here to avoid spurious lines in the Embrace summary table at the end of the run.

arg6 = 1.0: Save only the lowest energy conformation of the receptor (this value must be specified as 1.0).

`DEMX`: arg5 = 500.0 means retain only the conformations within 500 kJ/mol of the minimum energy conformation found so far.

`SUBS`: Substructures may be used. All opcodes related to the substructure (`SUBS`, `FXAT`, `FXDI`, `FXBA`, and `FXTA`) should be defined in an `.sbc` file. The `.sbc` file should contain only lines referring to the receptor with the receptor atoms numbered starting from 1.

`READ`: The first `READ` command obtains the structure of the receptor.

`AUTO`: The first `AUTO` automatically sets up the MCMM conformational search of the receptor. Here arg6 must be absent or 0.0 since the search of the receptor is not regarded as part of a serial search.

`MINI`: Minimize the energy of each conformation generated in the search of the receptor using the PRCG minimization technique, arg1=1, for up to arg3 (2000) iterations.

`MCMM`: The second `MCMM` causes Monte Carlo Multiple Minimum searching to be used in the search of each isolated ligand and each ligand-receptor complex. Arg1 defines the number of steps to use for each search. This need not match that used in the search of the receptor.

`AUTO`: The second `AUTO` sets up the MCMM search for each isolated ligand and each ligand-receptor complex. Here arg6 should be set to 1 because the searches of the isolated ligands and the complexes are regarded as part of a serial search.

`BGIN`: Start the loop which processes each ligand in turn.

`MCOP`: Monte Carlo options that determine what and how often data is written to the log file and how many structures are saved for the searches of the ligand-receptor complexes.

> arg1 = 1: Print information for every search step to the log file.
>
> arg3 = 10000: Use a large number here to avoid spurious lines in the Embrace summary table at the end of the run.
>
> arg6 = 2.0: Save only the two lowest energy conformations from the complexes for each ligand.

`READ`: Sequentially read the ligand structures.

`MINI`: Minimize the energy of each conformation generated in the search of each ligand and each ligand-receptor complex using the PRCG minimization technique, arg1=1, for up to arg3 (2000) iterations.

`END`: End the loop for the ligands.

`MBAE`: Turn off Embrace and summarize the results. arg1 = –1

13.4.2.2 Low-Mode Conformational Search

This example performs a low-mode conformational search. The input file is similar to that of the MCMM search example above. A copy of this file is available at:

`$SCHRODINGER/macromodel-v`*version*`/samples/Examples/embrace_lmcs.com`

The example is followed by descriptions of the opcodes. Opcode descriptions that are omitted may be read from the corresponding opcodes in the MCMM conformational search example given above, in Section 13.4.2.1 on page 134.

```
embrace_input.mae
embrace_lmcs-out.mae
 FFLD      11      1      0      0     1.0000     0.0000     0.0000     0.0000
 SOLV       3      1      0      0     0.0000     0.0000     0.0000     0.0000
 EXNB       0      0      0      0     0.0000     0.0000     0.0000     0.0000
 BDCO       0      0      0      0    89.4427 99999.0000     0.0000     0.0000
```

```
MSYM       0      0       0       0      0.0000     0.0000     0.0000     0.0000
MBAE       1      1       0       0      0.0000     0.0000     0.0000     0.0000
LMCS      10      0       0       0      0.0000     0.0000     3.0000     6.0000
MCSS       2      0       0       0    500.0000     0.0000     0.0000     0.0000
MCOP       1      0    1000       0      0.0000     1.0000     0.0000     0.0000
DEMX       0      0       0       0    500.0000     0.0000     0.0000     0.0000
SUBS       0      0       0       0      0.0000     0.0000     0.0000     0.0000
READ       0      0       0       0      0.0000     0.0000     0.0000     0.0000
AUTO       0      0       0       0     -1.0000     0.0000     0.0000     0.0000
CONV       2      0       0       0      0.0500     0.0000     0.0000     0.0000
MINI       1      0    2000       0      0.0000     0.0000     0.0000     0.0000
LMCS      10      0       0       0      0.0000     0.0000     3.0000     6.0000
MCOP       1      0   10000       1      0.0000     2.0000     0.0000     0.0000
BGIN       0      0       0       0      0.0000     0.0000     0.0000     0.0000
READ       0      0       0       0      0.0000     0.0000     0.0000     0.0000
MINI       1      0    2000       0      0.0000     0.0000     0.0000     0.0000
END        0      0       0       0      0.0000     0.0000     0.0000     0.0000
MBAE      -1      0       0       0      0.0000     0.0000     0.0000     0.0000
```

`LMCS`: Use the low-mode conformational search method. Arg1 defines the number of steps to use for the search of the receptor.

`AUTO`: Set up comparison atom lists, torsional constraints, and chirality checks for all the LMCS searches in this calculation. Arg5 should be set to –1 so that torsional moves (`TORS`) are not identified, and arg6 should be 0.0 or absent as the serial aspects of the searches are controlled by the next `MCOP`.

`MINI`: Minimize the energy of each conformation generated in the search of the receptor using the PRCG minimization technique, arg1=1, for up to arg3 (2000) iterations.

`LMCS`: The second `LMCS` causes low-mode conformational searching to be used in the search of each isolated ligand and each ligand-receptor complex. Arg1 defines the number of steps to use for each search. This need not match that used in the search of the receptor.

`MCOP`: Monte Carlo options that determine what and how often data is written to the log file and how many structures are saved for the searches of the ligand-receptor complexes.

arg1 = 1: Print information for every search step to the log file.

arg3 = 10000: Use a large number to avoid spurious lines in the Embrace summary table at the end of the run.

arg4 = 1: Indicates that the searches are considered to be part of a serial calculation.

arg6 = 2.0: Saves only the two lowest energy conformations from the complexes for each ligand.

13.4.2.3 MCMM Conformation Search With COPY/ALGN

Embrace conformational searches may also be used with `COPY` and `ALGN` to position ligands using a crystal structure of a complex of a closely related ligand. While this combination of commands may be useful, the positioning is crude and the searching of conformational space is slow and quite limited compared to that available in Schrödinger's docking program, Glide.

Below is an example command file for `COPY` and `ALGN` in combination with `MBAE` for an MCMM conformational search. Apart from the alignment step, the input file is similar to that of the MCMM search example above. A copy of the input file is available at:

`$SCHRODINGER/macromodel-v`*version*`/samples/Examples/embrace_algn.com`

The example is followed by descriptions of the opcodes. Opcode descriptions that are omitted may be read from the corresponding opcodes in the MCMM conformational search example given above, in Section 13.4.2.1 on page 134.

```
embrace_input.mae
embrace_algn-out.mae
 FFLD     11      1      0      0     1.0000     0.0000     0.0000     0.0000
 SOLV      3      1      0      0     0.0000     0.0000     0.0000     0.0000
 EXNB      0      0      0      0     0.0000     0.0000     0.0000     0.0000
 BDCO      0      0      0      0    89.4427 99999.0000     0.0000     0.0000
 READ      0      0      0      0     0.0000     0.0000     0.0000     0.0000
 WRIT      0      0      0      0     0.0000     0.0000     0.0000     0.0000
 READ      0      0      0      0     0.0000     0.0000     0.0000     0.0000
 COPY      0      0      0      0     0.0000     0.0000     0.0000     0.0000
 BGIN      0      0      0      0     0.0000     0.0000     0.0000     0.0000
 READ      0      0      0      0     0.0000     0.0000     0.0000     0.0000
 ALGN      3      1      5      0     0.0000     0.0000     0.0000     0.0000
 END       0      0      0      0     0.0000     0.0000     0.0000     0.0000
 RWND      0      0      0      0     0.0000     0.0000     0.0000     0.0000
 MSYM      0      0      0      0     0.0000     0.0000     0.0000     0.0000
 MBAE      1      1      1      0     0.0000     0.0000     0.0000     0.0000
 MCMM     10      0      0      0     0.0000     0.0000     0.0000     0.0000
 MCNV      1      4      0      0     0.0000     0.0000     0.0000     0.0000
 MCSS      2      0      0      0   500.0000     0.0000     0.0000     0.0000
 MCOP      1      0  10000      0     0.0000     1.0000     0.0000     0.0000
 DEMX      0      0      0      0  1000.0000     0.0000     0.0000     0.0000
 SUBS      0      1      0      0     0.0000     0.0000     0.0000     0.0000
 READ      0      0      0      0     0.0000     0.0000     0.0000     0.0000
 AUTO      0      0      0      0     0.0000     0.0000     0.0000     0.0000
 MINI      1      0   2000      0     0.0000     0.0000     0.0000     0.0000
 MCMM     50      0      0      0     0.0000     0.0000     0.0000     0.0000
 AUTO      0      0      0      0     0.0000     1.0000     0.0000     0.0000
 BGIN      0      0      0      0     0.0000     0.0000     0.0000     0.0000
 MCOP      1      0  10000      0     0.0000     2.0000     4.0000     0.0000
 READ      0      0      0      0     0.0000     0.0000     0.0000     0.0000
 MINI        1       0    2000       0      0.0000      0.0000      0.0000      0.0000
 END         0       0       0       0      0.0000      0.0000      0.0000      0.0000
 MBAE       -1       0       0       0      0.0000      0.0000      0.0000      0.0000
```

`READ`: The first `READ` reads in the receptor.

`WRIT`: Record the receptor structure in the output structure file.

`READ`: The second `READ` reads in the first (reference) ligand.

`COPY`: Copy the current structure to the reference storage area for use in aligning the ligands.

`BGIN`: Loop over all remaining ligands in the input structure file.

`READ`: Read in the next ligand structure.

`ALGN`: Align the current structure with the reference structure by center of mass and moments of inertia (arg1 = 3), weighting atom positions by mass (arg2=1), and write all 4 such alignments to the output structure file (arg3 = 5).

`END`: End of loop for aligning the ligands.

`RWND`: Rewind the output structure file and use it as the input structure file for the remainder of the calculation.

`DEMX`: arg5 = 1000.0 means retain only the conformations within 1000 kJ/mol of the minimum energy conformation found so far. We recommend using a large value to permit seed structures with different alignments to be restrained and sampled.

`SUBS`: Substructures may be used. All opcodes related to the substructure (`SUBS`, `FXAT`, `FXDI`, `FXBA`, and `FXTA`) should be defined in an `.sbc` file. The `.sbc` file should contain only lines referring to the receptor with the receptor atoms numbered starting from 1. In this case, we specify arg2 = 1 to indicate that the name of the original *jobname*`.sbc` file is to be used.

`READ`: This `READ` command obtains the structure of the receptor.

`MCMM`: The second `MCMM` selects Monte Carlo Multiple Minimum searching for the search of each isolated ligand and each ligand-receptor complex. Arg1 defines the number of steps to use for each search. This need not match that used in the search of the receptor.

`MCOP`: Monte Carlo options that determine what and how often data is written to the log file and how many structures are saved for the searches of the ligand-receptor complexes.

> arg1 = 1: Print information for every search step to the log file.
>
> arg3 = 10000: Use a large number to avoid spurious lines in the Embrace summary table at the end of the run.

arg6 = 2.0: Save only the two lowest energy conformations from the complexes for each ligand.

arg7 = 4.0: Seed the search for each ligand using all four alignments previously generated using the `ALGN` command.

13.4.3 Distributed Embrace Calculations

Any Embrace job may be distributed across a number of processors. There are two methods for distributing Embrace jobs: by specifying multiple processors when running the job (external distribution), or by adding an `NPRC` opcode to the `.com` file (internal distribution). Specifying multiple processors requires no changes to the `.com` file. The examples in this section are for the use of the `NPRC` opcode.

To distribute an Embrace job with `NPRC`, add the `NPRC` opcode early in the `.com` file, such as immediately above the `FFLD` line. See the description of the `NPRC` opcode in the *MacroModel Reference Manual* for more information. For example,

```
 NPRC       2     16      0      0     0.0000     0.0000     0.0000     0.0000
```

would run the job on two processors with 16 ligands processed in each subjob.

For energy difference Embrace calculations, the receptor is processed first within the parent process prior to starting up any child processes.

There are two restrictions on distributed Embrace jobs. First, the output structure mode, which is controlled by arg3 of the `MBAE` opcode, must be either 0 (complexes only) or 1 (ligands only) and not 2 (receptor, ligands and complexes). Second, distributed Embrace runs may not contain `COPY` and `ALGN` commands to pre-position the ligands. Instead the pre-positioning of the ligands must be accomplished in a separate non-distributed calculation. The output from that calculation is then used as input for the distributed Embrace conformational search calculation.

An example command file to align ligands prior to an Embrace conformational search calculation is given below. This file is available at

`$SCHRODINGER/macromodel-v`*version*`/samples/Examples/algn_pv.com`

```
embrace_input.mae
algn_pv-out.mae
 READ       0      0      0      0     0.0000     0.0000     0.0000     0.0000
 WRIT       0      0      0      0     0.0000     0.0000     0.0000     0.0000
 READ       0      0      0      0     0.0000     0.0000     0.0000     0.0000
 COPY       0      0      0      0     0.0000     0.0000     0.0000     0.0000
 BGIN       0      0      0      0     0.0000     0.0000     0.0000     0.0000
 READ       0      0      0      0     0.0000     0.0000     0.0000     0.0000
 ALGN       3      1      5      0     0.0000     0.0000     0.0000     0.0000
 END        0      0      0      0     0.0000     0.0000     0.0000     0.0000
```

`READ`: The first `READ` reads in the receptor.

`WRIT`: Record the receptor structure in the output structure file.

`READ`: The second `READ` reads in the first (reference) ligand.

`COPY`: Copy the current structure to the reference storage area for use in aligning the ligands.

`BGIN`: Loop over all remaining ligands in the input structure file.

`READ`: Read in the next ligand structure.

`ALGN`: Align the current structure with the reference structure by center of mass and moments of inertia (arg1 = 3), weighting atom positions by mass (arg2=1), and write all 4 such alignments to the output structure file (arg3 = 5).

`END`: End of loop for aligning the ligands.

An example command file is given below for a distributed Embrace conformational search that uses the output of the separate `COPY`/`ALGN` job above. The file is available at:

`$SCHRODINGER/macromodel-v`*version*`/samples/Examples/`
`dist_algn_embrace_mcmm.com`

The descriptions of most of the opcodes may be read from the corresponding opcodes in the MCMM conformational search example given above, in Section 13.4.2.1 on page 134.

```
dist_algn_embrace_mcmm.mae
dist_algn_embrace_mcmm-out.mae
# arg2 of NPRC must be a multiple of 4 since each search is seeded by 4 structures
 NPRC       2      4      0      0     0.0000     0.0000     0.0000     0.0000
 FFLD      11      1      0      0     1.0000     0.0000     0.0000     0.0000
 SOLV       3      1      0      0     0.0000     0.0000     0.0000     0.0000
 EXNB       0      0      0      0     0.0000     0.0000     0.0000     0.0000
 BDCO       0      0      0      0    89.4427 99999.0000     0.0000     0.0000
 MSYM       0      0      0      0     0.0000     0.0000     0.0000     0.0000
#Turn on MBAE Conformational Searching (arg2 = 1)
# in Energy Difference mode (arg1 = 1)
 MBAE       1      1      1      0     0.0000     0.0000     0.0000     0.0000
#Each search will have 12 steps in it.
 MCMM      12      0      0      0     0.0000     0.0000     0.0000     0.0000
#Change a relatively small number of torsions (between 1 and 4) in each move.
 MCNV       1      4      0      0     0.0000     0.0000     0.0000     0.0000
 MCSS       2      0      0      0   500.0000     0.0000     0.0000     0.0000
#For MCOP
# arg1 = 1 means printout something for each conformation generated
# arg3 is set to a large value to avoid spurious lines
#      in the table at the end of the log file.
# arg6 = 1.0 means save only the lowest energy receptor structure.
```

```
 MCOP       1      0  10000      0     0.0000     1.0000     0.0000     0.0000
 DEMX       0      0      0      0  1000.0000     0.0000     0.0000     0.0000
# Note that unlike the combined COPY/ALGN-MBAE job SUBS arg2 is 0 since we
# have not issued a RWND command
 SUBS       0      0      0      0     0.0000     0.0000     0.0000     0.0000
 READ       0      0      0      0     0.0000     0.0000     0.0000     0.0000
#Use AUTO to set up the MCMM conformational search but arg6 must be 0.0
# or absent because the search of the receptor is not regarded as
# part of a serial search.
 AUTO       0      0      0      0     0.0000     0.0000     0.0000     0.0000
 MINI       1      0   2000      0     0.0000     0.0000     0.0000     0.0000
#Prepare for searches of ligands and complexes.
#Use AUTO to set up the MCMM conformational search but arg6 must be 1.0
# because the searches for each ligand and each ligand-receptor complex
# is regarded as part of a serial search.
 AUTO       0      0      0      0     0.0000     1.0000     0.0000     0.0000
# Optional search command for ligands and complexes.
# Each search will have 12 steps in it (it may differ from the previous one).
 MCMM      12      0      0      0     0.0000     0.0000     0.0000     0.0000
 BGIN       0      0      0      0     0.0000     0.0000     0.0000     0.0000
#For MCOP
# arg1 = 1 means printout something for each conformation generated
# arg3 is set to a large value to avoid spurious lines
#      in the table at the end of the log file.
# arg6 = 2.0 means save structures for the 2 lowest energy conformers
#      for each complex.
# arg7 = 4.0 means use the 4 input poses from the COPY/ALGN proceedure
#        to seed the search for each ligand.
 MCOP       1      0  10000      0     0.0000     2.0000     4.0000     0.0000
 READ       0      0      0      0     0.0000     0.0000     0.0000     0.0000
 MINI       1      0   2000      0     0.0000     0.0000     0.0000     0.0000
 END        0      0      0      0     0.0000     0.0000     0.0000     0.0000
#turn MBAE off.
 MBAE      -1      0      0      0     0.0000     0.0000     0.0000     0.0000
```

`NPRC`: Distribute the calculation. Arg1 is the number of processors to use and arg2 is the number of input structures to process in each subjob. Arg2 should be a multiple of 4 for ligands prepared with `ALGN` so that all four alignments of each ligand are sent to the same child process.

Free Energy Simulations

This chapter provides some of the theory behind the powerful simulation method of free energy perturbation calculations, and describes how to perform a free energy perturbation with MacroModel. If you intend to perform free energy perturbations, we strongly recommend that you become familiar with the current literature on this topic (see Section 14.5 on page 156).

Note: The MM2* and MM3* force fields should not be used for Free Energy Perturbation.

14.1 Free Energy Perturbation

The fundamental expression for free energy calculations as implemented in MacroModel is:

$$G_B - G_A = \Delta G = -RT\ln\left\langle \exp\left[\frac{-(H_B - H_A)}{RT}\right]\right\rangle_A \tag{1}$$

where H_A and H_B are the Hamiltonians for the two systems, A and B, and the <> notation represents an ensemble average over system A. This expression is valid only when there is a very small difference between the two systems, A and B (i.e., H_B–H_A ≈ RT). To perform free energy calculations on meaningful systems, you generally perform a series of smaller simulations (windows), which can be summed to obtain a total free energy difference.

The coupling parameter <λ> is used to define each window in terms of the two endpoints, A and B, so that at any stage the Hamiltonian over which the ensemble average will be generated for the system is described in terms of the Hamiltonians for the endpoints:

$$H(\lambda) = \lambda H_B + (1-\lambda)H_A \tag{2}$$

Then the overall expression for performing a simulation becomes:

$$\Delta G = \sum_{i=0}^{n} -RT\ln\left\langle \exp\left[\frac{-(H_{\lambda_{i+1}} - H_{\lambda_i})}{RT}\right]\right\rangle_{\lambda_i} \tag{3}$$

where n is the number of windows to be used in the simulation. At any window the “perturbation” is between a state at λ and that at λ + dλ. The method for performing this simulation as implemented in MacroModel is described as “single topology.” That is, at any value of λ, a set of interactions is generated by mixing the interactions of the endpoints. For example, if an atom has a charge x in the starting point of the simulation and a charge y in the end point, at any given value of λ (i.e., at any window), the charge on the atom will be $z = y\lambda + (1-\lambda)x$. This process will be repeated for all the interactions in the system and the simulation will be performed with this new set of interactions. At points during the simulation, the energy, H(λ), will be evaluated. Then the interactions corresponding to the "next" value of lambda (λ+dλ) will be generated and H(λ+dλ) will be calculated. These terms are used in the exponential of expression (3). In MacroModel we usually take advantage of the fact that for most values of λ, the energy at λ-dλ can also be evaluated—this procedure is known as "double wide sampling," and effectively allows a "forward" and "reverse" simulation to be performed in one simulation.

14.2 Setting Up FEP Calculations

Currently free energy calculations can be set up only by direct manipulation of the MacroModel command files. There is no Maestro interface for this procedure. The individual MacroModel opcodes are described in detail in Chapter 4 of the *MacroModel Reference Manual*. This section illustrates how to combine these opcodes and produce a sensible command file. To do this we will use a simple example—a calculation of the free energy difference between D and L forms of the "alanine dipeptide"—this should give a free energy difference of zero in isolation. Though this is not a very useful simulation, it illustrates free energy perturbation calculations. Note that most of the perturbations we have tested involve small perturbations with changes of up to only three atoms.

14.2.1 The Structure File

When you perform a free energy perturbation calculation in MacroModel, the input structure file (`.mae`) must contain two separate MacroModel structures. These correspond to the start and end point for the calculation. The numbering of the structures is critical—all "equivalent" atoms must have the same number in each structure. Dummy atoms (MacroModel atom type 61) must be used as place holders for atoms created or destroyed during the perturbation. The numbering of the structures used in our example is shown in Figure 14.1.

Here we are performing a FEP calculation between the L and D enantiomers of the alanine dipeptide using the AMBER* force field, so we have no explicit hydrogen on the alpha carbon. The perturbation involves changing dummy atom 13 in the starting (L) structure to a united atom methyl in the (D) structure. At the same time, atom number 9, which begins as a united atom methyl in the starting structure, is extinguished to a dummy atom in the structure at the end of the simulation. Numbering of the structures must be checked very carefully before a free energy simulation is attempted.

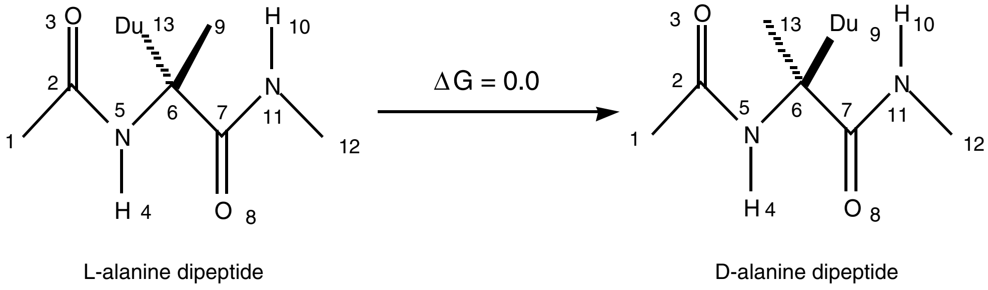


Figure 14.1. ***Numbering system for the alanine dipeptide.***

14.2.2 The Command File

The command file used to perform the perturbation of the L to the D form is shown below.

```
fep.mae
fep-out.mae
 EXNB      0      0      0      0     0.0000     0.0000     0.0000     0.0000
 BDCO      0      0      0      0     0.0000     0.0000     0.0000     0.0000
 FFLD      3      0      0      0     0.0000     0.0000     0.0000     0.0000
 READ      0      0      0      0     0.0000     0.0000     0.0000     0.0000
 FEIA      0      0      0      0     0.0000     0.0000     0.0000     0.0000
 MCNV      1      2      0      0     0.0000     0.0000     0.0000     0.0000
 MCSD      1      2      0      0     0.0000     0.0000   300.0000     0.0000
 TORS      5      6      0      0    30.0000   180.0000     0.0000     0.0000
 TORS      6      7      0      0    30.0000   180.0000     0.0000     0.0000
 BGIN      0      0      0      0     0.0000     0.0000     0.0000     0.0000
 FEAV      0      0      0      0     0.0000     0.0000     0.0000     0.0000
 MINI      1      0   1000      0     0.0000     0.0000     0.0000     0.0000
 MDIT      0      0      0      0   300.0000     0.0000     0.0000     0.0000
 MDYN      0      0      1      0     1.5000    10.0000   300.0000     0.0000
 FESA      0      0      0      0     0.0000     0.0500     0.0000     0.0000
 MDYN      0      0      1      0     1.5000   100.0000   300.0000     0.0000
 END       0      0      0      0     0.0000     0.0000     0.0000     0.0000
 FESU      0      0      0      0     0.0000     0.0000     0.0000     0.0000
```

`EXNB`: Extended non-bonded cutoffs should be used in free energy calculations.

`FFLD`: This example used the AMBER* force field with the default distance-dependent dielectric treatment for electrostatics. We also recommend the use of the GB/SA solvation model in a real application.

`READ`: Required to read the first structure in the `.mae` file.

`FEIA`: Reads the second structure (the perturbation endpoint) from the `.mae` file and sets up the arrays for averaging of the interactions.

`MCNV`: Specifies that between 1 and 2 of the torsions specified by `TORS` commands will be varied at each MC step.

`MCSD`: In this example we use mixed-mode Monte Carlo/Stochastic Dynamics to perform the simulation. Because a successful FEP simulation requires a thorough sampling of the conformation space at each window, we strongly recommend using the mixed-mode simulation method during free energy calculations.

`TORS`: Defines the torsion angles used in the Monte Carlo part of the `MCSD` calculation. In this case it is the ϕ and ψ angles of the dipeptide.

`BGIN`: Free energy perturbation calculations are usually performed in a `BGIN/END` loop, which completes all the windows from λ=0 to λ=1.

`FEAV`: Generates the interactions corresponding to the value of λ given in arg5. The interpretation of this command can lead to some confusion. The first time it is encountered, it will generate an interaction array with the value of lambda as given in arg5. When placed inside a `BGIN/END` pair, as in this example, the behavior is slightly different—when subsequently encountered, the `FEAV` command will increment the current value of λ (i.e., the middle of the double wide sampling window) by the difference between arg5 and arg6 in the following `FESA` command. In this case it will be incremented to 0.05 the first time through the `BGIN/END` loop.

`MINI`: Once the interactions are generated, it is necessary to minimize the structure with this set of interactions in order to remove any excess potential energy before beginning the dynamics part of the simulation.

`MDIT`: Sets initial random velocities corresponding to an initial temperature of the value of arg5 (in this case, 300° K).

`MDYN`: At any given value of l (i.e., at the start of every window) it is good practice to first perform a short equilibration simulation before collecting data for the free energy calculation.

`FESA`: Begins the collection of free energy samples for the current value of (λ)—the middle of the window. The values of arg5 and arg6 indicate the values of the left side of the window and the right side of the window for double wide sampling. The difference between the values of arg5 and arg6 controls the number of windows which will be performed in the simulation—the relationship is #windows = 1/(arg6–arg5) + 1. A maximum of 101 windows can be used in a MacroModel FEP calculation so the minimum permissible value of the difference between arg5 and arg6 is 0.01. When used inside a `BGIN/END` pair, as in this example, arg5 and arg6 will be incremented each pass through the loop, as illustrated in the table below.

Most simulations require at least 20 windows to ensure that the free energy difference per window (which must be <3 RT) remains sufficiently small.

Pass	λ(left)	λ(middle)	λ(right)	
1	0.00	0.00	0.05	(no middle->left sampling)
2	0.0	0.05	0.10	(middle->left and left->right)
3	0.05	0.10	0.15	(middle->left and left->right)
4	0.10	0.15	0.20	(middle->left and left->right)
20	0.90	0.95	1.00	(middle->left and left->right)
21	0.95	1.0	1.0	(no middle->right sampling)

`MDYN`: Describes the dynamics simulation in which the free energy sampling is done. In this simple case we are only doing 100 ps of mixed-mode simulation at each window. In real cases > 500 ps simulation per window will probably be required.

`END`: The end of the `BGIN/END` loop. The loop terminates when the value of λ (middle) exceeds 1.0.

`FESU`: Prepare a summary of the free energy change during the simulation.

14.2.3 Other Possible Command Files

Performing specific windows:

Although it is useful to wrap free energy calculations in a BGIN/END loop and simulate with all windows, it is possible to run any specific windows. The windows are independent and free energy differences obtained from separate runs can be usefully combined. It is also sometimes useful to re-run specific windows from a simulation to obtain better statistics. The command file below performs double wide sampling for two windows with λ(middle)=0.5 and λ(middle)=0.55:

```
fep-diala.mae
fep-diala-out.mae
 EXNB       0      0      0      0     0.0000     0.0000     0.0000     0.0000
 BDCO       0      0      0      0     0.0000     0.0000     0.0000     0.0000
 FFLD       3      0      0      0     0.0000     0.0000     0.0000     0.0000
 READ       0      0      0      0     0.0000     0.0000     0.0000     0.0000
 FEIA       0      0      0      0     0.0000     0.0000     0.0000     0.0000
 MCNV       1      2      0      0     0.0000     0.0000     0.0000     0.0000
 MCSD       1      2      0      0     0.0000     0.0000   300.0000     0.0000
 TORS       5      6      0      0    30.0000   180.0000     0.0000     0.0000
 TORS       6      7      0      0    30.0000   180.0000     0.0000     0.0000
 FEAV       0      0      0      0     0.5000     0.0000     0.0000     0.0000
 MINI       1      0   1000      0     0.0000     0.0000     0.0000     0.0000
```

```
MDIT       0      0      0      0   300.0000     0.0000     0.0000     0.0000
MDYN       0      0      1      0     1.5000    10.0000   300.0000     0.0000
FESA       0      0      0      0     0.4500     0.5500     0.0000     0.0000
MDYN       0      0      1      0     1.5000   100.0000   300.0000     0.0000
FEAV       0      0      0      0     0.5500     0.0000     0.0000     0.0000
MINI       1      0   1000      0     0.0000     0.0000     0.0000     0.0000
MDIT       0      0      0      0   300.0000     0.0000     0.0000     0.0000
MDYN       0      0      1      0     1.5000    10.0000   300.0000     0.0000
FESA       0      0      0      0     0.5000     0.6000     0.0000     0.0000
MDYN       0      0      1      0     1.5000   100.0000   300.0000     0.0000
FESU       0      0      0      0     0.0000     0.0000     0.0000     0.0000
```

Using the distributed MacroModel procedure to perform a free energy calculation:

For more information about distributed MacroModel, see Section 3.3 on page 28. The `NPRC` command can be used to distribute the free energy calculation over a number of hosts in a manner similar to that described for Monte Carlo conformational searching. In this case the smallest job that can be distributed on any host is one complete FEP window, so the opportunities for load balancing are limited.

14.2.4 The Output File

Similar output for each window of the perturbation will be obtained in the `.log` file. The output obtained when the command file described above was run appears below. This output is for the value of λ(middle)=0.4

```
FEP averaged force field, lambda =  0.400000
Loading FEP coordinates 1
Starting conjugate gradient minimization.
Minimization converged; gradient = 0.335E-01 .LT.  0.500E-01
Iterations =  120  out of   1000
Conf    1 E =  -138.502 (  0.034) kJ/mol

BatchMin V8.5    Stochastic Dynamics Simulation
               MC/SD Mixed Mode

    Time       Total E     0.5ps T         Ave H       Ave T    Ave H( 300)
    (ps)      (kJ/mol)     (deg K)       (kJ/mol)     (deg K)      (kJ/mol)
     1.000       -60.8       297.1          -97.2       295.4         -96.5
     2.001       -72.2       172.3         -103.0       260.1         -96.5
     3.000       -45.2       263.5         -102.1       265.5         -96.5
     4.001       -62.5       308.8         -101.7       272.9         -97.3
     5.001       -70.4       252.1         -101.4       269.1         -96.4
     6.000       -73.9       272.6         -102.2       267.2         -96.9
     7.001       -51.8       276.7         -101.6       265.1         -96.0
     8.001       -60.0       271.4         -101.2       266.0         -95.7
     9.000       -55.4       266.2         -101.5       268.3         -96.4
    10.000       -54.3       261.5         -101.0       268.6         -95.9

Final potential and kinetic energy =      -93.46       39.77 kJ/mol
CPU Time =      1.2 sec

Average potential energy =     -101.02 kJ/mol
```

```
Acceptance Ratio for Mixed Mode Simulation   =  0.02744863
Average kinetic energy   =       43.55 kJ/mol (Av temperature =  268.6 deg K)
Average total energy     =      -57.47 kJ/mol (Std dev =   11.13 kJ/mol)

Average potential energy <H> scaled to 300.0 deg K =      -95.92 kJ/mol
                                   Av stretch             12.72 kJ/mol
                                   Av bend                15.59 kJ/mol
                                   Av torsion             25.90 kJ/mol
                                   Av van der Waals       12.57 kJ/mol
                                   Av electrostatic      -162.70 kJ/mol
                                   Av solvation 1          0.00 kJ/mol
                                   Av solvation 2          0.00 kJ/mol

BatchMin V8.5    Stochastic Dynamics Simulation
               MC/SD Mixed Mode

Free energy perturbation calculation
Averaged interaction array, lambda =  0.400000


FEP Window  9:
    Time   Total E   0.5ps T  Average T  <G>0.35-0.40  <G>0.45-0.40
    (ps)  (kJ/mol)   (deg K)    (deg K)      (kJ/mol)      (kJ/mol)
    10.000     -35.5     306.2      284.0        -1.11          0.89
    20.001     -46.5     294.7      304.7        -1.16          0.98
    30.000     -19.9     292.4      302.6        -1.04          0.82
    40.000     -24.2     366.7      301.7        -0.98          0.77
    50.001     -57.1     282.4      300.0        -0.95          0.72
    60.000     -24.3     317.0      299.3        -0.94          0.70
    70.001     -24.6     265.0      302.7        -0.96          0.72
    80.001     -31.0     310.1      301.0        -0.94          0.71
    90.002     -31.1     287.6      300.3        -0.95          0.71
   100.001     -59.1     277.9      298.9        -0.97          0.74

Final potential and kinetic energy =      -84.41       38.78 kJ/mol
CPU Time =     12.5 sec

Average potential energy =      -96.48 kJ/mol

Acceptance Ratio for Mixed Mode Simulation   =  0.03341983
Average kinetic energy   =       48.46 kJ/mol (Av temperature =  298.9 deg K)
Average total energy     =      -48.02 kJ/mol (Std dev =   15.55 kJ/mol)

Average potential energy <H> scaled to 300.0 deg K =      -96.29 kJ/mol
                                   Av stretch             15.40 kJ/mol
                                   Av bend                17.81 kJ/mol
                                   Av torsion             27.36 kJ/mol
                                   Av van der Waals       13.13 kJ/mol
                                   Av electrostatic      -169.99 kJ/mol
                                   Av solvation 1          0.00 kJ/mol
                                   Av solvation 2          0.00 kJ/mol

Free energy perturbation window   9
  Lambda (left,center,right) =   0.350   0.400   0.450
 Number of samples in average =        3921
                     Bonded Nonbonded Solvation     Total       +/-
 G( left-center) =    -0.094    -0.867     0.000    -0.967     0.134 kJ/mol
 G(right-center) =     0.059     0.694     0.000     0.737     0.166 kJ/mol
```

Similar output was obtained for the other 20 windows of this simulation. The first part of the output shows the result of the minimization at this value of λ and the equilibration simulation. Then the actual free energy sampling is performed and a free energy for the “forward” and "reverse" simulation is reported. Notice that free energy reported should become stable to a few hundredths of a kJ/mol near the end of the sampling period. Note, however, that this is a necessary but insufficient indicator of convergence during the simulation, as this may only indicate that the simulation is stuck in one of a number of accessible potential energy minima.

After the actual sampling phase is over, a summary of the free energy change for this window will be printed as shown above. *Note that the "component" (e.g., Bonded, Nonbonded, and Solvation) free energies are only approximate and do not necessarily sum to the "Total" value.* After all windows are completed, a summary is printed of the perturbation over all windows (with the "forward" and "reverse" simulations shown side by side). At the end of the file is a summary of the complete perturbation:

```
Summary of total perturbation. Lambda: 0.000 -> 1.000
                           Forward     Reverse     Average
        G(bonded)    =     0.10200     0.07908     0.09054 kJ/mol
        G(nonbonded) =     1.08893    -1.12791    -0.01949 kJ/mol
        G(solvation) =     0.00000     0.00000     0.00000 kJ/mol
        G(total)     =     1.03701    -0.90759     0.06471 kJ/mol
        Standard dev =     0.31498     0.29871             kJ/mol
```

This is probably the most important part of the output. The average of the "forward" and "reverse" total free energies gives an estimate of the free energy change for the entire simulation. In this case it is close to zero, as expected (sampling for 500 ps per window reduces the value to 0.02 kJ/mol). The standard deviation of the free energy is useful as an estimate of the error. In this case the free energy would be reported as 0.1±0.3 kJ/mol. At 500 ps sampling it is 0.02±0.14 kJ/mol. Note that we use the larger of the "forward" or "reverse" standard deviation values and not the average. As mentioned above, only the "total" free energies are strictly meaningful—*the components are only estimates*. In addition to examining the final free energy summary, it is useful to plot the progress of the perturbation.

There should be a continuous, smooth change in free energy from window to window with no large "jumps" or sudden increases in the standard deviation. If discontinuities are observed, then the simulation should probably be run with more sampling per window and possibly more windows.

14.3 Criteria for a Successful Simulation

The answers to the following questions will help you perform successful simulations.

1. Do the input structures have the same numbering with dummy atoms used for atoms which are present in one structure and not in another?

One useful way to check the input file is to set up a simple command file as follows:

```
fep-diala.mae
fep-diala-out.mae
 EXNB       0      0      0      0     0.0000     0.0000     0.0000     0.0000
 BDCO       0      0      0      0     0.0000     0.0000     0.0000     0.0000
 FFLD       3      0      0      0     0.0000     0.0000     0.0000     0.0000
 READ       0      0      0      0     0.0000     0.0000     0.0000     0.0000
 FEIA       0      0      0      0     0.0000     0.0000     0.0000     0.0000
 FEAV       0      0      0      0     0.0000     0.0000     0.0000     0.0000
 MINI       1      0   1000      0     0.0000     0.0000     0.0000     0.0000
 FEAV       0      0      0      0     1.0000     0.0000     0.0000     0.0000
 MINI       1      0   1000      0     0.0000     0.0000     0.0000     0.0000
```

This command file performs a minimization with $\lambda = 0$ and $\lambda = 1$. Record the final minimized energies in each case. Now swap the structures in the input (`.mae`) file and re-run the calculation. For the second run, with the order of the structures reversed, the energy at the $\lambda = 1$ point should be the same as the energy from the $\lambda = 0$ point of the first run. Similarly, the energy of the $\lambda = 0$ point in the second run should be the same as the $\lambda = 1$ energy of the first. If these do not match exactly, then there is likely to be something wrong with the numbering of the input structure files.

2. Is sufficient sampling being performed?

While the expression (1) described above is exact, it is implicit that at each window complete sampling of all important conformations is achieved. In practice this is very difficult to achieve and we strongly recommend the use of the mixed-mode simulation methodology to help solve the sampling problem. Even so, you should also perform independent tests for convergence with simulations at least as long as the sampling period in the free energy calculations. Ideally you should be satisfied that you can achieve a reasonable degree of convergence for simulations of the starting structures in the length of time used for the sampling in the free energy calculations. Good tests for convergence are starting from different starting geometries and (for non-chiral molecules) monitoring the populations of “equivalent” conformations (e.g., +gauche/-gauche torsion angles). Another possibility is to compare structures sampled from a simulation with those obtained from an extensive conformational search. Most importantly note that stable values for the average free energy change over several tens of picoseconds of sampling, while required for a converged simulation, are not alone proof of convergence. If the simulation is “stuck” in one particular minimum, then the average quantities will look to be converged but the actual free energy may be incorrect.

3. Is the force-field adequate?

Free energy perturbation assumes that the potential energy of the system is well described by force field parameters. You cannot expect to get quantitative agreement with experimental free energies (or make useful predictions) unless you are using a high quality force field. You should carefully check the quality of all the parameters in use in a free energy calculations. A summary of the parameter quality is printed at the top of the `.log` file.

4. Are the dummy atom parameters appropriate?

Natural bond lengths to dummy atoms may need to be set to approximate the length of the real counterpart of the dummy atom, e.g., the atom into which the dummy “grows.” In the past we have used very short (ca. 0.5 Å) natural bond lengths for dummy atoms. However, this can cause problems when the dummy atom has significant non-bonded interaction with other parts of the molecule early in the simulation. For example, in the case of perturbing axial methyl cyclohexane to equatorial methyl cyclohexane using the united atom model for AMBER*, if the dummy atoms start at 0.5 Å the free energy difference between the two conformations is greatly overestimated. Setting the dummy atom natural bond length to ca. 1.5 Å gives a much more reasonable free energy difference. Unfortunately there is not one “perfect” length for all systems, and you should experiment with the dummy atom natural bond length to ascertain that the results obtained are independent of the dummy atom parameters.

14.4 Other Types of Free Energy Calculations

Free energy calculations have been possible for some time—most of the simulation methods in MacroModel result in ensembles that actually sample the free energy surface of the molecule. In some cases actual values for free energy differences between well-defined conformational states can be obtained. This is best illustrated with two examples from our own laboratory.

14.4.1 Hydrogen Bonding Preference of a Glycyl Lactam in Organic Solution

Infrared studies of the glycyl lactam in CH_2Cl_2 [30] suggest that both hydrogen bonded and non-hydrogen bonded forms are significantly populated at room temperature. In this case conformational searching (which can be considered an “Enthalpy” at 0°K) was apparently at variance with experiment because the global minimum, a hydrogen bonded form, is 2.7 kcal/mol lower in energy than the most stable non-hydrogen bonded energy minimum. When molecular dynamics simulations were performed at 300°K [31] and the hydrogen bond population was monitored during the simulation, we found that only 56% of the conformations could be considered hydrogen bonding at this temperature. Thus the hydrogen bonded populations reflect the experimentally observed populations because the simulation is exploring the “free energy surface” of the molecule. The “minima” obtained from the conformational search are the lowest points on very broad energy wells, which include many states lacking well-defined hydrogen bonds. So even using a high quality force field and solvation model was not sufficient to reproduce experiment without the use of the appropriate simulation methodology.

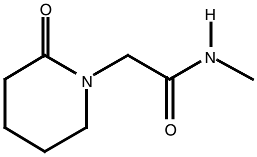


Figure 14.2. The glycyl lactam structure.

14.4.2 Conformational Free Energies for a Diamide in Solution

This compound was one of a series studied by Gellman et al. [32] using variable temperature IR and NMR in organic solvent. If we consider that there are two states of the molecule (hydrogen bound and non-hydrogen bound, each representing many possible conformations), we can perform a series of simulations at different temperatures in the range 200-300°K and perform a van't Hoff analysis on the results, which is directly comparable to that reported from the experiments [33]. The results of this analysis are shown in the following figure and table:

ΔH (kcal/mol)				**ΔS (e.u.)**		
N	^{1}H NMR	IR	Simulation	^{1}H NMR	IR	Simulation
4	–1.5	–1.4	–1.65	–8.0	–6.8	–8.6

In this system, where there are two well-defined states, we have been able to obtain thermodynamic parameters (which agree well with experiment) from direct simulation without using any perturbation techniques.

The reason that both of these simulations were able to obtain quantitative agreement with experiment is that we closely followed this procedure:

- Careful parametrization of the gas-phase conformational energy differences against ab initio calculations at the 6-31G* level.
- Use of the GB/SA solvent model.

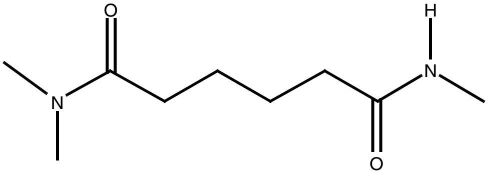


Figure 14.3. The diamide structure.

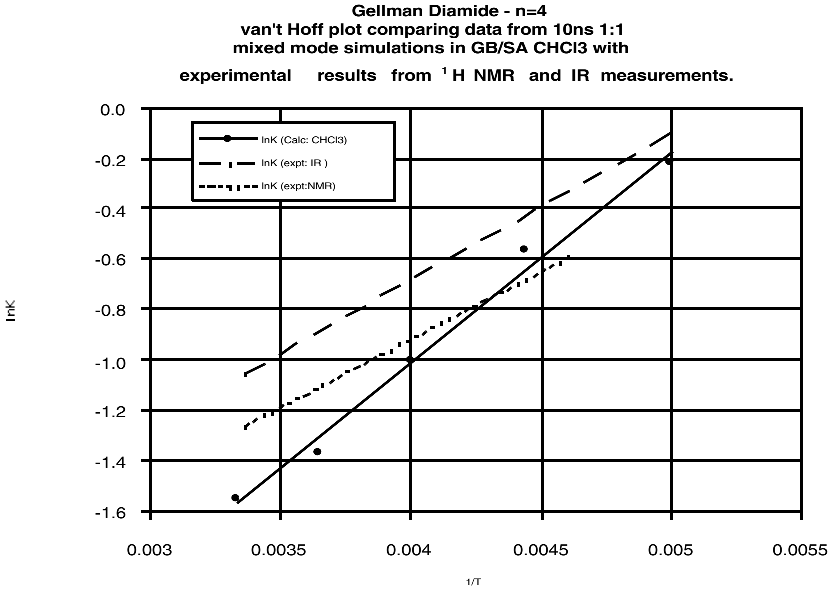


Figure 14.4. van't Hoff analysis of the diamide.

- Assuring converged simulations by long runs (ns time scale) using the mixed-mode MC/SD procedure. We were also careful to test for convergence by ensuring that we could obtained the same results from different starting conformations.

14.5 Literature

This section is not an exhaustive review of the free energy literature, but rather a set of representative reviews and papers covering both theory and applications of free energy calculations. Those marked with a "*" are particularly important for their critical analysis of the potential problems of free energy calculations, and we urge you to read and understand these reviews before attempting free energy calculations.

Kollman, P. A. *Chem Rev.* **1993,** *93,* 2395.

Beveridge, D. L.; DiCapua, F. M. *Annu. Rev. Biophys. Biophys. Chem.* **1989,** *18,* 431.

Jorgensen, W. L. *Acc. Chem. Res.* **1989,** *22,* 184.

Jorgensen, W. L. *J. Am. Chem. Soc.* **1989,** *111,* 755.

Cieplak, P.; Kollman, P. A. *J. Comput.-Aided Mol. Des.* **1993,** *7,* 291.

* van Gunsteren, W. F., Weiner, P. K., Eds. *Computer Simulations of Biomolecular Systems;* ESCOM: Leiden, 1990.

* van Gunsteren, W. F, Mark, A. E. *Eur. J. Biochem*. **1992,** *204,* 947.

* Mark, A. E.; van Helden, S. P.; Smith, P. E.; Janssen, L. H. M.; van Gunsteren, W. F. *J. Am. Chem. Soc*. **1994,** *116,* 6293.

* Mitchell, M. J.; McCammon, J. A. *J. Comput. Chem.* **1991,** *12,* 271.

Smith, P. E.; van Gunsteren, W. F. *J. Phys. Chem.* **1994,** *98,* 13735.

Redundant Conformer Elimination

Collections of conformers can contain structures that are essentially the same, based upon energetic or geometric considerations. The redundant conformer elimination facility allows you to use MacroModel to remove the extra conformers rapidly without reminimizing or re-evaluating the energy.

One important use for this facility is in the complementary use of MacroModel and Jaguar to conduct a very high quality conformational search. In this approach, MacroModel is used to perform the initial conformational search, typically using GB/SA solvation. To increase the accuracy of the geometries and energies of the structures produced, Jaguar is used to reminimize the collection of conformers. Sometimes the molecular mechanics potential energy surface differs enough from the ab initio potential energy surface that structures considered distinct in the original conformational search minimize to the same structure in the Jaguar calculations. The redundant conformer elimination facility can then be used to remove these redundant conformers without adjusting the energy or geometry of the molecules.

Another use of the redundant conformer elimination facility is to reprocess the results of a MacroModel conformational search with less strict criteria for distinguishing conformers or a smaller energy range for retained conformers. As well, on occasions MacroModel conformational searches might not eliminate all of the redundant conformers. Running the structures through this facility provides a rapid mechanism for dealing with such cases.

15.1 Eliminating Redundant Conformers Using Maestro

You can set up and run jobs to eliminate redundant conformers from a set of conformers with the Redundant Conformer Elimination panel. To open this panel, choose Redundant Conformer Elimination from the MacroModel submenu of the Applications menu in the main menu bar. The upper and lower parts of the panel are common to all MacroModel panels. These components are described in detail in Section 4.1 on page 31. Redundant conformer elimination performs no energy calculations, so the tabs present in other MacroModel panels are not present. The remaining controls in this panel are for selecting atoms and settings for structural comparison.

Define comparison atoms

You can define comparison atoms by picking atoms in one of the conformers, which must be included in the Workspace, or by selecting one of the predefined sets of comparison atoms: Heavy Atoms + O-H, S-H or Heavy Atoms. Heavy atoms are defined as non-hydrogen atoms.

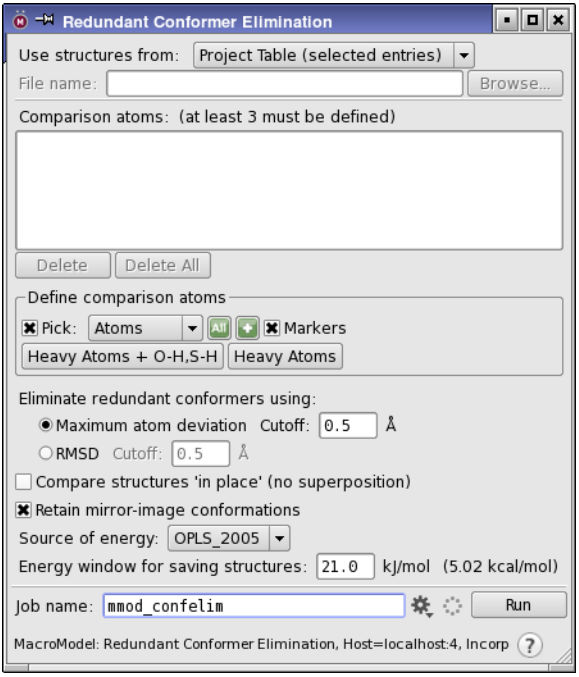
The first option of the two predefined sets includes hydrogen atoms attached to O and S atoms. The comparison atoms are listed as atom numbers in the Comparison atoms list. To delete one or more atoms from the list, select them in the list and click Delete. You can delete all comparison atoms from the list by clicking Delete All.

Eliminate redundant conformers using

If comparison atoms are chosen or Perform automatic setup during calculation is selected, structures produced during a conformational search are compared to see if they are unique. Two options are available for the conditions for structures to be considered different:

- Maximum atom deviation—consider structures to be different if the maximum atom deviation for any atom exceeds the threshold given in the Cutoff text box.
- RMSD—consider structures to be different if the RMS deviation for all compared atoms exceeds the threshold given in the Cutoff text box.

The default cutoff is 0.5 Å for both options.



Use structures from: Project Table (selected entries)
File name:
Comparison atoms: (at least 3 must be defined)
☒ Pick: Atoms ☒ Markers
Eliminate redundant conformers using:
● Maximum atom deviation Cutoff: 0.5 Å
○ RMSD Cutoff: 0.5 Å
☐ Compare structures 'in place' (no superposition)
☒ Retain mirror-image conformations
Source of energy: OPLS_2005
Energy window for saving structures: 21.0 kJ/mol (5.02 kcal/mol)
Job name: mmod_confelim
MacroModel: Redundant Conformer Elimination, Host=localhost:4, Incorp

Figure 15.1. The Redundant Conformer Elimination panel.

Compare structures 'in place' (no superposition)

If you select this option, the conformers are not superimposed (translated and rotated to obtain the closest alignment of atoms) in the process of comparing the structures, but the comparison is made with the structures in their given locations. By default, the structures are superimposed. This option could be used, for example, for Glide poses, where the position of the ligands with respect to the receptor is important.

Retain mirror-image conformations

By default, both a structure and its mirror image are considered when comparing with the other structures. If one of the two matches, the structure is eliminated. To ensure that both the structure and its mirror image are kept, select this option.

Source of energy

If you choose to include the energy as a criterion for comparison, structures whose energy differs by more than 4.184 kJ/mol (1 kcal/mol) are considered inequivalent. This test is applied before the geometries are compared and can considerably speed up the comparison. You can choose from the Jaguar energy or the energies for various force fields supplied with MacroModel, or you can disable energy comparison by selecting None. If the particular energy you selected is not available among the properties of the structure, the calculation stops. You should therefore choose an energy that has been calculated for all structures.

Energy window for saving structures

This is the threshold value for comparison of structures. Structures are kept only if their energy is less than this value above the current global minimum. Lowering this value results in fewer structures saved. The default value is 21 kJ/mol.

15.2 Command File Examples

To run MacroModel calculations, a molecular structure file and a command file are required. The molecular structure file contains the structures to be used as input in the calculation. The command file contains the name of the input structure file, the name of the output structure file, and an ordered list of operation codes (*opcodes)* for the calculations.

When you start a job from Maestro or write out the input files, Maestro writes out a molecular structure file and a command file. For many types of jobs, command files written this way are complete and adequate. For some types of jobs, however, you may need to adjust the Maestro-generated command file.

Redundant conformer elimination is performed in MacroModel by the `ADDC` opcode. Each time `ADDC` is executed, it tries to add the current structure to the collection of output conformers. The same checks for redundancy and transformations, such as net translation and rotation of the current structure, are performed as in a multiple minimization of conformers (see `COMP`), but no minimization or energy estimation is conducted. Energies from earlier MacroModel or Jaguar calculations may be used as part of the comparison process (see `DEMX` in the *MacroModel Reference Manual*). If MacroModel energies are to be used, then a `FFLD` line with the same force field used to calculate the energies must precede the `ADDC` line. See the *MacroModel Reference Manual* for information on `ADDC`.

`ADDC` should be used much like `MINI` in a multiple minimization calculation. The command file below reads in conformers from the file `addc.mae`, orders them by increasing Jaguar energy, superimposes the structures on the lowest Jaguar energy structure, eliminates redundant conformers, and writes out the resulting structures to `addc-out.mae`.

```
addc.mae
addc-out.mae
 DEMX       0      0      0      0    10.0000    10.0000     0.0000     0.0000
 MSYM       1      0      0      0     0.0000     0.0000     0.0000     0.0000
 BGIN       0      0      0      0     0.0000     0.0000     0.0000     0.0000
 READ       0      0      0      0     0.0000     0.0000     0.0000     0.0000
 AUTO       1      0      0      0     0.0000     0.0000     0.0000     0.0000
 ADDC      -1      0      0      0     0.0000     0.0000     0.0000     0.0000
 END        0      0      0      0     0.0000     0.0000     0.0000     0.0000
```

`DEMX`: Arg5 sets the energy window to 10 kJ/mol.

`BGIN`: Begin the loop over conformers.

`READ`: Read the next conformer.

`AUTO`: Automatically set up comparisons. Arg1 = 1 sets up comparisons only once, and uses this set for the remaining conformers.

`ADDC`: Check the current conformer. Arg1 = –1 indicates that Jaguar energies should be present in the input structure file and that these energies are to be used in the process that eliminates redundant conformers.

Macrocycle Conformational Sampling

Macrocycles hold significant promise for certain drug projects, and performing conformational searches of macrocycles is an important component of a structure-based project. Macrocycle-containing molecules present challenges for molecular modeling because the rings and often the molecules themselves are atypically large for drug-like molecules. MacroModel's defaults were selected for searching typical drug-like molecules or dense protein-ligand complex environments, but is not necessarily well-suited for this task. To provide a more suitable strategy, many conformational searches of macrocyclic systems were performed, testing many variations on the search strategies available in MacroModel. The results of these tests were used to develop a Python script that is more suited to macrocycle conformational sampling than the standard conformational search algorithms.

The default protocol employed by this script uses a hybrid search method, employing 5000 cycles of MD simulated annealing followed by 5,000 cycles of LLMOD (large-scale low mode) search steps. The simulated annealing stage produces a set of nonredundant candidate structures that are used as seeds for the conformational search. Broad large-scale low-mode moves are allowed and the eigenvalues are calculated more frequently than in a default LLMOD calculation.

The script is available both from the command line and from Maestro.

16.1 The Macrocycle Conformational Sampling Panel

The Macrocycle Conformational Sampling panel allows you to run the sampling protocol and change the most important parameters of the protocol. Other parameters are available from the command line, but these should not usually need to be changed.

To open the Macrocycle Conformational Sampling panel, choose Applications → MacroModel → Macrocycle Conformational Sampling or Tasks → Conformational Search → Macrocycle Sampling.

The input structures can come from either the Project Table or from a file. You can choose from the range of force fields available in MacroModel, and select an electrostatic treatment. You can also set the number of heating/cooling cycles in the simulated annealing stage, and the number of search moves made in the low-scale low-mode conformational search. If you set either of these to zero, the stage is skipped: thus, you can perform just a simulated annealing calculation to generate conformers, or a large-scale low-mode search.

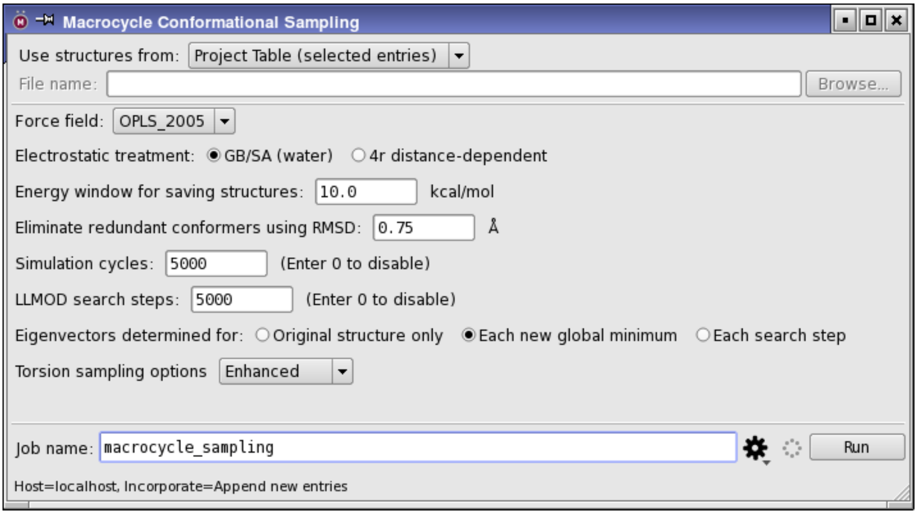
In the conformational search stage, you can control when the eigenvectors of the Hessian (used in the search moves) are recalculated, by selecting one of three options:

- Original structure only—Calculate the eigenvectors for the input structure only. This option saves time, but the results may be of lower quality.
- Each new global minimum—Calculate the eigenvectors each time a new global minimum is found. This ensures that the eigenvectors are updated for the most important conformers.
- Each search step—Calculate the eigenvectors for every new conformer that is found. This provides the highest accuracy in the eigenvectors, but is also the most time-consuming.

You can also control the automatic setup of torsional sampling around some bonds that generally have high barriers. These bonds are the C-N and C-O single bonds in groups like amides and esters, and C=N and N=N double bonds. The four levels of sampling are as follows:

- Restricted—Do not sample any of these bonds
- Intermediate—Sample C-N and C-O single bonds other than in standard amides and esters
- Enhanced—Sample all C-N and C-O single bonds
- Extended—Sample all C-N and C-O single bonds and C=N and N=N double bonds

Conformers that are too high in energy or that are redundant with other conformers can be eliminated by setting thresholds on the energy relative to the global minimum and the RMSD of a search result relative to any of the current set of conformers.



Use structures from: Project Table (selected entries)
File name:
Force field: OPLS_2005
Electrostatic treatment: ● GB/SA (water) ○ 4r distance-dependent
Energy window for saving structures: 10.0 kcal/mol
Eliminate redundant conformers using RMSD: 0.75 Å
Simulation cycles: 5000 (Enter 0 to disable)
LLMOD search steps: 5000 (Enter 0 to disable)
Eigenvectors determined for: ○ Original structure only ● Each new global minimum ○ Each search step
Torsion sampling options Enhanced
Job name: macrocycle_sampling
Host=localhost, Incorporate=Append new entries

Figure 16.1. The Macrocycle Conformational Sampling panel.

16.2 The macrocycle_conformational_sampling.py script

Macrocycle conformational sampling can be run using the Python script, with the full range of options for running the stages of the calculation. The syntax is:

`$SCHRODINGER/run macrocycle_conformational_sampling.py` *input-file* `[`*options*`]`

A search is performed for each compound in the input (Maestro) file. The task can be distributed over multiple processors, up to the number of compounds in the input file. The standard Job Control options, given in Table 2.1 of the *Job Control Guide*, are accepted, and the -WAIT, -LOCAL and `-NOJOBID` options given in Table 2.2 of the *Job Control Guide*. The script options are given in Table 16.1.

Table 16.1. Options for the macrocycle_conformational_sampling.py script.

Option	Description
`-v[ersion]`	Show the script version and exit.
`-h[elp]`	Show the usage message and exit.
`-f[fld] MM2|MM3|AMBER|AMBER94|OPLS|OPLS_2001|OPLS_2005|MMFF94|MMFF94s`	Force field. Default: `OPLS_2005`.
`-i|-sim_iterations` *sim-iter*	Number of simulated annealing iterations. Default: 5000.
`-t|-sim_time` *sim-time*	Length of each simulated annealing high-temperature sampling, in picoseconds. Default: 0.5 ps
`-sim_time_cool` *cool-time*	Length of each simulated annealing cooling phase, in picoseconds. Default: 0.5 ps
`-T|-sim_step` *sim-step*	Length of the simulation time step, in femtoseconds. Default: 3.0 fs.
`-K|-sim_temp` *temp*	Initial simulation temperature, in Kelvin. Default: 1000.0 K
`-s[olvation] WATER|OCTANOL|CHCL3|4RDDD|VACUUM`	Solvation model. `4RDDD` is the 4r distance-dependent dielectric model. Default: `WATER`.
`-I|-iterations` *steps*	Number of large-scale low-mode search moves to take. Default: 5000
`-iter_per_dof` *steps-per-dof*	Number of steps per degree of freedom to take. The conformational search takes this number of iterations if it is less than the total number. Default: 500.
`-m[in_move]` *min-move*	Minimum move size, in angstroms. Default: 3.0

Table 16.1. Options for the macrocycle_conformational_sampling.py script. (Continued)

Option	Description
`-M|-max_move` *max-move*	Maximum move size, in angstroms. Default: 18.0
`-eigen_recalc new_min|global_min|once`	Specify when to recalculate Hessian eigenvectors. `once`—only calculate eigenvectors for the input geometry. Faster but the results may be lower quality. `global_min`— recalculate when a new global minimum is found. `new_min`—recalculate for every new structure. Takes longer but the results may be higher quality. Default: `global_min`.
`-rmsd_cutoff` *cutoff*	RMSD cutoff, in angstroms, for eliminating redundant conformers. Default: 0.75
`-E|-energy_window` *window*	Energy cutoff, in kcal/mol, for saving structures. Conformers with energies higher than this value above the global minimum are discarded. Default: 10.0
`-compound_id` *property*	Property name that is used to distinguish chemical species. Default: `s_m_title`
`-p[lanar_torsion_sampling] restricted|intermediate|enhanced|extended`	Specify sampling restrictions about planar groups. `restricted`—do not sample around planar groups. intermediate—do not sample standard ester or amide linkages. `enhanced`— sample of amides and esters but not azo and imino. `extended`—sample all amide, ester, azo and imino linkages. Default: `enhanced`.
`-x|-alternate_sampling` *N*	Reverse the order of the stages: large-scale low-mode first, then simulated annealing of the lowest *N* conformers.
`-mixed_mode_ratio` *ratio*	Run mixed MCMM/LLMOD sampling, with this ratio of torsional to low-mode moves. Default: 0.0 (pure large-scale low mode).
`-auto_min_ring_size` *size*	Minimum ring size for automatically assigning ring closures and torsional sampling parameters. Relevant only when `-mixed_mode_ratio` is greater than zero. Default: 50 atoms, which skip most rings and focus torsional moves on the appendages to the macrocycle.
`-j|-JOBNAME` *jobname*	Optional base name for the job output files. Default: stem of the input file name.
`-g[ui]`	Open the Macrocycle Conformational Sampling panel as a standalone panel.

Additional Features

MacroModel has some additional features that are not available from the Maestro interface. Command file examples of some of these types of calculations are provided in this chapter. Note that some of these command files do not have the first `MMOD` line as seen in command files generated from the interface. The `MMOD` monitoring command is not necessary because these files cannot currently be monitored from Maestro.

17.1 Geometry Calculations

Geometric properties such as atom positions, distances, angles, and dihedrals can be measured using the `GEOM` opcode. Below is an example of the command file for a geometry calculation and explanations of the opcodes contained in the file.

```
geom.mae
geom-out.mae
 BGIN
 READ       0      0      0      0    0.0000    0.0000    0.0000    0.0000
 GEOM      14     15     16     17    0.0000    0.0000    0.0000    0.0000
 END
```

`BGIN`/`END`: Geometries can be calculated for a number of input structures within this loop.

`READ`: Read the input file.

`GEOM`: Use the number of non-zero arguments in arg1-arg4 to define whether atom coordinates, distances, bond angles, or dihedral angles should be calculated. For all of arg1-arg4 non-zero, the value of the dihedral angle defined by the atom numbers listed in arg1-arg4 is reported. Arg5 is used for calculation of spin-spin coupling constants. For details of arg1-arg5 of the `GEOM` opcode, see Section 4.15 of the *MacroModel Reference Manual*.

17.2 Calculating Interaction Energies Using ASET

`ASET` can be used to examine the interaction energies between sets of atoms within a structure. In the sample below, the interaction energy between LYS 33 and the ligand in the CDK2 structure 1e1v is computed. There is also a Python script available from the Script Center (http://www.schrodinger.com/scriptcenter) for ASET calculations. This script is installed by default on the Scripts menu (Scripts > Molecular Mechanics > Component Interactions).

```
aset.mae
aset-out.mae
 DEBG       1       0       0       0     0.0000     0.0000     0.0000     0.0000
 FFLD      11       1       0       0     1.0000     0.0000     0.0000     0.0000
 BDCO       0       0       0       0    89.4427 99999.0000     0.0000     0.0000
 ASET     466     487       0       0     1.0000     2.0000     0.0000     0.0000
 ASET    4599    4633       0       0     2.0000     2.0000     0.0000     0.0000
 ASNT       1       2       1       1     0.0000     0.0000     0.0000     0.0000
 READ       0       0       0       0     0.0000     0.0000     0.0000     0.0000
 ELST      -1       0       0       0     0.0000     0.0000     0.0000     0.0000
```

`DEBG`: Different debugging flags can be set. In this example, arg1=1 tells MacroModel to write out the set relationships to the log file.

`ASET`: Defines atom sets. Arg1-arg4 lists atom-numbers of atoms that participate in a set. The set number is defined in arg5. Arg6 allows for numerous alternatives of defining sets using arg1-arg4. For instance, a range of atoms can be added to a set by using arg1 and arg2 (first and last atom number of atoms to be included in the set) only, with arg6=2, as in this example. Arg7 and arg8 control ASET properties written to the output structure file. For details on the `ASET` opcode, see Section 4.5 of the *MacroModel Reference Manual*.

`ASNT`: Turn on/off set interactions. Arg1 and arg2 give the set numbers (as defined in `ASET`). Arg3=1 turns force field interactions on and arg4=1 turns constraint interactions on.

`READ`: Read in the structure.

`ELST`: Calculate the interaction energies between the specified sets and record only limited information on the interactions in the system to the `.log` file.

17.3 Rewinding the Output File for Additional Minimization

The `RWND` opcode, when included in a MacroModel command file, uses the input or output file again for a second computation. Currently this is supported only for a conformation search followed by a multiple minimization of the resulting conformers. This enables a conformation search to be performed with a higher minimization gradient specified for generated conformers, followed by minimization to a lower gradient with duplicate elimination, all in a single MacroModel computation. This workflow can be more efficient than minimizing all generated structures during the conformation search itself. The same potential settings should be used for both segments of the calculation.

The command file below combines a low-mode conformation search with a subsequent multiple minimization of the generated conformers to efficiently produce a set of low-energy conformers with a small convergence gradient.

```
rwnd.mae
rwnd-out.mae
 MMOD       0      1      0      0     0.0000     0.0000     0.0000     0.0000
 FFLD      11      1      0      0     1.0000     0.0000     0.0000     0.0000
 SOLV       3      1      0      0     0.0000     0.0000     0.0000     0.0000
 EXNB       0      0      0      0     0.0000     0.0000     0.0000     0.0000
 BDCO       0      0      0      0    89.4427 99999.0000     0.0000     0.0000
 READ       0      0      0      0     0.0000     0.0000     0.0000     0.0000
 LMCS    1000      0      0      0     0.0000     0.0000     3.0000     6.0000
 MCSS       2      0      0      0    50.0000     0.0000     0.0000     0.0000
 MCOP       1      0      0      0     0.0000     0.0000     0.0000     0.0000
 DEMX       0   1666      0      0    50.0000   100.0000     0.0000     0.0000
 COMP       0      0      0      0     0.0000     0.0000     0.0000     0.0000
 MSYM       0      0      0      0     0.0000     0.0000     0.0000     0.0000
 CONV       2      0      0      0     0.0500     0.0000     0.0000     0.0000
 MINI       1      0   5000      0     0.0000     0.0000     0.0000     0.0000
 RWND       0      0      0      0     0.0000     0.0000     0.0000     0.0000
 BGIN       0      0      0      0     0.0000     0.0000     0.0000     0.0000
 READ       0      0      0      0     0.0000     0.0000     0.0000     0.0000
 CONV       2      0      0      0     0.0200     0.0000     0.0000     0.0000
 MINI       1      0   1300      0     0.0000     0.0000     0.0000     0.0000
 END        0      0      0      0     0.0000     0.0000     0.0000     0.0000
```

All opcodes above the `RWND` command are necessary for a standard low-mode search. The `COMP` command indicates that all heavy atoms are to be used for the redundant conformer comparison. The minimizations, after conformation generation, are continued only until a gradient of 0.05 is reached, as seen in arg5 of the `CONV` command.

Arg1 of the `RWND` command indicates that the output file of the low-mode search is to be used as the input of the subsequent multiple minimization. The second argument indicates that the intermediate conformation search output structure file is to be discarded, and only the final results of the minimization are retained.

Below the `RWND` command are the opcodes necessary for the subsequent multiple minimization. Another `CONV` is included which indicates, with the fifth argument, a tighter convergence than was used in the conformation search.

17.4 Visualizing Vibrational Modes

Vibrational modes from MacroModel calculations can be visualized in Maestro using the ePlayer. Two opcodes, `VIBR` and `VBR2`, generate structures for a set of selected vibrational modes. `VIBR` is used for small molecules, whereas `VBR2` is used for larger molecules of the order of seven residues or larger. `VBR2` utilizes the same techniques as a large-scale low-mode conformation search, which include using ARPACK routines to solve the large eigenvalue problem. The arguments to `VIBR` and `VBR2` specify the first and last mode to visualize, as well as the number of frames to include for the period of each vibration. Rotational and translational modes are automatically disregarded. The structural output consists of a series of frames (structures) that together animate the vibration when imported and played in Maestro's ePlayer.

`VIBR` and `VBR2` computations are started from the command line with an appropriate command file and an input structure file. The input structure should be minimized to a low gradient. Below is a sample command file for a `VIBR` calculation.

```
vibr.mae
vibr-out.mae
 FFLD     10      1      0      1     1.0000     0.0000     0.0000     0.0000
 SOLV      3      1      0      0     0.0000     0.0000     0.0000     0.0000
 EXNB      0      0      0      0     0.0000     0.0000     0.0000     0.0000
 BDCO      0      0      0      0    89.4427 99999.0000     0.0000     0.0000
 READ      0      0      0      0     0.0000     0.0000     0.0000     0.0000
 CONV      2      0      0      0     0.0200     0.0000     0.0000     0.0000
 MINI      1      0    500      0     0.0000     0.0000     0.0000     0.0000
 VIBR      1      5      5      0     3.0000     0.0000     0.0000     0.0000
```

The command file specifies `vibr.mae` as the input structure file, and the output structural animations will reside in the file `vibr-out.mae`. The following four opcodes specify the potential settings, here MMFFs as the force field with solvation, extended non-bonded cutoffs, and `BDCO` usage for electrostatic interactions. After the structure is read, a minimization is requested (`CONV`, `MINI`). The `VIBR` opcode selects the first five non-trivial modes for animation, with five frames per quarter vibration cycle.

17.5 Alignment of Structures

The combination of the `COPY` and `ALGN` opcodes permits very rapid approximate alignment of ligands with a reference ligand using center of mass and moments of inertia. These commands may be used to preposition ligands crudely prior to an Embrace conformational search calculation based upon the position of a reference ligand already positioned in the active site. While this combination of commands may be useful the positioning is crude, and searching conformational space is slow and quite limited compared to that available in Schrödinger's docking program, Glide.

Note that `COPY`/`ALGN` may require the purchase of a separate license.

Below is an example command file for a `COPY`/`ALGN` run to align structures in the input structure file (`algn.mae`). Opcode descriptions follow.

```
algn.mae
algn_all-out.mae
 READ       0      0      0      0     0.0000     0.0000     0.0000     0.0000
 COPY       0      0      0      0     0.0000     0.0000     0.0000     0.0000
 BGIN       0      0      0      0     0.0000     0.0000     0.0000     0.0000
 READ       0      0      0      0     0.0000     0.0000     0.0000     0.0000
 ALGN       3      1      5      0     0.0000     0.0000     0.0000     0.0000
 END        0      0      0      0     0.0000     0.0000     0.0000     0.0000
```

`READ`: The first `READ` command reads in the first structure, which is the reference structure. The remaining structures are read in a `BGIN`/`END` loop with another `READ` command.

`COPY`: Saves a copy of the current structure in internal reference arrays for later use.

`BGIN`/`END`: Loop over the remaining structures in the file.

`ALGN`: Aligns each structure with the reference structure by the center of mass and moments of inertia (arg1=3, arg2=1). All four equivalent alignment are generated (arg3=5).

17.6 autoref: Restrained Minimizations

A script is available for performing restrained minimizations. The primary application is for the minimization of protein-ligand structures just after hydrogen atoms have been added. In this procedure, the heavy atoms in the system are restrained using a harmonic potential while the hydrogen atoms do not have restraining potentials applied.

Before you use `autoref`, you must convert the structure file from Maestro format (`.mae`) to MacroModel format (`.dat`) using the structure conversion utility `maemmod`, as shown below.

```
$SCHRODINGER/utilities/maemmod basename.mae basename.dat
```

The syntax of the `autoref` command is as follows:

```
$SCHRODINGER/utilities/autoref [options] basename.dat
```

The output file is named *basename*`_ref.dat`. The command options are given in Table 17.1.

Table 17.1. Options for the autoref utility.

Option	Description
`-k`	Keep output structure files from intermediate stages.
`-l` *rmsd*	Specify maximum RMS deviation from the initial structure; default is 0.3.
`-m`	Use the MMFFs force field instead of the default OPLS_2005.
-o	Use the OPLS_2001 force field instead of the default OPLS_2005.
`-v`	Print version number and exit.

17.7 serial_split: Split Serial Search Output Structure Files

A serial search produces an output structure file that contains the collections of conformations produced from a number of different molecules. The `serial_split` utility can be used to produce separate files, each containing the conformers produced for a different molecule. The `serial_split` utility can be used on the output files of Embrace conformational searches if there are only ligands or only complexes in the file. The syntax of the command is:

`SCHRODINGER/utilities/serial_split [`*options*`]` *input*`.mae [`*basename*`]`

where *input*`.mae` contains the results of the previously performed serial conformational search. The options are given in Table 17.2.

Table 17.2. Options for the `serial_split` *utility.*

Option	**Description**
`-h`	Print usage summary and exit.
`-b` *number*	The serial number of the first molecule for which conformations should be extracted. If omitted, start with the first serial number present.
`-e` *number*	The serial number of the last molecule for which conformations should be extracted. If omitted, continue to the last serial number present.
`-p` *property*	Use values of the specified property for *basename*, which can then be omitted. The property must be a string property `s_`*family_propertyname*.
`-v`	Print the version number.

The files produced are named *basename*<serial number>`.mae` where <serial number> is the `i_mmod_Serial_Number` property listed for each structure in the input structure file. This number is usually a running count of the distinct structures present. The base for the serial number can be controlled with the `SRNO` opcode in the MacroModel calculation.

17.8 ring_templating: Creating Ring Conformation Templates

Several of Schrödinger's products rely on the templating of ring conformations for selection of the physically reasonable, low-energy conformations. A library of ring templates is included in the software distribution, but of course there are always rings that are not included in the library. For this reason, the `ring_templating` utility has been added, so that you can create templates for the rings that lack templates in the supplied library.

This utility was designed to generate new ring templates by doing a large-scale conformational search with MacroModel (and so requires a MacroModel license). This process can take several hours for each ring system, so it is not practical for automatic generation of templates during routine runs of applications such as LigPrep, ConfGen, Glide, or Phase. However, it does allow you to add templates that are automatically included in these applications.

17.8.1 Generating Ring Conformation Templates

The command syntax is as follows:

`ring_templating [`*options*`]` *input-file*

(This command must be preceded by `$SCHRODINGER/utilities/` on Linux and Mac.) The options are given in Table 17.3.

Table 17.3. Options for the ring_templating command.

Option	**Description**
`-a|-attachments`	Retain at least one atom in each attachment to the ring. This can help to produce better conformations.
`-f|-force_generate_tmpl`	Forcibly generate templates even if pre-existing templates are found.
`-g|-skip_general_tmpl`	Ignore general templates in `ring_conf` and generate exact templates.
`-h|--help`	Show usage message and exit.
`-K|-keep` *numconf*	Number of conformers to retain.
`-M|-macrocycle`	Use macrocycle sampling script for ring systems with more than 8 atoms. See Section 16.2 on page 165 for more information.
`-o|-ffld` *int*	Numeric index of force field for calculations. Default: 14 (OPLS_2005).
`-P|-prioritize` *occur*	Generate templates only if the ring system is found more than *occur* times in the input file.
`-s|-short_search`	Run a shorter calculation (3000 steps) to generate templates. Use this option with smaller ring systems (simple rings with < 7 atoms). The default search has 20000 steps.
`-T|-SearchType {LMCS|MCMM}`	Set the conformational search method (LMCS or MCMM). See Section 8.1 on page 75 for information on these methods.
`-version`	Show program version number and exit.
`-z[ip_file]` *zipfile*	Name of the archive file in which the templates will be stored. The name must end in `.zip`. If you provide the name of an existing archive, it is overwritten. The default is `ring_templates.zip`.

The input file is the file that contains the ring systems to be templated. It is advisable to include structures in this file for all desired stereoisomers of the ring system, as the ring templating process does not generate different stereoisomers. This utility is primarily designed to use the file that the `ring_conf` stage of a LigPrep creates for ligands that lack ring templates, which is named *jobname*`_rcout_tmpl_att_prob.mae`. Because the `ring_conf` stage follows the `stereoizer` stage in most LigPrep runs, the appropriate chiral centers are already labeled in this file. Ring templates are generated for all structures in the input file.

The output consists of a zip archive (`.zip`) that contains an index file, `index.json`, and structure files in Maestro format that contain the templates. This is a change in format from the 2014-2 release. The previous format is no longer supported, so you must convert any existing templates to the new format, which you can do with the `zip_templates` utility:

`zip_templates` *template-dir zipfile*

where *template-dir* is the directory that contains the `mmconf_rc_list` file along with the Maestro template files in the old format, and *zipfile* is the name of the archive file in the new format. (This command must be preceded by `$SCHRODINGER/utilities/` on Linux and Mac.)

17.8.2 Using Ring Conformation Templates

When you have created the new templates, you can add them to the following directory for automatic inclusion in any application that you run that needs ring templates:

Linux, Mac:	`$HOME/.schrodinger/mmshare/mmconf_rc_templates`
Windows:	`%APPDATA%\Schrodinger\mmshare\mmconf_rc_templates`

To make the templates available for use by all users, you can install them in the `mmshare-v`*version*`/data/mmconf_rc_database` directory in the installation, if you have write permission. Alternatively, you can put them in a public location, and users who want to use them can set the environment variable `SCHRODINGER_RING_TEMPLATE_PATH` to point to this location. If there are multiple template locations, you can concatenate them in this environment variable as for the `PATH` variable, by separating each location with colons (`:`) on Linux and Mac, and semicolons (`;`) on Windows.

For information on setting environment variables, see Appendix A of the *Installation Guide*.

When a program that uses the conformer generation library (`mmconf`) requires ring templates, it searches the following locations, in order:

1. The directory set in the `SCHRODINGER_RING_TEMPLATE_PATH` environment variable.

2. The ring template directory in your local application data directory,

   `$HOME/.schrodinger/mmshare/mmconf_rc_templates` or

   `%APPDATA%\Schrodinger\mmshare\mmconf_rc_templates`

3. The installation directory `mmshare-v`*version*`/data/mmconf_rc_database`.

If you are running a MacroModel, LigPrep, or ConfGen job remotely, the ring templates are copied from the first two locations on the local host, and transferred to the job directory on the remote host. The copied templates are used in preference to the ones in the first two locations listed above. As the copied templates are placed in a single directory, you should ensure that the names of the zip files are unique. The templates in the installation on the execution host (whether remote or local) are always used as a fallback.

The copying of the templates ensures that you get the same results whether the job is run locally or remotely. This imposes some restrictions:

- Custom templates that are only available on the remote host will not be used, unless they are placed in the installation.
- Custom templates must be placed in the job launch directory (or the installation) if you run the job with `-LOCAL`, as Job Control does not copy any files in this case.

There is a way of bypassing the custom templates in the first two locations and specifying the zip files to use directly. These zip files are then copied to the remote host. To do this, you can set the environment variable `SCHRODINGER_RING_TEMPLATE_SOURCES` to a list of paths to each zip file, separated by `:` on Linux and Mac, and `;` on Windows. Setting this environment variable to a null string has the effect of suppressing the use of the custom templates. This envi environment variable is primarily intended for internal use on the remote host, to use the copied templates; however it can be used on the local host to override the normal custom template selection mechanism.

For other applications, such as Glide, the files must exist in the appropriate locations on the remote host. In this case, you could set the `SCHRODINGER_RING_TEMPLATE_PATH` environment variable in the entry for the remote host in the hosts file (`schrodinger.hosts`) to identify a custom templates directory on that host.

17.9 Python Scripts

The MacroModel distribution contains several Python scripts that can be used for running various jobs. These scripts are stored in `$SCHRODINGER/macromodel-v`*version*`/python`. To run a script, use the following syntax:

```
$SCHRODINGER/run -FROM macromodel script-name [options]
```

For more information on using Python for scripting, see the *Python Module Overview*, which you can open from the Help menu in Maestro or from the HTML documentation index. See Chapter 15 of the *Maestro User Manual* for information on using scripts with Maestro.

Getting Help

Information about Schrödinger software is available in two main places:

- The `docs` folder (directory) of your software installation, which contains HTML and PDF documentation. Index pages are available in this folder.
- The Schrödinger web site, http://www.schrodinger.com/, In particular, you can use the Knowledge Base, http://www.schrodinger.com/kb, to find current information on a range of topics, and the Known Issues page, http://www.schrodinger.com/knownissues, to find information on software issues.

Finding Information in Maestro

Maestro provides access to nearly all the information available on Schrödinger software.

To get information:

- Pause the pointer over a GUI feature (button, menu item, menu, ...). In the main window, information is displayed in the Auto-Help text box, which is located at the foot of the main window, or in a tooltip. In other panels, information is displayed in a tooltip.

  If the tooltip does not appear within a second, check that Show tooltips is selected under General → Appearance in the Preferences panel, which you can open with CTRL+, (⌘,). Not all features have tooltips.

- Click the Help button in the lower right corner of a panel or press F1, for information about a panel or the tab that is displayed in a panel. The help topic is displayed in the Help panel. The button may have text or an icon:

  ?

- Choose Help → Online Help or press CTRL+H (⌘H) to open the default help topic.
- When help is displayed in the Help panel, use the navigation links in the help topic or search the help.
- Choose Help → Manuals Index, to open a page that has links to all the PDF documents. Click a link to open the document.

- Choose Help → Search Manuals to search the manuals. The search tab in Adobe Reader opens, and you can search across all the PDF documents. You must have Adobe Reader installed to use this feature.

For information on:

- Problems and solutions: choose Help → Knowledge Base or Help → Known Issues → *product*.
- New software features: choose Help → New Features.
- Scripts available for download: choose Scripts → Update.
- Python scripting: choose Help → Python Module Overview.
- Utility programs: choose Help → About Utilities.
- Keyboard shortcuts: choose Help → Keyboard Shortcuts.
- Installation and licensing: see the *Installation Guide*.
- Running and managing jobs: see the *Job Control Guide*.
- Using Maestro: see the *Maestro User Manual*.
- Maestro commands: see the *Maestro Command Reference Manual*.

Contacting Technical Support

If you have questions that are not answered from any of the above sources, contact Schrödinger using the information below.

Web: http://www.schrodinger.com/supportcenter
E-mail: help@schrodinger.com
Mail: Schrödinger, 101 SW Main Street, Suite 1300, Portland, OR 97204
Phone: +1 503 299-1150 (USA, 9am – 5pm Pacific Time)
+49 621 438-55173 (Europe, 9am – 5pm Central European Time)
Fax: +1 503 299-4532 (USA, Portland office)
FTP: ftp://ftp.schrodinger.com

Generally, using the web form is best because you can add machine output and upload files, if necessary. You will need to include the following information:

- All relevant user input and machine output
- MacroModel purchaser (company, research institution, or individual)
- Primary MacroModel user
- Installation, licensing, and machine information as described below.

Gathering Information for Technical Support

The instructions below describe how to gather the required machine, licensing, and installation information, and any other job-related or failure-related information, to send to technical support. Where the instructions depend on the profile used for Maestro, the profile is indicated.

For general enquiries or problems:

1. Open the Diagnostics panel.
    - **Maestro:** Help → Diagnostics
    - **Windows:** Start → All Programs → Schrodinger-2014-3 → Diagnostics
    - **Mac:** Applications → Schrodinger2014-3 → Diagnostics
    - **Command line:** `$SCHRODINGER/diagnostics`
2. When the diagnostics have run, click Technical Support.

    A dialog box opens, with instructions. You can highlight and copy the name of the file.
3. Attach the file specified in the dialog box to the web form.

    If you have already submitted a support request, use the upload link in the email response from Schrödinger to upload the file. If you need to submit a new request, you can attach the file when you fill in the form.

If your job failed:

1. Open the Monitor panel, using the instructions for your profile as given below:
    - **Maestro/Jaguar/Elements:** Tasks → Monitor Jobs
    - **BioLuminate/MaterialsScience:** Tasks → Job Monitor
2. Select the failed job in the table, and click Postmortem.

    The Postmortem panel opens.
3. If your data is not sensitive and you can send it, select Include structures and deselect Automatically obfuscate path names.
4. Click Create.

    An archive file is created in your working directory, and an information dialog box with the name of the file opens. You can highlight and copy the name of the file.
5. Attach the file specified in the dialog box to the web form.

    If you have already submitted a support request, use the upload link in the email response from Schrödinger to upload the file. If you need to submit a new request, you can attach the file when you fill in the form.

6. Copy and paste any log messages from the window used to start the interface or the job into the web form (or an e-mail message), or attach them as a file.
   - **Windows:** Right-click in the window and choose Select All, then press ENTER to copy the text.
   - **Mac:** Start the Console application (Applications → Utilities), filter on the application that you used to start the job (Maestro, BioLuminate, Elements), copy the text.

If Maestro failed:

1. Open the Diagnostics panel.
   - **Windows:** Start → All Programs → Schrodinger-2014-3 → Diagnostics
   - **Mac:** Applications → SchrodingerSuite2014-3 → Diagnostics
   - **Linux/command line:** `$SCHRODINGER/diagnostics`
2. When the diagnostics have run, click Technical Support.

   A dialog box opens, with instructions. You can highlight and copy the name of the file.
3. Attach the file specified in the dialog box to the web form.

   If you have already submitted a support request, use the upload link in the e-mail response from Schrödinger to upload the file. If you need to submit a new request, you can attach the file when you fill in the form.
4. Attach the error files to the web form.

   The files should be in the following location:

- **Windows:** `%LOCALAPPDATA%\Schrodinger\appcrash`
  (Choose Start → Run and paste this location into the Open text box.)
  Attach `maestro_error`*pid*`.txt` and `maestro.exe`*pid_timestamp*`.dmp`.
- **Mac:** `$HOME/Library/Logs/CrashReporter`
  (Go → Home → Library → Logs → CrashReporter)
  Attach `maestro_error`*pid*`.txt` and `maestro_`*timestamp_machinename*`.crash`.
- **Linux:** `$HOME/.schrodinger/appcrash`
  Attach `maestro_error`*pid*`.txt` and `crash_report_`*timestamp_pid*`.txt`.

References

1. Allinger, N. L. Conformational Analysis 130. MM2. A Hydrocarbon Force Field Utilizing V1 and V2 Torsional Terms. *J. Am. Chem. Soc.* **1977,** *99,* 8127.

2. Allinger, N. L.; Yuh, Y. H.; Lii, J-H. Molecular Mechanics. The MM3 Force Field for Hydrocarbons. *J. Am. Chem. Soc.* **1989,** *111,* 8551.

3. Ferguson, D. M.; Kollman, P. A. Can the Lennard-Jones 6-12 function replace the 10-12 form in molecular mechanics calculations? *J. Comput. Chem.* **1991,** *12,* 620.

4. McDonald, D. Q.; Still, W. C. AMBER torsional parameters for the peptide backbone. *Tetrahedron Lett.* **1992,** *33,* 7743.

5. Jorgensen, W. L.; Tirado-Rives, J. The OPLS Potential Functions for Proteins. Energy Minimization for Crystals of Cyclic Peptides and Crambin. *J. Am. Chem. Soc.* **1988,** *110,* 1657.

6. Jorgensen, W. L.; Maxwell, D. S.; Tirado-Rives, J. Development and Testing of the OPLS All-Atom Force Field on Conformational Energetics and Properties of Organic Liquids. *J. Am. Chem. Soc.* **1996,** *118,* 11225-11235.

7. Kaminski, G. A.; Friesner, R. A.; Tirado-Rives, J.; Jorgensen, W. L. Evaluation and Reparametrization of the OPLS-AA Force Field for Proteins via Comparison with Accurate Quantum Chemical Calculations on Peptides. *J. Phys. Chem. B* **2001,** *105,* 6474-6487.

8. Cornell, W. D.; Cieplak, P.; Bayly, C. I.; Gould, I. R.; Merz, K. M.; Ferguson, D. M.; Spellmeyer, D. C.; Fox, T.; Caldwell, J. W.; Kollman, P. A. A second generation force field for the simulation of proteins, nucleic acids and organic molecules. *J. Am. Chem. Soc.* **1995,** *117,* 5179.

9. Halgren, T. A. Merck Molecular Force Field. I. Basis, Form, Scope, Parameterization and Performance of MMFF94. *J. Comput. Chem.* **1996,** *17,* 490-519.

10. Halgren, T. A. Merck Molecular Force Field. II. MMFF94 van der Waals and Electrostatic Parameters for Intermolecular Interactions. *J. Comput. Chem.* **1996,** *17,* 520-552.

11. Halgren, T. A. Merck Molecular Force Field. III. Molecular Geometrics and Vibrational Frequencies for MMFF94. *J. Comput. Chem.* **1996,** *17,* 553-586.

12. Halgren, T. A.; Nachbar, R. B. Merck Molecular Force Field. IV. Conformational Energies and Geometries. *J. Comput. Chem.* **1996,** *17,* 587-615.

13. Halgren, T. A. Merck Molecular Force Field. V. Extension of MMFF94 using Experimental Data, Additional Computational Data and Empirical Rules. *J. Comput. Chem.* **1996,** *17,* 616-641.

14. Halgren, T. A. MMFF VI. MMFF94s Option for Energy Minimization Studies. *J. Comput. Chem.* **1999,** *20,* 720-729.

15. Halgren, T. A. MMFF VII. Characterization of MMFF94, MMFF94s and Other Widely Available Force Fields for Conformational Energies and for Intermolecular Interaction Energies and Geometries. *J. Comput. Chem.* **1999,** *20,* 730-748.

16. Still, W. C.; Tempczyk, A.; Hawlely, R. C.; Hendrickson, T. A General Treatment of Solvation for Molecular Mechanics. *J. Am. Chem. Soc.* **1990,** *112,* 6127.

17. Polak, E.; Ribiere, G. Note sur la Convergence de Méthodes de Directions Conjuguées. *Revenue Francaise Informat. Recherche Operationelle, Serie Rouge,* **1969,** *16,* 35.

18. Ponder, J. W.; Richards, F. M. An Efficient Newton-like Method for Molecular Mechanics Energy Minimization of Large Molecules. *J. Comput. Chem.* **1987,** *8,* 1016.

19. Oren, S. S.; Spedicato, E. Optimal conditioning of self-scaling variable metric algorithms. *Mathematical Programming* **1976,** *10,* 70.

20. Chang, G.; Guida, W.; Still, W. C. An internal coordinate Monte-Carlo method for searching conformational space. *J. Am. Chem. Soc.* **1989,** *111,* 4379.

21. Saunders, M.; Houk, K. N.; Wu, Y.-D.; Still, W. C.; Lipton, M.; Chang, G.; Guida, W. C. Conformations of Cycloheptadecane: A Comparison of Methods for Conformational Searching. *J. Am. Chem. Soc.* **1990,** *112,* 1419.

22. Goodman, J. M.; Still, W. C. Searching Conformation Space. *J. Comput. Chem.* **1991,** *12,* 1110.

23. Kolossváry, I.; Guida, W. C. Low Mode Search. An efficient, automated computational method for conformational analysis: application to cyclic and acyclic alkanes and cyclic peptides. *J. Am. Chem. Soc.* **1996,** *118,* 5011.

24. Kolossváry, I.; Keseru, G. M. Hessian-Free Low-Mode Conformational Search for Large-Scale Protein Loop Optimization: Application to c-jun N-Terminal Kinase JNK3. *J. Comput. Chem.* **2001,** *22*, 21.

25. Keseru, G. M.; Kolossváry, I. Fully Flexible Low-Mode Docking: Application to Induced Fit in HIV Integrase. *J. Am. Chem. Soc.* **2001,** *123*, 12708.

26. Kolossváry, I.; Guida, W. C. Low-mode Conformational Search Elucidated. Application to $C_{39}H_{80}$ and Flexible Docking of 9-Deazaguanine Inhibitors to PNP. *J. Comput. Chem.* **1999,** *20,* 1671.

27. van Gunsteren, W. F.; Berendsen, H. J. C. A leap-frog algorithm for stochastic dynamics. *Mol. Simul.* **1988,** *1*, 173.

28. Ryckaert, J.-P.; Ciccotti, G.; Berendsen, H. J. C. Numerical integration of the Cartesian equation of motion of a system with constraints: molecular dynamics of N-alkanes. *J. Comput. Phys.* **1977,** *23,* 327.

29. Guarnieri, F.; Still, W. C. A Rapidly Convergent Simulation Method: Mixed Monte Carlo/Stochastic Dynamics. *J. Comput. Chem.* **1994,** *15,* 1302-1310.

30. Dado, G. P.; Gellman, S. H. On the Use of AM1 Calculations for the Study of Intramolecular Hydrogen Bonding Phenomena in Simple Amides. *J. Am. Chem. Soc.* **1992,** *114,* 3138.

31. McDonald, D. Q.; Still, W. C. An AMBER* study of Gellman's amides. *Tetrahedron Lett.* **1992,** *33,* 7747.

32. Gellman, S. H.; Dado, G. P.; Liang, G-B.; Adams, B. R. Conformation-Directing Effects of a Single Intramolecular Amide-Amide Hydrogen Bond: Variable Temperature NMR and IR Studies on a Homologous Diamide Series. *J. Am. Chem. Soc*. **1991,** *113,* 1164.

33. McDonald, D. Q.; Still, W. C. Conformational Free Energies from Simulation: Stochastic Dynamics/Monte Carlo Simulations of a Homologous Series of Gellman's Diamides. *J. Am. Chem. Soc.* **1994,** *116,* 11550.

34. Hornak, V.; Abel, R.; Okur, A.; Strockbine, B.; Roitberg, A.; Simmerling, C. *Proteins: Struct., Funct., Bioinf.* **2006**, *65*, 712.

Index

A

alignment of structures.................................. 170
AMBER* 10
    compatible H-treatment............................. 21
AMBER94 10
angles
    coordinate scan 71
    monitored in dynamics calculations......... 100
arguments, opcode 25
ASL, use in substructure files 37
atom positions
    constraining 37
    fixing and freezing...................................... 33
atom sets, interaction energies between.......... 167
automatic setup
    in CSearch calculations 80
    in MC/SD calculations 107

B

bond dipole cutoffs (BDCO)...................... 14, 16

C

charges, using from structure files 32
Chiral Atoms panel 88
chiral atoms, selecting for CSearch 88
command file 24
    format 24
command file examples
    alignment of structures 170
    ASET 167
    coordinate scans... 71
    CSearch 91
    current energy 46
    dynamics.. 104
    Embrace... 128
    Embrace conformational search 134
    Embrace, distributed................................ 141
    free-energy perturbation 147
    geometry calculations............................... 167
    LOOP... 117
    MC/SD calculations 109
    minimization.. 60
    MINTA 113
    partition coefficient estimation.................. 65
    redundant conformer elimination 162
    rewinding output file for minimization 168
    vibrational animation................................ 170
comparison atoms
    in CSearch calculations 87
Comparison Atoms panel 88
comparison atoms,
  in minimization of conformers 55
Comparison tab, Minimization panel 55
conformational search
    macrocycles 163
    protein-ligand complexes 123
    rings 172
    small molecules 75
    See also CSearch calculations
Conformational Search panel 77
conformations, mirror-image 57, 81, 161
constraints
    on bond lengths using SHAKE 102
    for fixing atoms 34
    on positions, distances, angles, torsions..... 37
    reading and writing.................................... 39
    removing.. 39
Constraints tab 37, 38
conventions, document xi
convergence guidelines.................................... 17
    for CSearch jobs 17
    for MCSD jobs 18
    for minimization jobs 17
convergence threshold...................................... 54
Coordinate Scan panel 70
coordinate scans... 69
    command file examples.............................. 71
    defining range and increment..................... 71
CSearch calculations.. 75
    automatic setup of 80
    chiral atoms 88
    command file examples.............................. 91
    comparison atoms....................................... 87
    defining ring closures 83
    defining torsion rotations........................... 84
    distance check 89
    energy window 81
    ligand bonds 90
    markers 83
    number of steps 81
    number of steps per rotatable bond 81
    resetting variables................................ 80, 83
    torsion check 90

- translations and rotations 85
- variables 79

CSearch methods
- LMCS 75
- MCMM 75
- MCMM/LMCS 76
- selecting 77
- SPMC 75

CSearch tab 77, 124
current energy calculations 45
Current Energy panel 45

D

dielectric constant, modifying 32
directory
- installation 4
- Maestro working 4

Distance Check panel 89
distances
- checked in CSearch 89
- constraining 37
- coordinate scan 69
- monitored in dynamics calculations 99

distributed calculations 28, 43
dynamics calculations 97
- command file examples 104
- equilibration time 103
- method 102
- monitored angles 100
- monitored dihedrals 100
- monitored distances 99
- monitored H-bonds 101
- monitored surface areas 99
- SHAKE 102
- simulation time 103
- temperature 102
- timestep 103

Dynamics panel 97, 98, 103
Dynamics tab 102, 103

E

ECalc tab 45, 46
electrostatic interaction truncation 14
electrostatic treatments 20
Embrace calculations 121
- association energy mode 123
- command file examples 128
- receptor entry 122
- sample output 130

Embrace Conformational Search panel 124
Embrace Minimization panel 122
Embrace tab 122
entry for receptor in Embrace 122
environment variable
- `SCHRODINGER` 4, 23
- `SCHRODINGER_TMPDIR` 23

equilibration time 103

F

FEP calculations 145
- hints for success 152
- literature references for 156
- preparing 146
- sample output 150
- structure file 146

file conversion 20
files
- atom type 21
- force field 21
- solvent 21

FMNR minimization method 53
force fields 9
- files 9
- selecting 32

free energy perturbation calculations—*see* FEP calculations

G

GB/SA 13
geometry calculations 167

H

H-bonds monitored in dynamics calculations. 101
hydrogen treatment 21

I

interaction energies using `ASET` 167
iteration limit, minimization calculations 54

J

job submission
    command line 26
    remote and distributed 28
jobs
    general setup 31
    input source 31
    monitoring 29, 43
jobs, restarting 27

L

LBFGS minimization method 53
ligand bonds checked in CSearch calculations . 90
Ligand Bonds panel 90
LMCS 75
LOOP calculations 115
    command file examples 117
    example output 119
    input restrictions 115
    structure handling 116
low-mode conformational search methods 75
low-mode settings 82
`.lsq` file example 116

M

Macrocycle Conformational Sampling panel . 164
macrocycles, conformational sampling of 163
`MBAE` opcode 129, 132
MC/SD calculations 107
    automatic setup 107
    command file examples 109
    MC to SD ratio 108
MC/SD panel 107, 108
MC/SD tab 108
MCMM 75
MCMM/LMCS 76
Mini tab 53
minimization calculations 51
    command file examples 60
    convergence parameters 54
minimization methods 52
    FMNR 53
    LBFGS 53
    OSVM 52
    PRCG 52
    SD 52
    TNCG 52
minimization of conformers
    comparison atoms for 55
    energy window 57
Minimization panel 51, 53
MINTA calculations 111
    command file examples 113
    energy evaluations per iteration 112
    hard limit 112
    input file 111
    number of iterations 111
    soft limit 113
    temperature 112
MINTA panel 111, 112
MINTA tab 112
mixed MCMM/Low-mode search method 76
MM2* 9
MM3* 10
MMFF 11
Molecule Trans/Rot panel 86
Monitor tab 97, 98
Monitored Angles panel 100
Monitored Atoms Surface Area panel 99
Monitored Distances panel 100
Monitored H-Bonds panel 101
Monitored Torsions panel 100
Monte Carlo Multiple Minimum search method 75
multi-ligand structure files, preminimization of 59

N

nonbonded cutoffs 19
    setting 32

O

opcodes 25
OPLS_2001 10
OPLS_2005 10
OPLS* 10
OPLSAA 10
OSVM minimization method 52

P

parameter quality considerations 11
partition coefficient estimation 57
    example 65

Potential tab 32, 33
PRCG minimization method............................ 52
`premin` script.. 59
product installation 178

R

receptor
    specifying entry for Embrace 122
    substructure in Embrace 127
Redundant Conformer Elimination panel....... 160
remote calculations 28
results, checking and interpreting
    CSearch 96
    current energy 50
    dihedral drives 72
    dynamics.. 106
    minimization.. 66
    MINTA 114
results, dependence on task distribution 29
Ring Closures panel 83
ring templates, generating 172

S

Scan tab.. 70
Schrödinger contact information 178
SD minimization method 52
shells, defining 41
simulated annealing 105
    in macrocycle conformational sampling .. 163
simulation temperature 102
simulation time 103
size limitations 19
solvation treatment
    description 13
    selecting... 32
SPMC... 75

structures, alignment of 170
substructure
    defining... 35, 40
    filling out residues in 40
    for freezing or fixing atoms 34
    shell radius.. 40, 41
    shells 41
    specifying for Embrace 126
substructure files
    ASL use in 37
    format 36
    reading and writing.................................... 42
Substructure tab 40, 41
Systematic Pseudo-Monte Carlo search
  method 75

T

temperature
    dynamics calculations 102
    MC/SD calculations 107
    MINTA calculations 112
TNCG minimization method 52
Torsion Check panel 89
Torsion Rotations panel 85
torsions
    checked in CSearch 90
    constraining 37
    in MC/SD calculations 107
    monitored in dynamics calculations......... 100
    sampling amide, ester, azo, and imido 80, 164
translations and rotations in MC/SD
  calculations 107

V

variable torsions in MC/SD calculations 108
vibrational modes, visualizing 169

120 West 45th Street
17th Floor
New York, NY 10036

101 SW Main Street
Suite 1300
Portland, OR 97204

245 First Street
Riverview II, 18th Floor
Cambridge, MA 02142

8910 University Center Lane
Suite 270
San Diego, CA 92122

155 Gibbs St
Suite 430
Rockville, MD 20850-0353

Dynamostraβe 13
D-68165 Mannheim
Germany

Zeppelinstraβe 73
D-81669 München
Germany

Potsdamer Platz 11
D-10785 Berlin
Germany

Quatro House
Frimley Road
Camberley GU16 7ER
United Kingdom

8F Pacific Century Place
1-11-1 Marunouchi
Chiyoda-ku, Tokyo 100-6208
Japan

No. 102, 4th Block
3rd Main Road, 3rd Stage
Sharada Colony
Basaveshwaranagar
Bangalore 560079, India

SCHRÖDINGER®