

# User's Manual



Calculation of hydration structures of molecular surfaces and interfaces

Version 1.1

[www.mobywat.com](http://www.mobywat.com)

---

## **DISTRIBUTION OF PROGRAM MOBYWAT**

THIS PROGRAM IS FREE SOFTWARE: YOU CAN REDISTRIBUTE IT AND/OR MODIFY IT UNDER THE TERMS OF THE GNU GENERAL PUBLIC LICENSE AS PUBLISHED BY THE FREE SOFTWARE FOUNDATION, EITHER VERSION 3 OF THE LICENSE, OR (AT YOUR OPTION) ANY LATER VERSION. THIS PROGRAM IS DISTRIBUTED IN THE HOPE THAT IT WILL BE USEFUL, BUT WITHOUT ANY WARRANTY; WITHOUT EVEN THE IMPLIED WARRANTY OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. SEE THE GNU GENERAL PUBLIC LICENSE FOR MORE DETAILS. YOU SHOULD HAVE RECEIVED A COPY OF THE GNU GENERAL PUBLIC LICENSE ALONG WITH THIS PROGRAM. IF NOT, SEE <[HTTP://WWW.GNU.ORG/LICENSES/](http://www.gnu.org/licenses/)>.

## **DISCLAIMER OF WARRANTY**

THERE IS NO WARRANTY FOR THE PROGRAM, TO THE EXTENT PERMITTED BY APPLICABLE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING THE COPYRIGHT HOLDERS AND/OR OTHER PARTIES PROVIDE THE PROGRAM "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

## **LIMITATION OF LIABILITY**

IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW OR AGREED TO IN WRITING WILL ANY COPYRIGHT HOLDER, OR ANY OTHER PARTY WHO MODIFIES AND/OR CONVEYS THE PROGRAM AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY GENERAL, SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE THE PROGRAM (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY YOU OR THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH ANY OTHER PROGRAMS), EVEN IF SUCH HOLDER OR OTHER PARTY HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES.

## **DISCLAIMER ON THE WEB DOCUMENTATION OF MOBYWAT**

IN NO WAY CAN ANY RIGHTS BE DERIVED FROM, OR CLAIMS MADE, WITH REGARD TO THE CONTENT OF THIS WEBSITE. ALTHOUGH THE GREATEST POSSIBLE CARE HAS BEEN TAKEN WITH THE COMPILATION OF THE CONTENT OF THIS WEBSITE, IT IS POSSIBLE THAT CERTAIN INFORMATION MAY (AFTER A WHILE) BE OUT-OF-DATE OR (NO LONGER) BE CORRECT. WE ARE NOT RESPONSIBLE FOR EVENTUAL DAMAGES ARISING FROM THE USE OF INFORMATION FROM THIS SITE. WE HEREBY REJECT ALL RESPONSIBILITY FOR DAMAGES AS A RESULT OF THE USE OF THIS INFORMATION OR INFORMATION TO WHICH LINKS REFER ON THIS SITE (THESE SITES). THE INFORMATION ON THIS SITE MAY BE CHANGED WITHOUT PRIOR WARNING. WE DO NOT GIVE GUARANTEES WITH REGARD TO THE NATURE AND THE CONTENT OF THIS SITE. ALL RESPONSIBILITY FOR POSSIBLE DAMAGES DUE TO ACCESS TO AND USE OF THE SITE IS EXPLICITLY REJECTED BY US. WE DO NOT ACCEPT ANY RESPONSIBILITY WITH REGARD TO THE CONTENT, ADVERTISEMENTS, PRODUCTS, OR OTHER ISSUES ON SUCH SITES OR SOURCES OR AVAILABILITY. WE ARE NOT RESPONSIBLE FOR ANY KIND OF DAMAGE OR LOSS CAUSED BY OR IN CONNECTION WITH THE USE OF, OR BY RELYING ON THE CONTENT, PRODUCTS OR SERVICES OFFERED ON SUCH SITES OR SOURCES.

## **NOTE ON FUTURE VERSIONS**

ANALYSIS MODE OF THE PROGRAM IS UNDER DEVELOPMENT AND ITS DETAILS WILL BE RELEASED AND/OR MODIFIED IN FUTURE PUBLICATIONS.

## **AUTHOR OF MOBYWAT**

MOBYWAT WAS PROGRAMMED BY CSABA HETÉNYI.

---

### **Last page image**

Hydration structure of an interface can be determined by dynamic soaking of a ligand-target complex with virtual water molecules used in excess. The resulted atomic resolution structure of the hydrated complex allows complete exploration of interfacial interactions corresponding to links of the network "reflected" in the water droplet at the bottom. Ligand and protein molecules are colored in blue and light purple, respectively.

### **Web**

<http://mobywat.com>  
<http://xray.bmc.uu.se/~csaba/>

### **E-mail**

[questions@mobywat.com](mailto:questions@mobywat.com)  
[csabahete@yahoo.com](mailto:csabahete@yahoo.com)

Copyright © 2015-2016 Csaba Hetényi, PhD

---

# Contents

<b>Scope and applications</b> .....	4
<b>1 Background</b> .....	5
1.1 Determination of hydration structure .....	5
1.2 Experimental methods .....	5
1.2.1 Crystallography.....	5
1.2.2 Nuclear magnetic resonance (NMR) .....	6
1.3 Theoretical methods .....	6
1.3.1 Static approaches .....	6
1.3.2 Dynamic approaches .....	7
<b>2 Glossary</b> .....	8
<b>3 Program modes</b> .....	10
3.1 Analysis mode.....	11
3.1.1 Overview .....	11
3.1.2 Inputs.....	11
3.1.3 Algorithm.....	11
3.1.4 Outcomes .....	13
3.1.5 Usage .....	13
3.2 Prediction mode .....	15
3.2.1 Overview .....	15
3.2.2 Inputs.....	15
3.2.3 Algorithm.....	15
3.2.4 Outcomes .....	20
3.2.5 Usage .....	20
3.3 Validation sub-mode .....	22
3.3.1 Overview .....	22
3.3.2 Inputs.....	22
3.3.3 Algorithm.....	22
3.3.4 Outcomes .....	22
3.3.5 Usage .....	26
3.4 NetDraw mode .....	28
3.4.1 Overview .....	28
3.4.2 Inputs.....	28
3.4.3 Algorithm.....	28
3.4.4 Outcomes .....	29
3.4.5 Usage .....	30
3.5 Editing mode .....	32
3.5.1 Overview .....	32
3.5.2 Inputs.....	32
3.5.3 Algorithm.....	32
3.5.4 Outcomes .....	32
3.5.5 Usage .....	32
<b>4 File types, ranges, parameters</b> .....	34
4.1 Input files.....	34
4.1.1. Binary trajectory file (*.xtc).....	34

4.1.2 Separate PDB files (*.pdb).....	34
4.1.3 NMR-type PDB files (*.pdb).....	34
4.1.4 Binary pool waters file (*.plw) .....	35
4.2 Output files.....	35
4.3 Input ranges.....	35
4.3.1 Frame range .....	35
4.3.2 Molecular ranges.....	35
4.4 Input parameters.....	35
<b>5 Program details.....</b>	<b>37</b>
5.1 Organization of code .....	37
5.2 Water pool data type .....	37
5.3 Binary trajectory files .....	38
<b>6 Installation and tests .....</b>	<b>39</b>
<b>7 Version history .....</b>	<b>39</b>
Version 1.0.....	39
Version 1.1.....	39
<b>8 How to cite? .....</b>	<b>39</b>
<b>9 Applications .....</b>	<b>40</b>
9.1 Surface hydration .....	40
9.1.1. Running MD calculations on a target protein .....	40
9.1.2 Preparation of the trajectory for MobyWat .....	41
9.2 Interface hydration.....	43
<b>10 References.....</b>	<b>44</b>

## Scope and applications

MobyWat is a program for calculation of hydration structures and networks of molecular surfaces and interfaces. The program uses a series of frames from molecular dynamics simulations. MobyWat has been thoroughly tested on protein surfaces and target-ligand interfaces, and can be recommended for experimental or theoretical investigations dealing with hydration problems. Possible applications may include but are not restricted to the following projects.

- Refinements and analyses of hydration structure assigned by crystallography. Prediction of hydration structure at problematic (overlapping, non-defined) regions of the density map.
- Prediction of hydration structure of solute molecules such as proteins or their complexes measured by nuclear magnetic resonance spectroscopy.
- Building hydration structure around homology modeled proteins and other modeled molecules or surfaces.
- Selection of structural water molecules for calculation of binding strength between molecular partners of complexes.
- Selection of surface-bound water molecules stabilizing protein structure.
- Selection of conserved water molecules.
- Estimation of local density and mobility of the hydration structure.
- Classification and analysis of hydration networks.

# 1 Background

## 1.1 Determination of hydration structure

Hydration is involved in most biological processes. It is remarked in a prominent text book on protein structure and function (Petsko and Ringe 2009) that "... waters in fixed positions should be considered as part of the tertiary structure, and any detailed structure description that does not include them is incomplete." Surface hydration is key determinant of solubility and aggregation of solute molecules (Israelachvili and Wennerström 1996). Protein-ligand interactions are also largely affected by interfacial water molecules (Baron et al. 2012), and therefore, knowledge of their location is of primary importance during structure-based drug design. Whereas resolving hydration structure is important, it is also a very difficult task and there is no ultimate method for determination of hydration structure at atomic level.

The difficulties come from mobility and complexity of interactions of water molecules located on a molecular surface. Residence of a water molecule on the surface is affected not primarily by the strength of its protein-water interaction. It is "rather a topography that prevents the water molecule from exchanging by a cooperative mechanism" (Halle 2004a). Importantly, such a cooperative mechanism of exchange also includes several water-water interactions often detected (Finney 1977) between surface or interface water molecules. Thus, it is very problematic to predict the residence of water molecules in the hydration layer of a protein using merely thermodynamic or kinetic approaches (Halle 2004a).

A brief outlook is provided in the forthcoming text on available experimental and theoretical methods for determination of hydration structure placing an emphasis on their limitations.

## 1.2 Experimental methods

### 1.2.1 Crystallography

Arrays of hydrated protein molecules arranged in a regular, repeating manner can form a crystal which acts as a diffraction grating and scatters radiation ending up in a diffraction pattern. The diffraction pattern can be converted into electron density maps and used for three-dimensional structural fits of the protein molecule and the surrounding hydration layer. X-ray and neutron crystallography are primary, indispensable, and direct methods for determination of atomic coordinates of hydrating water molecules (Savage 1986). Functionally important water molecules generally reside on the surface or at interface positions. Their determination is possible at resolutions of at least 2 Å (Carugo 1999, Finney 1977). There are more than eighty thousand crystallographic structures deposited in the Protein Data Bank (PDB, Berman et al. 2003) containing atomic level information on water structure of molecular surfaces and interfaces. Despite the large number of PDB entries, there are limitations of crystallographic determination of the hydration structure – few of which listed below.

1) Whereas the number of crystallographic structure refinement techniques is increasing (e.g. Afonine et al. 2013), assignation of electron density peaks to possible interface water positions is still not a routine job due to inherent mobility of water and high number of degrees of freedom (Badger 1997).

- 2) Quality of a solved structure depends on molecular size (Finney 1977). For small proteins assignation of electron density peaks to atomic positions is easier than it is for larger macromolecules.
- 3) Electron density peaks of water are generally smaller than those of the surrounding (protein) interface as measured by X-ray diffraction. Small electron density peaks are consequences of small X-ray scattering of the oxygen atom compared to the surrounding group of atoms in the interface, and very low scattering power of hydrogen atoms. Thus, X-ray crystallography of water structure is focusing on a difficult task of identification of a small effect from water hidden among large effects from the surrounding molecules (Finney 1977).
- 4) Protein hydration in the crystal is not the same as in solution (Halle 2004a). For small proteins, 30–40% of the solvent-accessible surface is usually buried at crystal contacts (Islam & Weaver 1990), where water molecules often mediate protein–protein interactions.
- 5) Assignation of electron densities to water molecules is often performed to improve the fit of data during structural refinement. Misleading identification of water sites at this stage was found to be a bad practice (Ladbury 1996).
- 6) Cryocrystallography used for protection of the protein molecules from damages caused by high energy synchrotron beams suffer from structural cryo-artefacts (Halle 2004b).  
etc.

### 1.2.2 Nuclear magnetic resonance (NMR)

While crystallographic methods measure long-time-averaged occupancy of a water positions and provide direct information on hydration structure of protein surfaces, NMR detects only water molecules with residence time of the same magnitude of tumbling time of the molecule in solution (Schoenborn et al. 1995). However, NMR and related techniques can provide useful information on instantaneous time behavior of structural water molecules, such as their residence time on protein surfaces (Halle 2004a).

## 1.3 Theoretical methods

### 1.3.1 Static approaches

There are various rapid methods for prediction of hydration sites on molecular surfaces or in interfaces. A common feature of these methods is that they are focused on the protein molecule or on protein–water interactions and completely neglect water–water interactions and co-operations. Notably, these interactions largely determine residence of water molecules in the hydration network (see Section 1.2 for explanation). Most rapid methods use a static picture not considering dynamic exchange (mobility) between surface and bulk water molecules and focusing on protein(ligand)–water interactions. Prominent examples of static algorithms are described below.

**Knowledge-based.** Using structural reference data sets distilled from crystal structures of the PDB early methods were published for detection of hydration sites (Pitt and Goodfellow 1991).

**Structural.** It was shown that using e.g. directionality of hydrogen bonds can be applied for systematic solvation of proteins (Vedani and Huhta 1991).

**Scoring.** A method based on docking of water molecules to the protein binding sites and subsequent use of a scoring scheme for the selection process was recently introduced (Ross et al. 2012).

**Thermodynamics.** A force field-based approach (Schymkowitz et al. 2005) used free energy calculations in combination with the knowledge-based method of Pitt and Goodfellow (1991). A statistical mechanics-based approach with Monte Carlo sampling of possible hydration site configurations (Michel et al. 2009) was also developed. The method starts with definition of the binding site and fills up a grid covering the site with water molecules. Dynamic exchange of water molecules between the binding site and the bulk is not performed explicitly: an idealized particle concept is used to calculate exchange thermodynamics between bulk and the site.

### 1.3.2 Dynamic approaches

Molecular dynamics (MD) has long been applied (Rossky and Karplus 1979, van Gunsteren et al. 1983, Pettitt and Karplus 1987) for investigation of hydration of peptides and proteins. All-atom MD with explicit water models is an invaluable source of mobility information of any hydrated biological systems. During the simulation time, movements (trajectory) and all interactions of water molecules can be followed at atomic level including not only protein-water, but also water-water contacts and exchanges of primary importance (Section 1.2). Two main branches of approaches applying MD for prediction of hydrate structure are discussed below.

**Density-based calculations.** Several studies (Virtanen et al. 2010, Makarov 1998, Lounnas 1994) have dealt with MD-based calculation of average solvent density and construction of proximal radial distribution function (pRDF) of hydration shells for different atom types occurring in proteins. The aim of these studies is to use the constructed, generalized pRDFs for the reconstruction of hydration shell density of any protein without MD simulation. In other words, this approach applies MD for calculation of solvent density and construction of pRDFs. Positions of individual water molecules can be obtained from fits to densities. Limitations of the radial distribution function-based approaches were discussed in details (Henchman and McCammon 2002).

**Occupancy-based calculations.** With advancement of computational infrastructure and theory speed of MD calculation have increased in the past decades (Dror et al. 2012). It has become a real alternative to perform atomic level MD with explicit water molecules for analysis (Schoenborn et al. 1995) and direct prediction of hydration structure of a protein or its complex. Whereas there are numerous analysis studies, there are much fewer studies on testing the usefulness of direct MD approaches for obtaining hydration sites (Huang et al. 2008, Henchman and McCammon 2002, Madhusudhan 2001). Direct MD approaches use individual positions of hydrating water molecules (instead of average densities) and apply various occupancy-based evaluation schemes to obtain hydration sites. For example Henchman and McCammon (2002) define time averaged positions for this purpose. MobyWat also works with occupancy values and uses water mobility for prediction or analysis of the hydration structure.

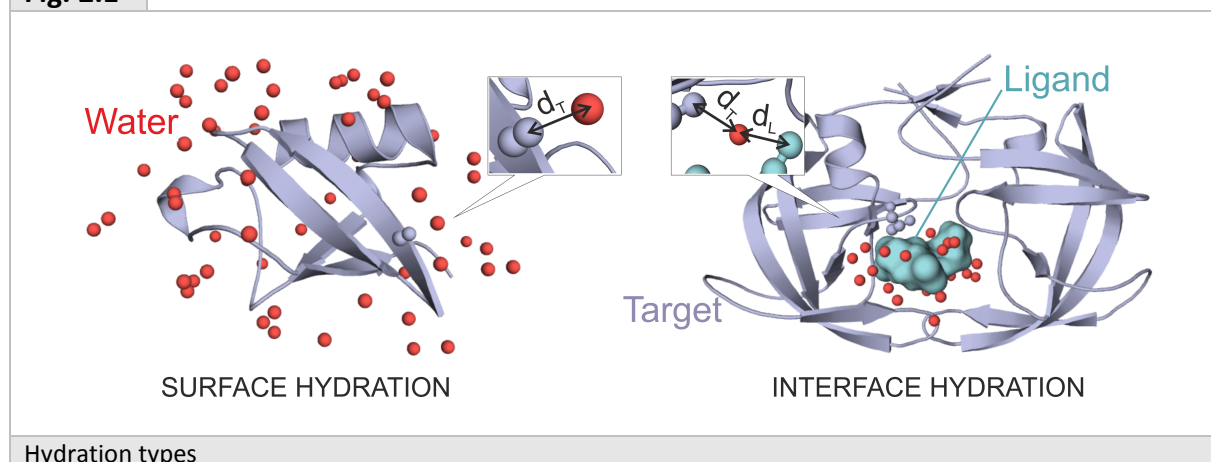


## 2 Glossary

**Frame.** Mobility of water molecules can be followed at an atomic scale by calculation methods such as molecular dynamics. As a result of a molecular dynamics run, spatial positions of all atoms are recorded at regular time steps. The set of coordinates of all atoms recorded after a time step is called a frame.

**Hydration types.** MobyWat predicts hydration of the surface (SF) of a single solute molecule (target) or the interface (IF) of a target-ligand complex. Tests of SF predictions were published (Jeszenői et al. 2015). Methods are also available for “filling” the IF region with water (Jeszenői et al. 2016a). Fig. 2.1 shows molecular situations for both hydration types.

**Fig. 2.1**



**Reference structure.** Experimental structure of a target or a target-ligand complex with the surrounding water molecules. Most of reference structures are produced by crystallography where water molecules are represented by their oxygen atoms. See also water pools.

**Solute.** A target (usually a large molecule) and optionally a ligand (usually a small molecule) surrounded by water or other solvent.

**Trajectory.** A joint collection (log-book) of a time-series of frames. A trajectory includes all four dimensional mobility information on water molecules including three Cartesian coordinates along the time dimension. Trajectories can be stored as separate or NMR-type PDB files or, preferably, in portable binary files (see Section 4.1 for details).

**Water pools.** Frames and reference structures include all water molecules at various distances from the solute. Importantly, during the generation of the frames molecular dynamics does not distinguish between bulk waters and others close to the solute molecule allowing continuous exchange of any water molecules with each-other in the system. For MobyWat evaluations it is reasonable to separate a pool of water molecules with possible structural role to distinguish them from bulk waters of no use. A maximal distance limit ( $d_{\max}$ ) is used for the distinction. In the case of surface hydration, a water molecule is selected for the reference or candidate pools if a distance ( $d_T$  in Fig. 2.1) measured between its oxygen atom and the closest heavy atom of the target satisfies Eq. 2.1.

**Eq. 2.1**

$$d_T \leq d_{\max}$$

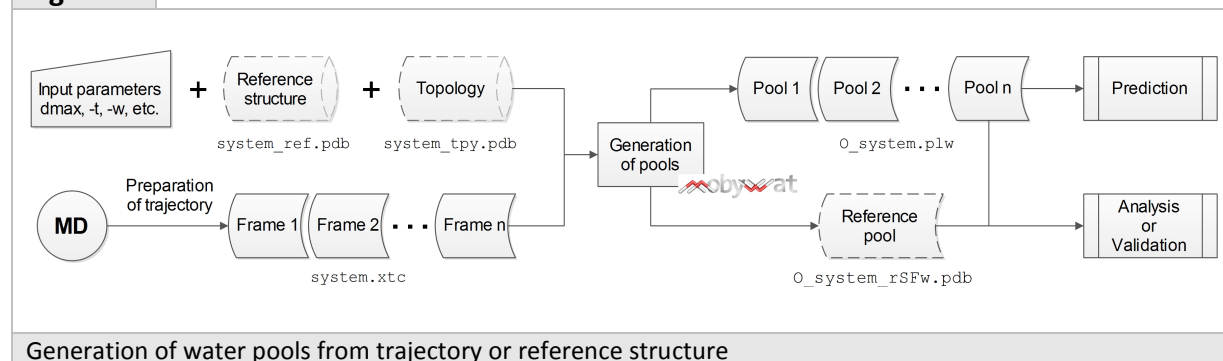
In the case of interface hydration distances measured from the closest ligand heavy atom is also considered for definition (Eq. 2.2) and selection of interface water molecules ( $d_L$  in Fig. 2.1).

### Eq. 2.2

$$d_T \leq d_{\max} \quad \text{and} \quad d_L \leq d_{\max}$$

MobyWat generates pools from experimental data (reference pool in analysis mode and validation sub-mode) or from frames of an MD trajectory (candidate pool in prediction mode and validation sub-mode) according to Fig. 2.2.

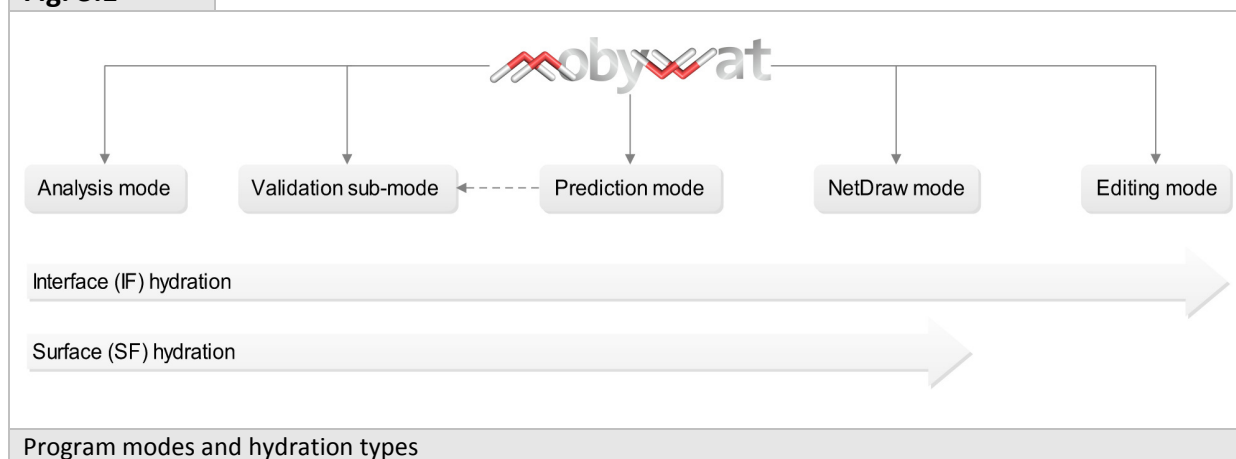
Fig. 2.2



### 3 Program modes

By default MobyWat works in Prediction mode (Section 3.2). To test the predictions and calibrate working parameters for surface (Jeszenői et al. 2015) and interface (Jeszenői et al. 2016a) hydration, a Validation sub-mode (Section 3.3) is available. MobyWat can be also used in Analysis mode (Section 3.1) to check e.g. the stability of crystallographic water positions. The Editing mode (Section 3.5) helps the prediction and the NetDraw mode (Section 3.4) can be used for network analysis of interface hydration structures (Fig. 3.1).

**Fig. 3.1**



## 3.1 Analysis mode

### 3.1.1 Overview

In analysis mode, MobyWat compares the positions of water molecules of a reference structure (see Section 2 for definition of reference and frames) with positions of water molecules from a molecular dynamics calculation. Commutability, mobility, and occupancy values are calculated for all water molecule helping an assessment of quality and stability of their experimental positions. Success rates are also calculated to estimate the quality of water positions from simulations.

### 3.1.2 Inputs

Analysis mode of MobyWat requires reference and frame structural files (Table 3.1) of the system. The solute molecules and the surrounding waters may be shifted to different spatial positions in different frames. Thus, superimposition of solute atoms of all frames to solute atoms of the reference structure is necessary to bring all molecules into the same coordinate system in the beginning of prediction process. Importantly, the position of water molecules relative to their interacting solute partners must not change during superimposition. Such superimposition or structural fit can be accomplished by most programs (GROMACS, PyMOL, LSQMAN, etc.). See also Section 9.2 for practical details of fitting frames of a trajectory.

MobyWat can print the per-frame series of root mean squared deviations (RMSD, Eq. 3.1) of a total of  $N$  backbone  $C_\alpha$  atomic vectors. This can be useful for checking the quality of fit between target molecules in the reference ( $R$ ) and in the  $n$ th frame ( $F_n$ ). An RMSD < 1.0–1.5 Å can be recommended for a meaningful analysis.

Eq. 3.1

$$\text{RMSD}_n = \sqrt{\frac{1}{N} \sum_{i=1}^N |\vec{F}_{n,i} - \vec{R}_i|^2}, \quad n = 1, 2, \dots, \text{number of frames}$$

Processing of structural reference inputs ends with separation of **reference pool** using a distance tolerance (Section 2,  $d_{\text{max}}$ ) and a B-factor limit ( $b_{\text{max}}$ , Table 3.1). This step allows exclusion of reference water molecules with large B-factors and also those which are located at a distance larger than  $d_{\text{max}}$  from the molecular surface analyzed (see also Section 2 for the role of  $d_{\text{max}}$  in the definition of surface and interface water molecules). Filtered reference water molecules are printed into a PDB file (Table 3.1).

### 3.1.3 Algorithm

MobyWat identifies water molecules in each frame closest to the selected reference waters. Distances between a reference and all water oxygen atoms in the frame are calculated and water molecules with the lowest distances are selected. The procedure is repeated for each frame and for all reference waters. Atom and residue serial numbers of the selected, closest frame oxygen atoms, as well as the corresponding distances are collected as matrices and printed into separate text files as diagnostic output (Table 3.1). Commutability, mobility, occupancy and success rate values are calculated from the matrices according to definitions of Section 3.1.4.

**Table 3.1**

ANALYSIS MODE		
INPUT FILES & DEFAULTS		
-f	system.xtc	Trajectory file (xdr binary, optional)
-f	system.plw	Pool waters file (MobyWat binary, optional)
-f	system_md1.pdb	Trajectory file (PDB NMR, optional)
-f	system_i.pdb	Trajectory file (PDB separate, i=1,2,...,#frame, optional)
-pli	system.pli	Pool information file (required if *.plw file is used)
-r	system_ref.pdb	Reference file (required)
-tpy	system_tpy.pdb	Topology file (required if *.xtc file is used)
INPUT PARAMETERS & DEFAULTS		
-bmax	75.000	B-factor limit, Å <sup>2</sup>
-dmax	3.500	Distance limit, Å
-m	Prediction	Program mode, Analysis/Prediction
-mtol	1.500	Match tolerance, Å
-n	0-10	Frame range, x-y
-v	Silent	Verbosity, Silent/Verbose/Diagnostic
INPUT RANGES		
-l	x-y/[xy...]	Ligand range, atom serial numbers/chain IDs
-t	x-y/[xy...]	Target range, atom serial numbers/chain IDs
-w	x-y/WAT/Auto	Waters range, atom serial numbers/residue name/automatic
OUTPUT FILES		
Silent output (default)		
	O_system.log	Log file
	O_system.pli	Pool information file (generated if not *.plw was used as input)
	O_system.plw	Pool waters file (generated if not *.plw was used as input)
	O_system_NoEX.txt	Commutability values
	O_system_asEX.txt	- Atom serial numbers (closest replacing waters)
	O_system_rsEX.txt	- Residue serial numbers (closest replacing waters)
	O_system_diEX.txt	- Distances (between reference and closest replacing waters)
	O_system_ocEX.txt	Mobility values
	O_system_occy.txt	Occupancy values
	O_system_rSFw.pdb	Reference water pool (surface evaluation) <sup>t</sup>
	O_system_succ.txt	Success rate values
	O_system_tpy.pdb	Topology file (generated if *.pdb was used as input)
Verbose output		
	O_system_rmst.txt	RMSD values (target-target C <sub>α</sub> )
	O_system_ref.t.pdb	Reference target coordinates
	O_system_refw.pdb	Reference waters coordinates
Diagnostic output		
	O_system_aser.txt	Atom serial numbers (closest frame waters)
	O_system_rser.txt	Residue serial numbers (closest frame waters)
	O_system_dist.txt	Distances (reference vs. closest frame waters)
	O_system_frft.pdb	Target coordinates of the first frame
	O_system_frftw.pdb	Water coordinates of the first frame
	O_system_fSFw.pdb	Pool waters in the first frame
	O_system_NoSF.txt	Number of pool waters per frame (surface evaluation)

### 3.1.4 Outcomes

#### 3.1.4.1 Commutability

During a time period, a water molecule can either remain at its starting position or be replaced by other water molecules. Commutability tells how many water molecules **of different identity** would be capable for such a replacement by coming close to the position of the reference water molecule during the simulation time and the result is printed for each reference water molecule into a text file (Table 3.1). Note that in the present version of MobyWat there is no maximal distance limit defined for the closest water molecules, the distance matrices (Table 3.1) include all distance values calculated.

#### 3.1.4.2 Mobility

For each replacing water molecules, the frequency of their occurrence during the simulation time is calculated, expressed in trajectory % and printed into a file (Table 3.1). High frequency values correspond to low mobility of a replacing water molecule.

Low commutability and mobility values may hint that a reference water is conserved.

#### 3.1.4.3 Occupancy (residence, match)

The occupancy value shows how often any water molecule approached the spatial position of a reference water molecule during the time period of the trajectory. The user can set a maximal distance tolerance (mtol, Table 3.1) to define a successful approach. Occupancy values are listed as trajectory % for all reference water molecules into a separate text file (Table 3.1). Note, that in contrast with commutability and mobility, occupancy does not account for the identity of the water molecules selected from a frame. Large occupancy value of a reference water position shows that the position is well-defined and its spatial location is probably correct and/or conserved.

#### 3.1.4.4 Success rate

Success rate quantifies the match between water positions produced by a simulation method and reference positions. The calculation is performed for all ( $n^{\text{th}}$ ) frames of the trajectory (Eq. 3.2) without clustering of the frames.

**Eq. 3.2**

$$SR_n = 100 \frac{\text{Number of matches in the } n^{\text{th}} \text{ frame}}{\text{Number of water molecules in the reference pool}} \%$$

The user can set a match tolerance (mtol, Table 3.1) as a maximum value between the oxygen atoms of the frame and reference water molecules. Success rate values are printed for all frames into a separate text file (Table 3.1). Notably, success rates are also calculated in Validation sub-mode for the prediction lists, i.e. not for the raw frames (Section 3.3.4).

### 3.1.5 Usage

#### 3.1.5.1 Sample command

Specification of trajectory file name is not necessary, if a file named **system.xtc** is placed in your working directory. The program expects that a **system\_ref.pdb** file including the reference coordinates and a topology file **system\_tpy.pdb** also exist or the user can specify an arbitrary file name using **-r** and **-tpy** (see Section 4.1 for detailed description of file types). Specification of ranges of target and waters in the trajectory frames is an obligatory part of the command (see Section 4.3 for details on input ranges).

```
$ mobywat -t [A] -w Auto -n 0-100 -m Analysis -v Diagnostic
```

### 3.1.5.2 Reference ranges

MobyWat requires definition of reference ranges of target and waters in the header of the reference coordinate file as a **REMARK** section. This is necessary as ranges in the reference file may be different from those of the trajectory defined in the command line. The entry has the following sample syntax with fixed key words **REMARK mobywat\_reference\_XXXXXX** succeeded by user-defined values as used in command line (see Section 4.3 for details).

```
REMARK mobywat_reference_target [A]  
REMARK mobywat_reference_waters Auto
```

! The Analysis mode of the program was tested for  $SR_n$  calculations only. A manuscript will be prepared with explanation & test of other quantities soon.

Test package of the Analysis mode can be downloaded from the MobyWat web site.

## 3.2 Prediction mode

### 3.2.1 Overview

The Prediction mode of MobyWat converts mobility information of water molecules into prediction of a hydration structure. Mobility information can be recorded from molecular dynamics calculations for all atoms of a system during a time period. Thus, the movements (trajectories) of all water molecules can be used in MobyWat. Various prediction schemes are implemented in MobyWat using clustering by identity information of individual water molecules or their spatial positions during along the molecular dynamics trajectory. The predictions have been validated and tested for reproduction of experimentally determined water positions of protein surfaces (Jeszenői et al. 2015) and target-ligand interfaces (Jeszenői et al. 2016a).

### 3.2.2 Inputs

Prediction mode of MobyWat requires the trajectory (frames) of a system in binary or PDB file formats (Table 3.2). Similarly to Section 3.1.2 superimposition of solute molecules is necessary as a preparatory step of prediction. In contrast with Section 3.1.2 the frames are superimposed on and RMSD values are calculated with the first frame ( $F_x$ ) instead of  $R$  in Eq. 3.1 before prediction ( $x$  is specified by  $-n\ x-y$ , Table 3.1). See also Section 9.2 for practical details of fitting frames of a trajectory.

### 3.2.3 Algorithm

MobyWat applies two approaches of clustering for conversion of mobility information into structural predictions.

**Identity-based clustering.** This type of clustering identifies a candidate water molecule by its identity (ID) numbers such as atom and residue serial numbers and uses the history of residence of each molecule on solute surface for mobility calculations.

**Position-based clustering.** In contrast with ID-based clustering, position (POS) -based clustering accounts for the history of the spatial positions occupied by water molecules irrespective of their identities. Technically, position-based clustering uses only Cartesian ( $x,y,z$ ) coordinates for representation of candidate water positions.

#### 3.2.3.1 Separation of candidate pools

Prediction mode of MobyWat starts with separation of candidate pools of water molecules from bulk waters using  $d_{\max}$  according to the definition given in Section 2. Plausibly, water molecules positioned far from the surface/interface region are of no interest for our algorithm. For each frame of the trajectory, a candidate pool of water molecules is stored (one pool per frame) in a binary file for further steps of the prediction process, and can be printed into files for program diagnostics, as well.

#### 3.2.3.2 Calculation of occupancy lists

In this step, the candidate pools created in Section 3.2.3.1 are transformed into occupancy lists (Fig. 3.3). According to the two types of clustering, two definitions for producing occupancy lists are given. Both clustering types accounts for all members of all pools. Thus, the number of frames in the trajectory is a natural upper limit of occupancy numbers in both types of occupancy lists.

**Identity-based occupancy list.** A list of water molecules occurring in at least one candidate pool of the trajectory is created, one row for each different ID. Occurrence of a molecule in



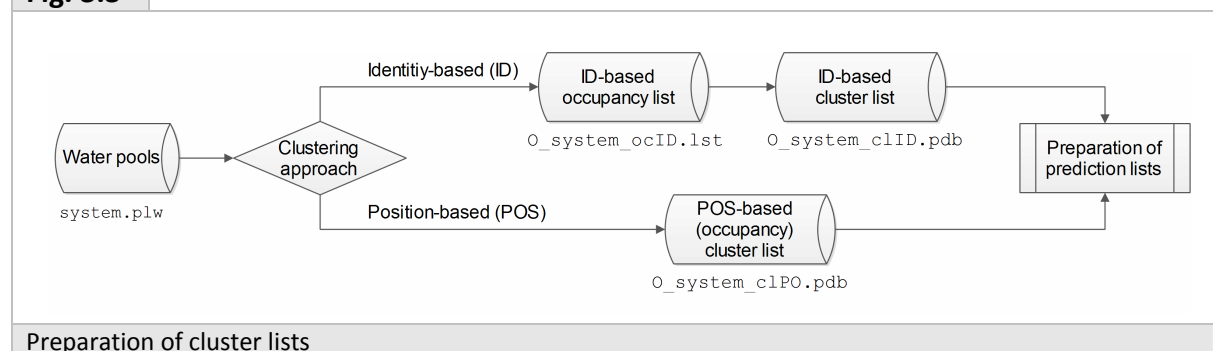
the pools with the same ID is counted during the whole trajectory and the count is registered in the list as an occupancy number corresponding to the ID. That is, the value of an occupancy number in the list is increased if a pool includes the water with the ID in question. After evaluating all candidate pools, occupancy lists are sorted by decreasing occupancies. An identity-based occupancy list can be considered as a result of clustering along the time dimension.

**Position-based (occupancy) cluster list.** A list of positions of water molecules occurring in the first candidate pool of the trajectory is created, one row for each different position. During evaluation of all pools, the occupancy number is increased by one, if the position of a water molecule of a pool is located closer to a previously listed position than a pre-defined clustering tolerance (ctol, Table 3.2). If such a position is found, an average of the previously listed and newly found positions will be calculated and used for comparison with the next pool, providing a dynamic (averaged) position definition. Notably, a close water position in a pool can increase occupancy of only one position of the occupancy list per frame (very large ctol values could allow increase of more than one list members, which is not desirable). If the distance between the position of a water molecule of a pool and a previously listed position is larger than or equal to ctol, then a new position (row) of the occupancy list is created. After evaluating all pools, occupancy lists are sorted by decreasing occupancies. A position-based occupancy list can be considered as a complete cluster list, based on time-dependent spatial mobility information.

### 3.2.3.3 Completion of identity-based clustering

Identity-based occupancy list represents distillation of the candidate pools of a trajectory along time dimension. A further, spatial clustering step was introduced to complete this type of clustering. Water molecules of different pools with the same ID (belonging to the same row of the occupancy list) are collected into a cluster using a pre-defined ctol value. That is, all water molecules in the cluster must have a maximal distance of ctol from each-other. In this way, all members of a row of the occupancy list are clustered, and the procedure is repeated for all rows of the list. Finally, the clusters are ordered by the count of their members and the average of the coordinates of the members is calculated for each cluster resulting in a representing entry for the identity-based cluster list.

**Fig. 3.3**



### 3.2.3.4 Calculation of prediction lists

As a final step, MobyWat creates prediction lists from the cluster lists described in the previous Sections. Prediction lists contain the atomic coordinates of water positions and the corresponding mobility (M) values as final outcomes of the prediction process. A mobility (M) value is calculated for each row of the prediction lists from normalized occupancy (O) values (Eq. 3.3). Normally, M values scale between 0 and 100 and zero corresponds to the least mobile, conserved predicted water position. Predicted water positions are listed in

order of increasing mobility in the prediction list. MobyWat produces four types of prediction lists (Fig. 3.4).

**Eq. 3.3**

$$O_i = \frac{\text{Occupancy of cluster}}{\text{Number of pools}}, \quad \text{where } i = 1, 2, \dots, \text{number of rows of prediction list.}$$

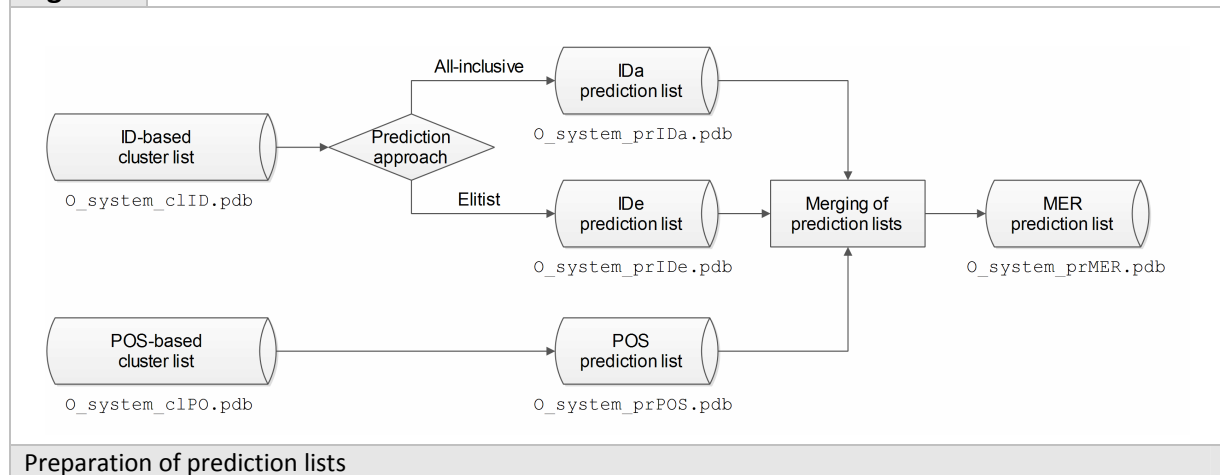
$$M_i = 100 \begin{cases} \frac{O_{\max} - O_i}{O_{\max} - O_{\min}}, & \text{if } O_{\max} > O_{\min}, \\ 1 - O_i, & \text{if } O_{\max} = O_{\min}, \end{cases} \quad \text{where } O_{\max/\min} = \max/\min O_i.$$

**Elitist prediction list (IDe).** This prediction list is produced from identity-based occupancy and cluster lists. The first clusters of each row of the occupancy list are used in the first round of the evaluation process. The largest first cluster is selected from among all rows of the occupancy list and placed at the top of the prediction list. Other first clusters are checked if the distance between their representing water position and that of the largest cluster is smaller than a pre-defined prediction tolerance (ptol) value. If the distance is smaller than ptol then the cluster is disqualified. Notably, this tolerance ensures to keep a minimal distance between the resulted members of the prediction list to avoid close contacts.

In the next step, the largest first cluster is selected again from among the clusters qualified in the first step and it is placed at the next position of the prediction list. The above procedure is repeated comparing prediction list members with available first clusters until first clusters of all rows of the occupancy list were either placed on the prediction list or disqualified. In the case if the first cluster of a row of the occupancy list was not placed on the prediction list, the procedure goes on with the second and higher clusters until all rows of occupancy list contributed a cluster to the prediction list.

This prediction list is called elitist as it uses up the first clusters of each row of the occupancy list first to fill up the prediction list.

**Fig. 3.4**



**All-inclusive prediction list (IDa).** This prediction list is produced from list of identity-based clusters irrespective of their location on the occupancy list. The largest cluster is selected from among all clusters and placed on the top of the prediction list. The other clusters are checked if the distance between their representing spatial water position and that of the

largest cluster is smaller than  $ptol$ . If the distance is smaller than  $ptol$  then the cluster is disqualified.

In the next step, the largest cluster is selected again from among the clusters qualified in the first step and it is placed on the next position of the prediction list. The above procedure is repeated comparing prediction list members with available clusters until all clusters were either placed on the prediction list or disqualified.

**Position-based prediction list (POS).** This prediction list is produced from position-based occupancy list, using the procedure detailed at “All-inclusive prediction list”.

**Merged prediction list (MER).** Merging ( $\cup$ ) of prediction lists is performed in a pair-wise manner. Two prediction lists are simply copied into the merged prediction list one after the other. As a final step, the merged prediction list is cleaned up by removing entries with a distance smaller than  $ptol$  with the procedure described at “All-inclusive prediction list”.

MobyWat creates merged prediction list from the three available prediction lists in the following order (Eq. 3.4).

**Eq. 3.4**

$$MER = (IDa \cup IDe) \cup POS$$

**Table 3.2**

PREDICTION MODE		
INPUT FILES & DEFAULTS		
-f	system.xtc	Trajectory file (xdr binary, optional)
-f	system.plw	Pool waters file (MobyWat binary, optional)
-f	system_md1.pdb	Trajectory file (PDB NMR, optional)
-f	system_i.pdb	Trajectory file (PDB separate, i=1,2,...,#frame, optional)
-pli	system.pli	Pool information file (required if *.plw file is used)
-tpy	system_tpy.pdb	Topology file (required if *.xtc file is used)
INPUT PARAMETERS & DEFAULTS		
-cls	IDa	Clustering algorithm, IDa/IDe/POS/MER
-ctol	1.000	Clustering tolerance, Å
-dmax	3.500	Distance limit, Å
-m	Prediction	Program mode, Analysis/Prediction
-n	0-10	Frame range, x-y
-ptol	2.500	Prediction tolerance, Å
-top	50.000	Top cut of short prediction list, %
-v	Silent	Verbosity, Silent/Verbose/Diagnostic
INPUT RANGES		
-l	x-y/[xy...]	Ligand range, atom serial numbers/chain IDs
-t	x-y/[xy...]	Target range, atom serial numbers/chain IDs
-w	x-y/WAT/Auto	Waters range, atom serial numbers/residue name/automatic
OUTPUT FILES		
Silent output (default)		
	O_system.log	Log file
	O_system.pli	Pool information file (generated if not *.plw was used as input)
	O_system.plw	Pool waters file (generated if not *.plw was used as input)
	O_system_prXXX.pdb	Long (full) prediction list (XXX = IDa/IDe/POS/MER)
	O_system_prXXX_top.pdb	Short (top-cut) prediction list (XXX = IDa/IDe/POS/MER)
	O_system_tpy.pdb	Topology file (generated if *.pdb was used as input)
Verbose output		
	O_system_rmst.txt	RMSD values (target-target C <sub>α</sub> )
	O_system_clID.pdb	Identity-based cluster list (for IDa/IDe/MER)
	O_system_clPO.pdb	Position-based cluster list (for POS/MER)
	O_system_ocID.lst	Identity-based occupancy list (for IDa/IDe/MER)
	O_system_NoSF.txt	Number of pool waters per frame (surface evaluation)
Diagnostic output		
	O_system_cSFw.pdb	List of clustered waters (surface evaluation)
	O_system_frft.pdb	Target coordinates of the first frame
	O_system_frww.pdb	Water coordinates of the first frame
	O_system_fSFw.pdb	Pool waters in the first frame

### 3.2.4 Outcomes

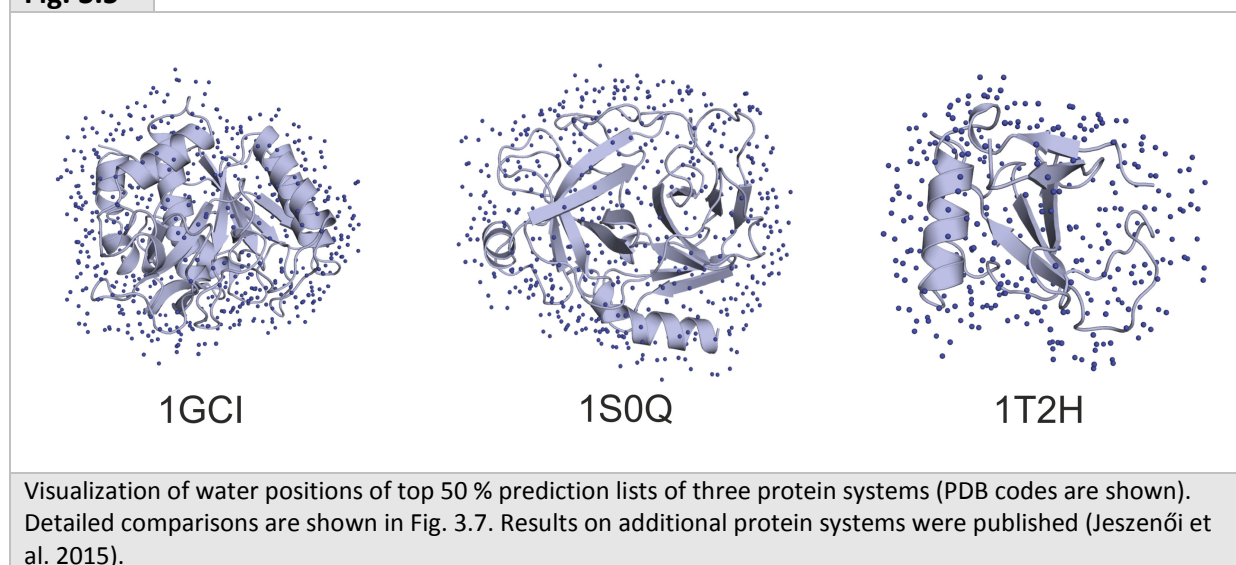
Primary outcomes of Prediction mode of MobyWat are prediction list. The lists are printed in standard PDB format and contain coordinates of oxygen atoms of predicted water positions (Fig. 3.5) in increasing order of mobility. Both types of mobility values of Eq. 3.3 are printed in crystallographic occupancy and B-factor columns, respectively.

The type of requested prediction lists can be specified by the user with the **-cls** switch. Notably, in the case of **-cls MER** all four (IDa, IDe, POS, MER) prediction lists are calculated and printed. Main parameters **ctol** and **ptol** can be also specified by the user at the command line.

Short (top-cut) version of the prediction lists are also printed including the top part (the least mobile) waters from the full prediction lists. The length of a short list can be specified with a cut switch **-top**. For most applications, the use of short prediction lists including the top 50 % (default in Table 3.2) and top 25 % of the predicted water molecules is recommended for SF prediction of the entire surface of protein molecules and of the binding (active) sites, respectively. For details on scoring performance of short-lists of water molecules, please refer to Section 3.3.4. In the case of IF predictions (Jeszenői et al. 2016a), full prediction lists may be used.

Other files listed in Table 3.2 such as occupancy and cluster lists, number of pool waters, etc. can be also printed by setting **-v** to **Verbose** or **Diagnostic**.

**Fig. 3.5**



### 3.2.5 Usage

#### Sample commands

Specification of trajectory file name is not necessary, if a file named **system.xtc** is placed in your working directory. The program expects that a topology file **system\_tpy.pdb** also exist or the user can specify an arbitrary file name using **-tpy** (see Section 4.1 for detailed description of file types). Program mode is **Prediction** by default, and therefore, use of **-m** is not necessary here. Specification of ranges of target and waters in the trajectory frames is an obligatory part of the command (see Section 4.3 for details on input ranges). The following commands can be used for different input situations.

Default trajectory and topology input files (**system.xtc** and **system\_tpy.pdb**), frames between 0 and 100, target is specified by its chain ID, merged prediction list is requested:

```
$ mobywat -t [A] -w Auto -n 0-100 -cls MER
```

Default trajectory and topology input files (**system.xtc** and **system\_tpy.pdb**), frames between 0 and 100, target is specified by range of atom serial number, merged prediction list is requested:

```
$ mobywat -t 1-925 -w Auto -n 0-100 -cls MER -m Prediction
```

Pool waters input file with corresponding pool information file (**O\_system.pli**), merged prediction list is requested, other settings are read from pool information file:

```
$ mobywat -f O_system.plw -cls MER
```

NMR-type PDB file (**2FMA\_md1.pdb**), other options are as above:

```
$ mobywat -f 2FMA_md1.pdb -t [A] -w Auto -n 1-100 -cls MER
```

Separate PDB files with root name “**2FMA**”, other options are as above:

```
$ mobywat -f 2FMA.pdb -t [A] -w Auto -n 1-100 -cls MER
```

For interface prediction Methods 1 and 3 (Jeszenői et al. 2016a) the range of the ligand molecule can be defined using an additional **-l** switch:

```
$ mobywat -m Prediction -f system.xtc -tpy system_tpy.pdb -t [A] -l [B] -w Auto -n 0-1000 -ctol 1 -ptol 2.5 -dmax 3.5
```

Further description of Methods 1 and 3 can be found in Section 9.2.

Test packages of the Prediction mode can be downloaded from the MobyWat web site.

## 3.3 Validation sub-mode

### 3.3.1 Overview

The validation sub-mode of MobyWat can be used for test and calibration of clustering algorithms implemented in Prediction mode and also in search for optimal molecular dynamics simulation parameters (Jeszenői 2016b). Validation sub-mode is mostly based on the algorithms of Prediction mode. Validation sub-mode can be recommended for testing new force fields and explicit water models (Jeszenői 2016b).

### 3.3.2 Inputs

Similarly to Analysis mode, both reference frame file and trajectory file are required. From the reference file, a reference water pool is separated and used for comparison with the results of predictions. The reference file is usually a crystallographic structure from the PDB with several water molecules assigned.

### 3.3.3 Algorithm

A detailed description of prediction algorithms were given in Section 3.2. In addition, validation sub-mode uses a match tolerance (mtol) for comparison of the location of water oxygen atoms in the reference pool and in the prediction lists. A match is defined in the prediction list if the distance between the reference and predicted oxygen atoms is smaller than mtol. Validation sub-mode takes all members of each prediction lists and checks if a member has a match with the reference pool or not. All members of the reference pool can be used only once for each prediction list during identification of matches.

### 3.3.4 Outcomes

**Match lists.** The results are saved in match lists (text files, Fig. 3.6) including water serial numbers, occupancy counts and the distances used for identification of a match, i.e. comparison with mtol. In the last column “M” of the match list “x” marks a match (Fig. 3.7).

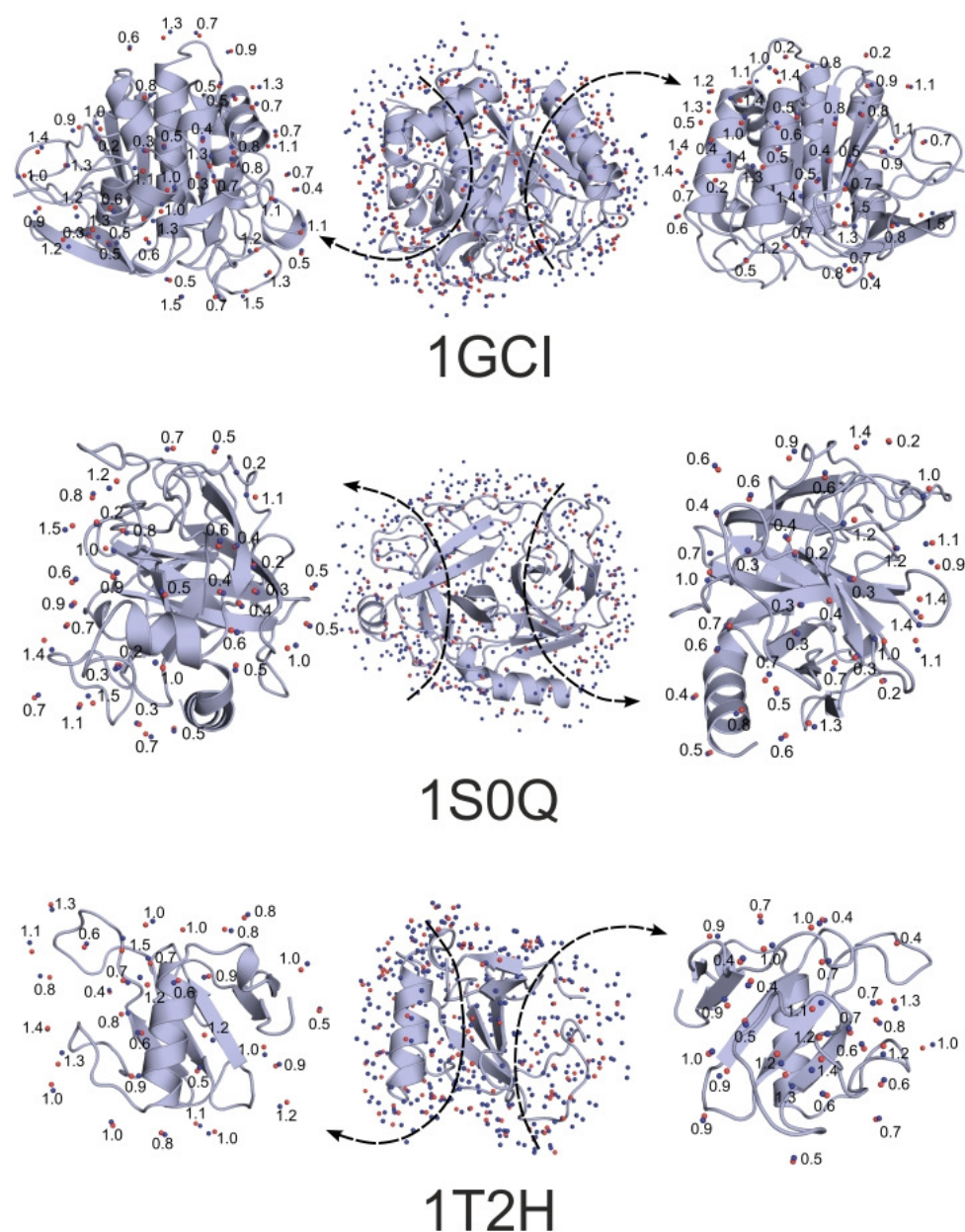
**Fig. 3.6**

Match List between ID-all-inclusive Prediction List and Reference List										
First frame #	:	0								
Last frame #	:	10000								
Match tolerance (A)	:	1.500								
B-factor limit	:	30.000								
Distance tolerance (A)	:	3.500								
Prediction tolerance (A)	:	2.500								
Clustering tolerance (A)	:	1.000								
=====										
Water (pl) #	Water (ol) #	Cluster (ol) #	Count		Ref. #	Ref. atom#	Ref. res. #	B-factor	Distance	M
-----										
1	1	1	9385		13	1706	614	7.64	0.144	x
2	2	1	9187		16	1709	617	8.62	0.352	x
3	4	1	8913		20	1713	621	9.18	0.796	x
4	5	1	5645		36	1731	639	11.76	0.672	x
5	3	1	4156		82	1779	687	16.50	7.437	-
6	7	1	3305		42	1737	645	12.13	0.225	x
7	8	1	1424		49	1744	652	12.50	0.798	x
8	11	1	1131		24	1717	625	10.48	0.171	x
✂-----✂										
649	3040	176	1		122	1831	739	21.82	6.217	-
650	3106	176	1		39	1734	642	12.05	3.804	-
651	4884	56	1		54	1749	657	13.77	1.508	-
=====										
SR	78.981									
=====										

Match list of PDB system 1T2H (abridged)



**Fig. 3.7**



Predicted water positions (blue spheres) of three protein systems matching with crystallographic reference water positions (red spheres). 83.9, 83.7, and 79.0 % of the crystallographic reference positions were found, respectively. PDB codes are shown. Matching pairs and distances (Å) are shown for clarity. Results on additional protein systems were also published (Jeszenői et al. 2015).



**Table 3.3**

VALIDATION SUB-MODE		
INPUT FILES & DEFAULTS		
-f	system.xtc	Trajectory file (xdr binary, optional)
-f	system.plw	Pool waters file (MobyWat binary, optional)
-f	system_md1.pdb	Trajectory file (PDB NMR, optional)
-f	system_i.pdb	Trajectory file (PDB separate, i=1,2,...,#frame, optional)
-pli	system.pli	Pool information file (required if *.plw file is used)
-r	system_ref.pdb	Reference file (required)
-tpy	system_tpy.pdb	Topology file (required if *.xtc file is used)
INPUT PARAMETERS & DEFAULTS		
-cls	IDa	Clustering algorithm, IDa/IDe/POS/MER
-ctol	1.000	Clustering tolerance, Å
-dmax	3.500	Distance limit, Å
-m	Prediction	Program mode, Analysis/Prediction
-mtol	1.500	Match tolerance, Å
-n	0-10	Frame range, x-y
-ptol	2.500	Prediction tolerance, Å
-top	50.000	Top cut of short prediction list, %
-v	Silent	Verbosity, Silent/Verbose/Diagnostic
INPUT RANGES		
-l	x-y/[xy...]	Ligand range, atom serial numbers/chain IDs
-t	x-y/[xy...]	Target range, atom serial numbers/chain IDs
-w	x-y/WAT/Auto	Waters range, atom serial numbers/residue name/automatic
OUTPUT FILES		
Silent output (default)		
	O_system.log	Log file
	O_system.pli	Pool information file (generated if not *.pli was used as input)
	O_system.plw	Pool waters file (generated if not *.plw was used as input)
	O_system_prXXX.pdb	Long (full) prediction list (XXX = IDa/IDe/POS/MER)
	O_system_prXXX_top.pdb	Short (top-cut) prediction list (XXX = IDa/IDe/POS/MER)
	O_system_tpy.pdb	Topology file (generated if *.tpy was used as input)
	O_system_mtXXX.lst	Match list (XXX = IDa/IDe/POS/MER)
	O_system_i_XXX.mat	Success rate matrix (XXX = IDa/IDe/POS/MER, i=1,2,...,num_mtol)
	O_system_rSFw.pdb	Reference water pool (surface evaluation) <sup>t</sup>
Verbose output		
	O_system_rmst.txt	RMSD values (target-target C <sub>α</sub> )
	O_system_clID.pdb	Identity-based cluster list (for IDa/IDe/MER)
	O_system_clPO.pdb	Position-based cluster list (for POS/MER)
	O_system_ocID.lst	Identity-based occupancy list (for IDa/IDe/MER)
	O_system_NoSF.txt	Number of pool waters per frame (surface evaluation)
	O_system_mtIDc.lst	Cluster match list (for IDa/IDe/MER)
	O_system_reft.pdb	Reference target coordinates
	O_system_refw.pdb	Reference waters coordinates
Diagnostic output		
	O_system_cSFw.pdb	List of clustered waters (surface evaluation)
	O_system_frft.pdb	Target coordinates of the first frame
	O_system_frww.pdb	Water coordinates of the first frame
	O_system_fSFw.pdb	Pool waters in the first frame

**Success rate (SR).** From the match lists SR values are calculated for each prediction (list) according to Eq. 3.5. The higher the SR value, the more successful a prediction is in comparison with crystallographic water positions (Fig. 3.7).

**Eq. 3.5**

$$SR_{xxx} = 100 \frac{\text{Number of matches in the XXX prediction list}}{\text{Number of water molecules in the reference pool}} \%,$$

where XXX = IDa/IDe/POS/MER.

**SR matrices.** Besides evaluation of matches on a single prediction list at fix (mtol, ptol, ctol) tolerance values, validation sub-mode of MobyWat can scan user-defined ranges of all three tolerance values and print the resulted SR values in (ptol,ctol) matrices for any mtol values. For this, MobyWat re-generates cluster and prediction lists as many times as required by the user and collects the resulted SR values into matrices. This option of the program is useful for the optimization of ptol and ctol values of the prediction algorithm (Section 3.2) and can be used as a development tool.

**Score performance (SP).** For many applications a short, top-cut version of the prediction lists (Section 3.2.4) can be useful excluding e.g. loosely bound, bulk-like waters and keeping only essential, structural ones. The quality of predicted water molecules at the top of the prediction list is expressed by the score performance ( $SP_x$ , Eq. 3.6). SP values are not calculated directly by MobyWat, but they can be easily obtained from the match lists (Fig. 3.6).

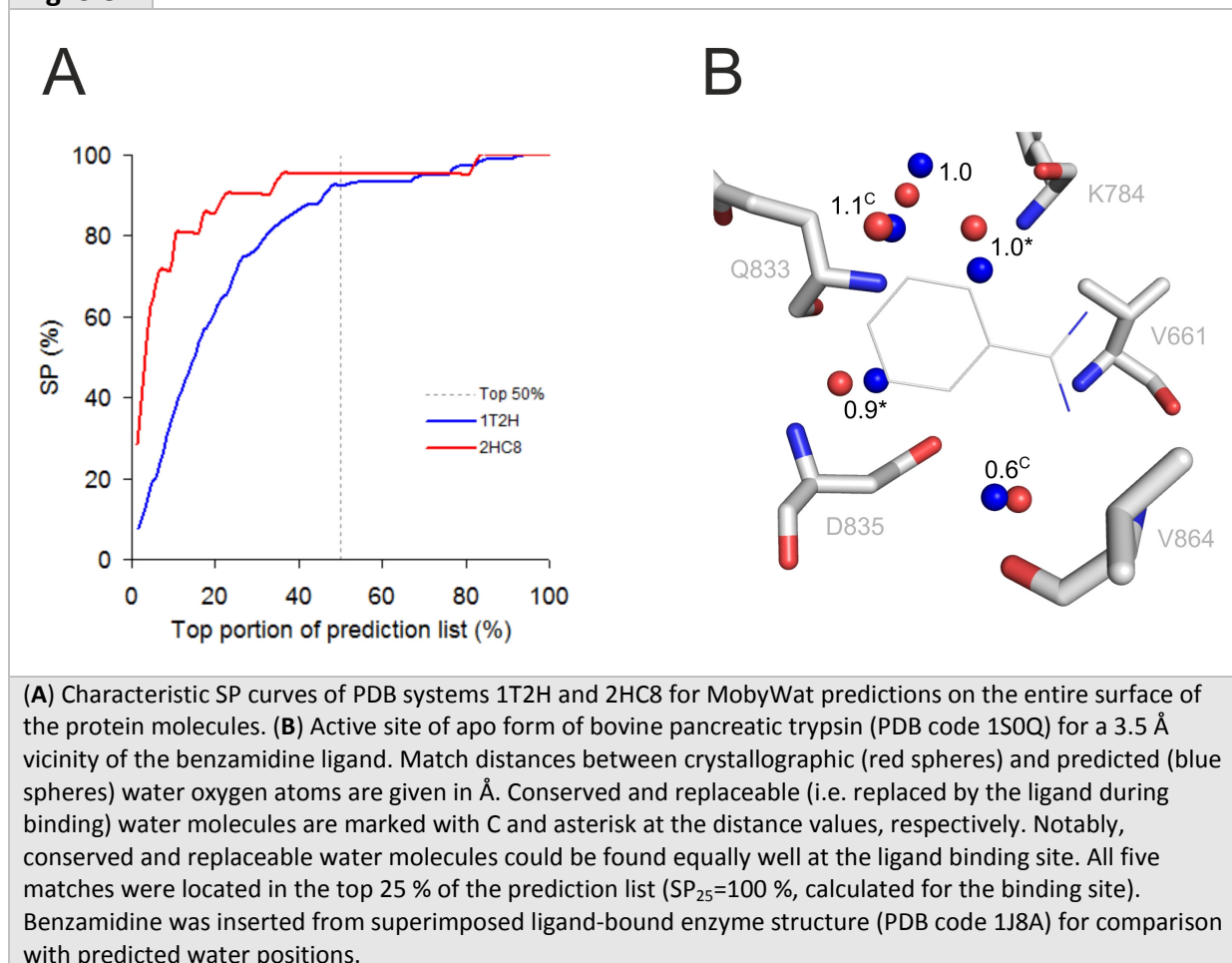
**Eq. 3.6**

$$SP_x = 100 \frac{\text{Number of matches in the top } x \% \text{ of the prediction list}}{\text{Number of matches in the full prediction list}} \%.$$

MobyWat provides water positions in an increasing order of their calculated M (Eq. 3.3) value in the prediction list. In other words, M is used as a score of the predicted water positions. SP quantifies the performance of M in placing matched positions to the top of the prediction list.

Characteristic SP curves (Fig. 3.8A) are useful for estimation of the top cut-off value for production of the short version of prediction lists. Fig. 3.8A shows that in cases of PDB systems 1T2H and 2HC8, the default short prediction list (top 50%, Table 3.3) contains more than 90 % of the matched positions, indicating that the full prediction list can be shortened to its top half without a big loss of valuable water positions. SP curves of other systems were also calculated (Jeszenői et al. 2015).

It was found (Jeszenői et al. 2015) that SP values are remarkably larger in the vicinity of the ligand binding sites than for the entire protein surface. For example, in the case of the benzamidine binding site of bovine pancreatic trypsin, an  $SP_{25}=100$  % was achieved (Fig. 3.8B), whereas an  $SP_{25}=70.6$  % was obtained for the entire protein surface. Thus, MobyWat score shows an excellent performance for active sites of protein surfaces. Thus, the use of a reduced prediction list including the top 25 % of the full prediction list can be used for prediction of hydration structure active sites. Other examples of the performance of MobyWat for active site hydration are included in the MobyWat publication (Jeszenői et al. 2015). Notably, the use of MobyWat for hydration of the protein-ligand binding interface is described in another publication (Jeszenői et al. 2016a).

**Fig. 3.8**

### 3.3.5 Usage

#### 3.3.5.1 Sample command

MobyWat automatically distinguishes between prediction and validation. Validation sub-mode is switched on if MobyWat finds a reference file in the working directory.

Specification of trajectory file name is not necessary, if a file named **system.xtc** is placed in your working directory. The program expects that a **system\_ref.pdb** file including the reference coordinates and a topology file **system\_tpy.pdb** also exist or the user can specify an arbitrary file name using **-r** and **-tpy** (see Section 4.1 for detailed description of file types). Specification of ranges of target and waters in the trajectory frames is an obligatory part of the command (see Section 4.3 for details on input ranges).

```
$ mobywat -t [A] -w Auto -n 0-100 -m Prediction -cls IDa
```

#### 3.3.5.2 Reference ranges and SR matrix settings

MobyWat requires definition of reference ranges of target and waters in the header of the reference coordinate file as a **REMARK** section. This is necessary as ranges in the reference file may be different from those of the trajectory defined in the command line. The entry has the following sample syntax with fixed key words **mobywat\_reference\_XXXXXX** succeeded by user-defined values as used in command line (see Section 4.3 for details).

The user can also request calculation of SR matrices described in Section 3.3.4. For this, additional lines are required in the **REMARK** section starting with **mobywat\_** defining the number of matrices (**num\_mtol**) and the number of data of the matrices (**num\_ptol** × **num\_ctol**). All these **num\_XXX** values have to be at least 1 to get the matrices requested. Entries **min\_XXX** and **step\_XXX** define the minimum value and step size of a parameter (mtol, ptol, or ctol).

```
REMARK mobywat_reference_target [A]
REMARK mobywat_reference_waters Auto
REMARK mobywat_min_ctol      1.0
REMARK mobywat_num_ctol      4
REMARK mobywat_step_ctol     0.50
REMARK mobywat_min_ptol      2.50
REMARK mobywat_num_ptol      1
REMARK mobywat_step_ptol     0
REMARK mobywat_min_mtol      1.500
REMARK mobywat_num_mtol      1
REMARK mobywat_step_mtol     0
```

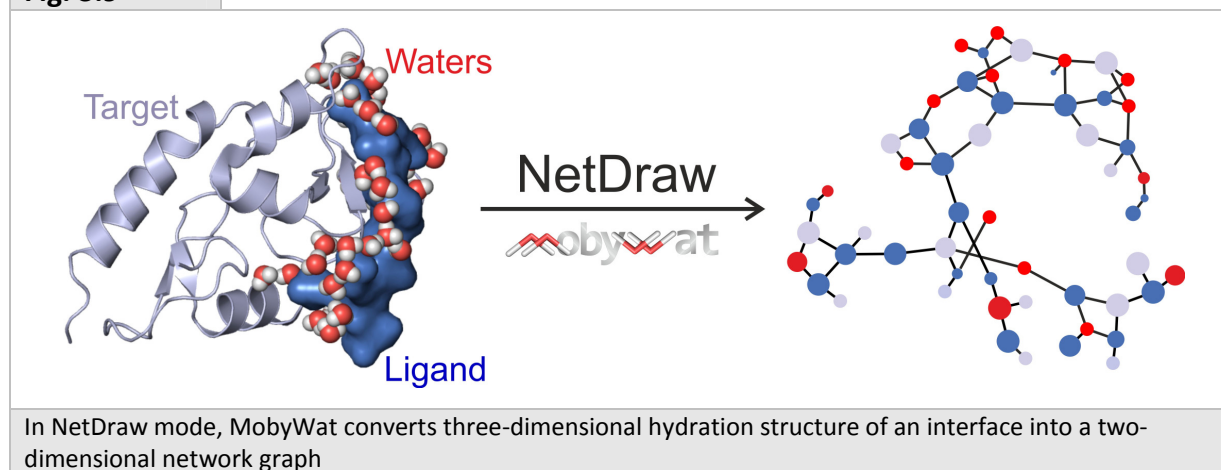
Test package of the Analysis mode can be downloaded from the MobyWat web site.

## 3.4 NetDraw mode

### 3.4.1 Overview

The NetDraw mode of MobyWat is a tool for drawing and characterizing the hydration network of a target surface or a target-ligand interface. Using an optimized three-dimensional structure file with solute and water molecules as an input, MobyWat produces the two-dimensional hydration network graph of the surface or interface (Fig. 3.9) as lists of edges and nodes. Using calculated mobility information (Section 3.2.3.4) MobyWat detects static and dynamic sub-networks of the network graph which can be used for characterization of integrity of the network, prediction of complex stability (Jeszenői 2016a), etc. MobyWat also provides the graph in commonly used formats readable by network visualization and analysis programs.

Fig. 3.9



### 3.4.2 Inputs

The NetDraw mode requires a single PDB file including optimized hydration structure of the surface/interface, bulk waters (optional), and solute molecules including target and/or ligand.

### 3.4.3 Algorithm

- 1) NetDraw detects and lists atomic pairs of the partner groups (ligand, target, water, bulk) with heavy atom distances up to **dmax** (Table 3.4). Lists are stored in distance files. Here, group water refers to interfacial water molecules.
- 2) List of edges of the graph is distilled from the distance files by eliminating redundancies and distances (edges) to carbon atoms (C-filtering).
- 3) Number of edges per node is limited to four, using the top four shortest edges only (4-filtering).
- 4) List of nodes is produced from the list of edges.
- 5) Classification of the nodes and edges. A node is classified **static** if it
  - is a solute (ligand/target) node or
  - is connected to four nodes of any type or
  - is connected to at least three solute nodes or
  - has a mobility  $\leq$  **mmax**.

Otherwise the node is **dynamic**. An edge is **static** if it connects two static nodes, otherwise it is **dynamic**.

**Table 3.4**

NETDRAW MODE		
INPUT FILES & DEFAULTS		
-f	system.pdb	Single PDB file
INPUT PARAMETERS & DEFAULTS		
-dmax	3.500	Distance limit, Å
-m	Prediction	Program mode, Analysis/Prediction/NetDraw/Editing
-mmax	50.000	Mobility tolerance
INPUT RANGES		
-l	x-y/[xy...]	Ligand range, atom serial numbers/chain IDs
-t	x-y/[xy...]	Target range, atom serial numbers/chain IDs
-w	WAT/Auto	Waters range, residue name/automatic
OUTPUT FILES		
	O_system.log	Log file
	O_system_water_target.dst	Distance file
	O_system_water_ligand.dst	Distance file
	O_system_water_water.dst	Distance file
	O_system_water_bulk.dst	Distance file
	O_system_all.nds	Node file
	O_system_all.dgs	Edge file
	O_system_all.dot	Network file (dot format)
	O_system_all.gdf	Network file (gdf format)
	O_system_wo_bulk.dgs	Edge file (without edges to bulk water nodes)
	O_system_wo_bulk.nds	Node file (without bulk water nodes)
	O_system_wo_bulk.dot	Network file (dot format, without bulk water nodes and edges)
	O_system_wo_bulk.gdf	Network file (gdf format, without bulk water nodes and edges)
	O_system_org.pdb	Original (input) PDB file

### 3.4.4 Outcomes

Lists of nodes and edges are stored in edge and node files (Table 3.4), respectively. A section of a node file is shown below. It includes the name of the node, residue serial number, type of the node (t/l/w/b refers to target/ligand/interface water/bulk water), number of edges, [number of edges to solute \(only for w nodes\)](#), mobility, classification of the node (s/d refers to static/dynamic).

```

A_GLN 383    t  1    0.000 s
A_ASP 384    t  5    0.000 s
A_SER 386    t  1    0.000 s
  WAT 1      w  3    3    0.000 s
  WAT 2      w  4    3    0.990 s
  WAT 3      w  2    1    2.760 s

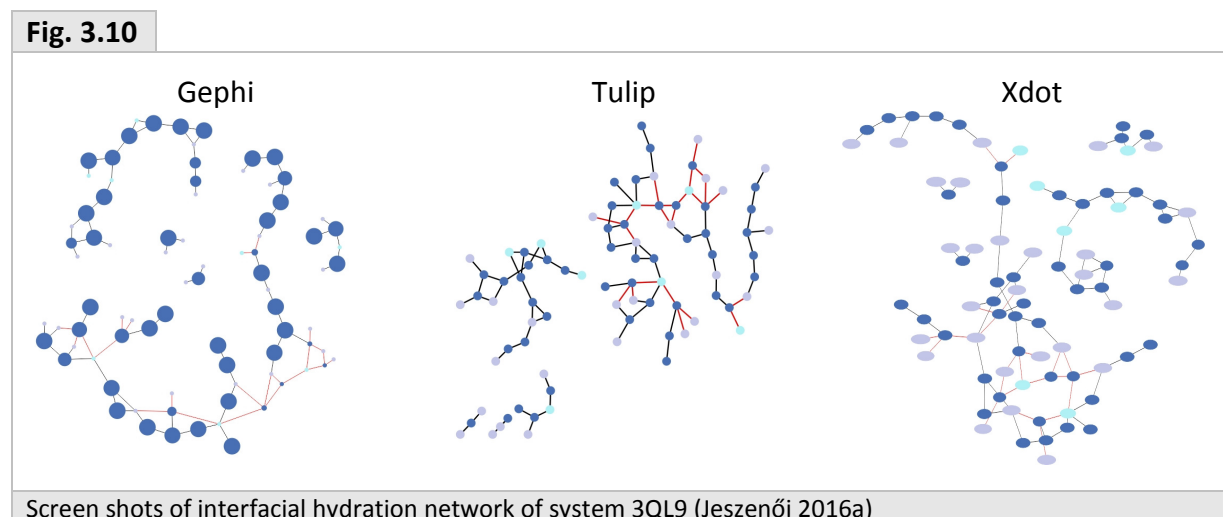
```

The edge files combine information from the distance and node files as shown in the following sample. A row starts with atom serial number, residue name, residue serial number and mobility of the interfacial water and then the same identifiers of partner node are printed. Type of partner node, distance (length of edge) in Å, and s/d classifiers of water node, partner node, and edge can be found at the end of a row.

List of edges in the hydration network									
WATER: 1	WAT 1	0.00	PARTNER: 599	A_LEU 215	0.00	t	DISTANCE:	2.665	s s s
WATER: 1	WAT 1	0.00	PARTNER: 3985	B_GLN 68	0.00	l	DISTANCE:	2.832	s s s
WATER: 1	WAT 1	0.00	PARTNER: 611	A_ASP 216	0.00	t	DISTANCE:	2.939	s s s
WATER: 2	WAT 2	0.99	PARTNER: 4659	B_THR 111	0.00	l	DISTANCE:	2.641	s s s
WATER: 2	WAT 2	0.99	PARTNER: 133	WAT 133	95.70	w	DISTANCE:	2.800	s d d
WATER: 2	WAT 2	0.99	PARTNER: 4610	B_ASN 108	0.00	l	DISTANCE:	2.976	s s s

NetDraw also produces network files readable by commonly used network visualization and analysis programs (Table 3.5) tested with **.dot** and **.gdf** input files. Screenshots of the visualization programs for the same hydration network are shown in Fig. 3.10.

Table 3.5		
NETWORK VISUALIZATION AND ANALYSIS PROGRAMS TESTED		
Name	File type (extension)	Web page
Gephi	.gdf, .dot	<a href="http://gephi.github.io/">http://gephi.github.io/</a>
Tulip	.dot	<a href="http://tulip.labri.fr/TulipDrupal/">http://tulip.labri.fr/TulipDrupal/</a>
Xdot	.dot	<a href="https://github.com/jrfonseca/xdot.py">https://github.com/jrfonseca/xdot.py</a>



In Fig. 3.10, static and dynamic edges are marked with **red** and black colors. A uniform coloring scheme is used for **ligand**, **target** and **water** nodes. All three programs can visualize the files produced with NetDraw (Fig. 3.10). In our analyses (Jeszenői 2016a), Gephi was thoroughly tested with gdf input. In this case, node size was scaled by mobility values using the “ranking” tab of the program. The best view of the graph could be obtained by a “Force Atlas + attraction distribution + adjust size” setting.

### 3.4.5 Usage

NetDraw mode requires the definition of surface/interface and bulk water molecules in the PDB file. If **-w Auto** is specified in the command line, residue name **WAT** will be used for surface/interface water molecules and **SOL** for bulk. At the **-w** switch, the user can define any three-letter residue name for surface/interface waters. In this case MobyWat assumes that bulk waters are marked with one of the H2O, HOH, SOL or WAT residue names not matching the user-defined residue name. In the following sample command, the target molecule is in chain **A**, whereas ligand is in chains **B** and **C**. The usage of text **-m NetDraw** is obligatory to initialize the present NetDraw mode of MobyWat (the default mode is Prediction).



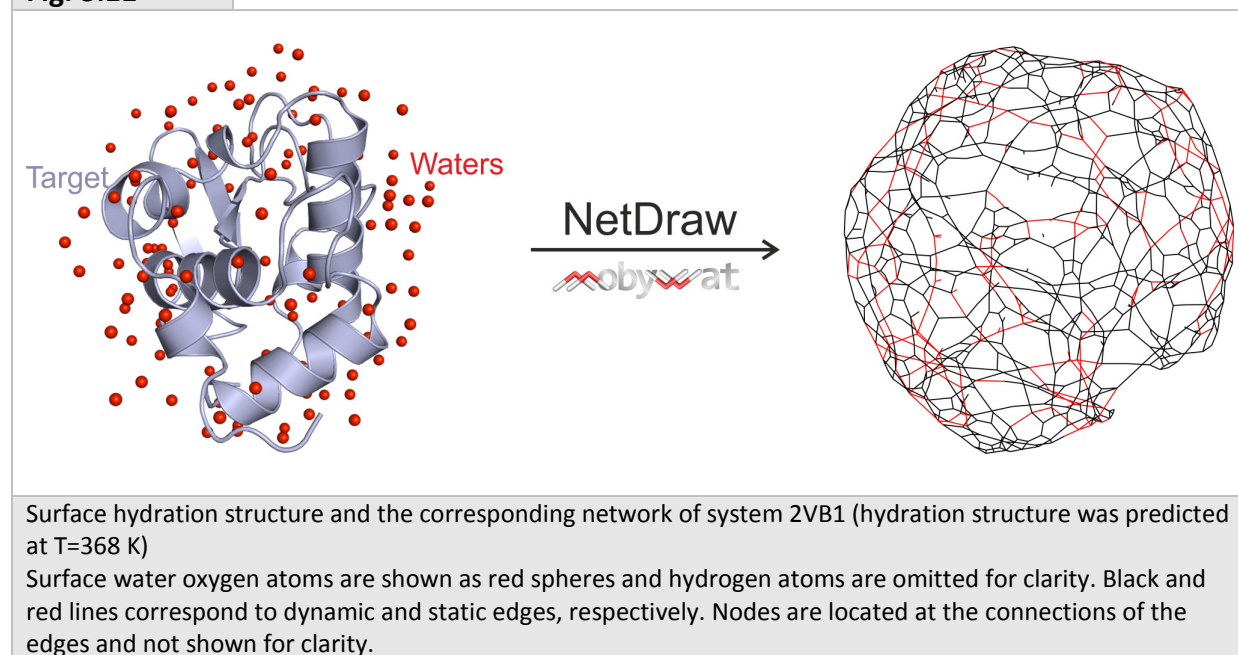
A sample command for a surface hydration network (Fig. 3.11):

```
$ mobywat -f system.pdb -t [A] -w Auto -m NetDraw -dmax 3 -mmax 50
```

A sample command for an interface hydration network (Fig. 3.10):

```
$ mobywat -f system.pdb -l [BC] -t [A] -w Auto -m NetDraw -dmax 3 -mmax 50
```

**Fig. 3.11**



Test packages of the NetDraw mode can be downloaded from the MobyWat web site.

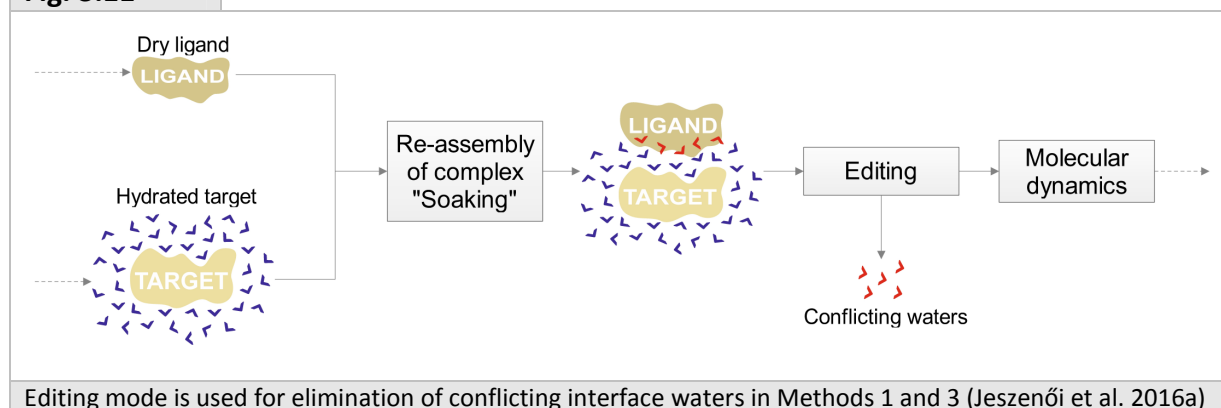


## 3.5 Editing mode

### 3.5.1 Overview

The Editing mode of MobyWat is used by Methods 1 and 3 (Jeszenői et al. 2016a) during the prediction of interface hydration structure. Editing mode works on the re-assembled solute (target+ligand) complex and water molecules predicted for the surface of the target molecule. Editing removes waters conflicting with the ligand structure (Fig. 3.11).

**Fig. 3.11**



### 3.5.2 Inputs

The Editing mode requires a single PDB file with water positions predicted for the target surface and the re-assembled solute complex including target and ligand.

### 3.5.3 Algorithm

- 1) Interface water molecules satisfying Eq. 2.2 are selected. For this, closest distances of water molecule OW atoms with ligand and target heavy atoms are calculated and distances  $\leq \mathbf{dmax}$  are listed. A common part of the two distance lists are produced.
- 2) Water molecules with closest distances  $\leq \mathbf{dmin}$  measured from ligand (conflicting waters) are eliminated from the final list of selected waters.

### 3.5.4 Outcomes

The list of selected water molecules and the edited PDB file with selected water positions are printed. Distance, log and original PDB files are also generated (Table 3.6).

### 3.5.5 Usage

In the following sample command, the target molecule is in chain **A**, whereas ligand is in chain **B**. The usage of text **-m Editing** is obligatory to initialize the present Editing mode of MobyWat (the default mode is Prediction).

```
$ mobywat -f system.pdb -l [B] -t [A] -w Auto -m Editing -dmax 5 -dmin 1.75
```

**Table 3.6**

EDITING MODE		
INPUT FILES & DEFAULTS		
-f	system.pdb	Single PDB file
INPUT PARAMETERS & DEFAULTS		
-dmax	3.500	Maximal distance limit, Å
-dmin	1.750	Minimal distance limit, Å
-m	Prediction	Program mode, Analysis/Prediction/NetDraw/Editing
INPUT RANGES		
-l	x-y/[xy...]	Ligand range, atom serial numbers/chain IDs
-t	x-y/[xy...]	Target range, atom serial numbers/chain IDs
-w	x-y/WAT/Auto	Waters range, atom serial numbers/residue name/automatic
OUTPUT FILES		
	O_system.log	Log file
	O_system_water_target.dst	Distance file
	O_system_water_ligand.dst	Distance file
	O_system_comm.res	List of interface water molecules selected for the edited PDB file
	O_system_dtd.pdb	Edited PDB file
	O_system_org.pdb	Original (input) PDB file

Test package of the Editing mode can be downloaded from the MobyWat web site.

## 4 File types, ranges, parameters

### 4.1 Input files

The present version of MobyWat accepts four types of structural input files. File types are automatically recognized by extensions of file names.

#### 4.1.1. Binary trajectory file (\*.xtc)

The xtc format is a portable binary format which can store trajectories in an efficient, compressed form. It uses the xdr routines for writing and reading data. Xtc files can be produced by MD package GROMACS. Note that xtc files contain only spatial coordinates x,y,z of all atoms of all frames of the trajectory. Similarly to other programs reading binary trajectories, MobyWat also requires the definition of atom types in a separate PDB-type topology file. The topology file must contain the same number of atoms in the same order as one frame. Production of such a topology file can be easily done along with the production of the xtc file in GROMACS (see Section 4.2 for details).

#### 4.1.2 Separate PDB files (\*.pdb)

For compatibility with any MD packages, MobyWat can also read a trajectory as a series of individual frames stored in standard PDB files. Obviously, the frames must have the same number of atoms in the same order. The files must have a sequential naming such as **system\_i.pdb**, where the use of **\_i.pdb** part of the file names is mandatory (i=x, x+1,x+2,...,y in MobyWat command line switch **-n x-y**, see also Section 4.3). Note, that topology file is not used for PDB files as they have all necessary information for MobyWat. For additional information on PDB files, please visit

<http://www.wwpdb.org/documentation/file-format>

#### 4.1.3 NMR-type PDB files (\*.pdb)

Instead of a series of PDB files a trajectory can be also stored in a single PDB file and frames can be handled as NMR models within the same file. MobyWat automatically recognizes NMR-type PDB files if the file name contains the token mdl in a form such as **system\_mdl.pdb**. Note, that in NMR-type PDB files the models (=frames) are defined using **MODEL** and **ENDMDL** tokens and model serial numbers specified after **MODEL** must be within the x-y range of switch **-n x-y** of MobyWat command line (see also Section 4.3). For additional information on NMR-type PDB files, please visit

<http://www.wwpdb.org/documentation/file-format-content/format33/sect9.html#MODEL>

It is important to note that storage of trajectories in PDB files may require much disk space, and therefore, xtc binary files can be recommended for routine use. However, MobyWat can also read PDB inputs described above to ensure compatibility with any MD software packages. When reading PDB input files MobyWat automatically converts the trajectories into xtc format and also creates a topology file used for subsequent calculations.

#### 4.1.4 Binary pool waters file (\*.plw)

MobyWat stores pool waters (Section 2) of the trajectory in a non-portable binary file which is automatically created as a first step of processing input files described in Sections 4.1.1-3. The file contains pool waters only from frames specified by the x-y range of switch **-n x-y** of MobyWat command line (see also Section 4.3). MobyWat also produces a pool information file (\*.pli) along with the plw file for future use. The pli file contains information such as dmax specifying the pool. As creation of water pools from the trajectory file may take considerable time for large systems, pool waters file can be useful if re-running of a prediction/analysis is required by the user e.g. for experimenting with various clustering schemes, tolerances, etc. Importantly, dmax cannot be changed in such repeated runs as it defines the pool, itself (Section 2). For information on the structure of pool waters file, please refer to Section 5.

## 4.2 Output files

MobyWat produces binary files and text files and all of them are marked with **O\_** in the beginning of the file name. Files xtc, plw, and pdb were described in Section 4.1. In other text files such as lst and txt additional information on e.g. clustering, RMSD, number of interface waters per frame, etc. are printed. MobyWat also creates a log file with details on input/output information.

## 4.3 Input ranges

### 4.3.1 Frame range

The user can specify the minimal (x) and maximal (y) serial numbers of frames with the use of switch **-n x-y** at MobyWat command line. Please, be sure that the frames exist in your trajectory input files (Section 4.1).

### 4.3.2 Molecular ranges

MobyWat requires definition of target and waters in the frames using **-t** and **-w** switches of the command line, respectively. There are three possibilities of definition.

- 1) Ranges of atom serial numbers can be given using minimal (x) and maximal (y) serial number in a form of **-t x-y** or **-w x-y**.
- 2) In the case of target, a list of one-letter chain IDs such as xy... can be also used in a form of **-t [xy...]**. Note, that use of [...] brackets is obligatory. This option is especially useful if atom list of target is non-continuous in the frame, and the target is stored in several chains.
- 3) In the case of waters, definition is also possible by specification of a single residue name such as **WAT** in the form **-w WAT**. MobyWat can automatically detect and assign water molecules in the frame using the **-w Auto** command line entry, as well. During automatic detection residue names **SOL**, **WAT** and **H2O** can be identified.

Besides their use for frames in the command line, the above specifications of molecular ranges can be used for the reference structure in Analysis mode and Validation sub-mode.

## 4.4 Input parameters

Required formats and default values of numerical and alphabetical parameters are listed e.g. in the quick help which can be printed to the screen by typing "mobywat" at the command prompt (Fig. 4.1).

**Fig. 4.1**

```
MobyWat
Calculation of hydration structures of molecular surfaces and interfaces

=====Ver=1.1=29=08=2015=====

Usage
$ mobywat -x <value_x> -y <value_y> ...

Input files
-f      system.xtc      Trajectory file with frames
-pli    system.pli      Pool information file
-r      system_ref.pdb  Reference file
-tpy    system_tpy.pdb  Topology file

Input ranges
-l      x-y/[xy...]     Ligand range, atom serial numbers/chain IDs
-t      x-y/[xy...]     Target range, atom serial numbers/chain IDs
-w      x-y/WAT/Auto    Waters range, atom serial numbers/residue name/automatic

Parameters & defaults
-bmax   30.000          B-factor limit, A^2
-cls    IDa             Clustering algorithm, IDa/IDe/POS/MER
-ctol   1.000           Clustering tolerance, A
-dmax   3.500           Max. distance limit, A
-dmin   1.750           Min. distance limit, A
-m      Prediction      Program mode, Analysis/Prediction/Editing/NetDraw
-mmax   50.000          Mobility tolerance
-mtol   1.500           Match tolerance, A
-n       0-10           Frame range, x-y
-ptol   2.500           Prediction tolerance, A
-ptop   50.000          Top cut of short prediction list, %
-v      Silent          Verbosity, Silent/Verbose/Diagnostic

For detailed instructions, please visit the web site at http://www.mobywat.com !

=====by=C=Hetenyi=====
```

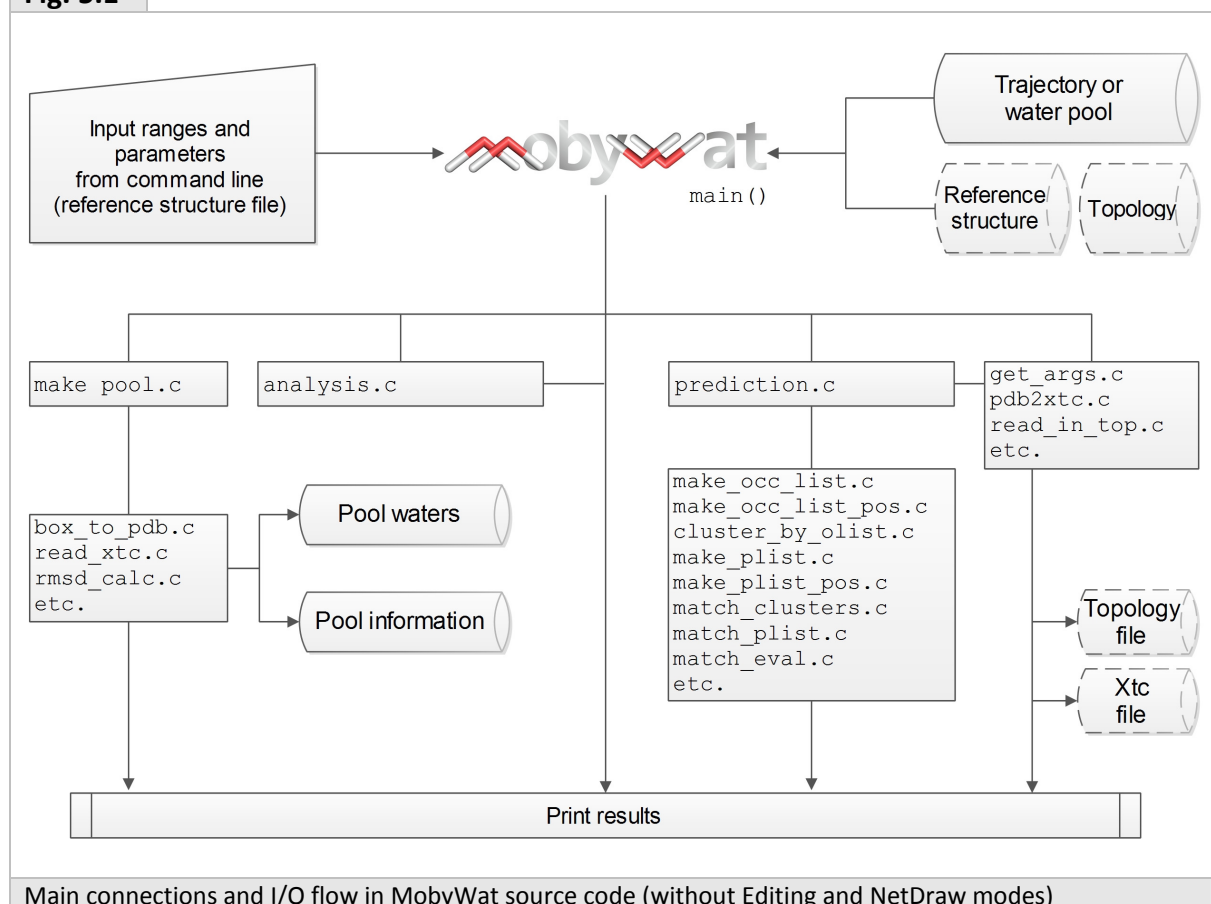
**Quick help**

## 5 Program details

### 5.1 Organization of code

MobyWat was written in standard C, and therefore, the source code shares known benefits of the language such as portability and efficient dynamic memory allocation. Thus, introduction of theoretical limit on system size was not necessary, which is beneficial for atomic systems of large target molecules of various sizes and up to tens of thousands of water molecules surrounding them in the simulation box. A schematic organization chart of the program is given in Fig. 5.1 as an overview of the main connections and I/O data flow in the code.

**Fig. 5.1**



### 5.2 Water pool data type

MobyWat uses a pool list with spatial coordinates, atom- and residue serial numbers of oxygen atoms for identification of a water molecule in the pool (Fig. 5.2). The binary pool waters (plw, Section 4.1.4) file stores pool lists of all selected waters and all frames specified at command line. Besides the benefits of shorter repeated runs, creation of plw files allows efficient use of disk space and memory.

**Fig. 5.2**

```
struct pool_list {  
    float x_coord;  
    float y_coord;  
    float z_coord;  
    int aser;           // atom serial no  
    int rser;           // res serial no  
} ;  
  
typedef struct pool_list pl;
```

Water pool data type

### 5.3 Binary trajectory files

At present, MobyWat handles xtc files. GROMACS xtc files are based on extended xdr libraries written using an algorithm where precision can be set by scale factor, typically 1000 used for multiplication of the coordinates in nm and the values are rounded as integers. The modified xdr libraries were adopted by MobyWat for reading and writing xtc files, and the precision was increased to 10000 to handle PDB coordinates in Å from any source. The modified xdr libraries and their documentation can be found at the following link.

[http://www.gromacs.org/Developer\\_Zone/Programming\\_Guide/XTC\\_Library](http://www.gromacs.org/Developer_Zone/Programming_Guide/XTC_Library)

## 6 Installation and tests

MobyWat is available free of charge for the scientific community. On the web page of MobyWat we provide a precompiled executable and test packages with detailed instructions which can be used without any further installation steps.

The User is encouraged to compile the program for his/her own machine. MobyWat is an open source program written in C, and therefore, it can be easily compiled and installed on practically any platforms.

For a standard GNU environment, some simple steps can be followed and the program is ready for use. First download the **mobywat.tgz** file from the program's web site into a directory called e.g. **\$HOME/download** and type the following text at the command prompt.

```
$ cd $HOME/download
$ tar -xvf mobywat.tgz
$ cd mobywat/src
$ make
```

Now, you have the executable file named **mobywat** in **mobywat/src**. You can also use command **make install** to copy the executable file into **mobywat/bin** and **\$HOME/bin**. Binary and object files can be removed from **mobywat/src** by typing **make clean**.

```
$ make install
$ make clean
```

## 7 Version history

### Version 1.0

- I/O and other core functions
- Prediction mode
- SR calculation in analysis mode (further tests and improvements of analysis mode will follow)

### Version 1.1

- NetDraw mode
- Editing mode

## 8 How to cite?

Jeszenői N, Horváth I, Bálint M, van der Spoel D, Hetényi C. (2015)  
Mobility-based prediction of hydration structures of protein surfaces.  
*Bioinformatics*, **31**, 1959-65.

Jeszenői N, Bálint M, Horváth I, van der Spoel D, Hetényi C. (2016)  
Exploration of interfacial hydration networks of target–ligand complexes.  
*J Chem Inf Model*, **56**, 148-58.



## 9 Applications

Trajectory files including all mobility information necessary for MobyWat analysis or prediction can be produced by any MD program package. Here, a short description is given about using MD program package GROMACS ver. 5.0 in conjunction with MobyWat. For specific information on the contents on input files, and procedures, please refer to GROMACS web site.

<http://www.gromacs.org>

All example files specified in the following sections can be downloaded from the web page of MobyWat.

### 9.1 Surface hydration

In the following sections, steps of preparation of trajectory of a target molecule called **system.pdb**.

#### 9.1.1. Running MD calculations on a target protein

##### 9.1.1.1 Preparation of a simulation box

As a first step target molecule is placed in a box and the box is filled up with water molecules. Coordinates of the box are stored in a file named **b4em.gro**. **Tip3p** water model, **Amber99sb-ildn** force field and a **dodecahedral** box with 10 Å (=1 nm) spacing were specified.

```
gmx pdb2gmx -water tip3p -ff amber99sb-ildn -ignh -f system.pdb
gmx editconf -o -d 1 -bt dodecahedron -f conf.gro
gmx solvate -cp out -cs -o b4em -p topol
```

If it is necessary, neutrality of the system can be achieved by adding **X** copies of positive (**Na<sup>+</sup>**) or negative (**Cl<sup>-</sup>**) ions to the box.

```
gmx grompp -v -f steep -c b4em -o em -p topol
gmx genion -s em.tpr -o ion_b4em -p topol -pname NA -np X
gmx genion -s em.tpr -o ion_b4em -p topol -nname CL -nn X
```

##### 9.1.1.2 Energy-minimization of the system

Before launching productive MD calculations it is advisable to energy minimize the content of the box. Here, commands of a two step minimization are shown including steepest descent and a conjugated gradient runs. Minimizations are performed by the **gmx\_d mdrun** (double precision executable) program and the binary inputs are produced by **grompp**. Note, that **-c ion\_b4em** can be specified instead of **-c b4em** if you have added neutralizing ions to your box (Section 9.1). It is important to put position restraints on all solute heavy atoms with 10<sup>3</sup> kJmol<sup>-1</sup>nm<sup>-2</sup> in both minimization steps. Position restraints must be defined in both mdp file ("define = -DPOSRES"), and the position restraints topology file ("posre.itp" generated at the first step) must be present in the working directory. If your system contains

structural ions, applying distance restraints between the ion and the coordinating residues would be useful to maintain proper geometry.

```
gmx grompp -v -f steep -c b4em -o st -p topol.top
gmx mdrun_d -v -s st -o st -c after_st -g st
gmx grompp -v -f cg -c after_st -o cg -p topol.top
gmx mdrun_d -v -s cg -o cg -c after_cg -g cg
```

### 9.1.1.3 Producing trajectory file

Final inputs are produced from the energy minimized system **after\_cg** and MD calculations can be launched using **mdrun**. Trajectory of the system is stored in an **md.trr** file specified at switch **-o**. It is important to put position restraints on all solute heavy atoms with  $10^3 \text{ kJmol}^{-1} \text{nm}^{-2}$  during mdrun. Position restraints must be defined in both mdp file ("define = -DPOSRES"), and the position restraints topology file ("posre.itp" generated at the first step) must be present in the working directory. If your system contains structural ions, applying distance restraints between the ion and the coordinating residues would be useful to maintain proper geometry.

```
gmx grompp -f md -o md -c after_cg -r after_cg -p topol.top -maxwarn 1
gmx mdrun -v -s md -e md -o md -c after_md -g md.log
```

A simulation example package can be downloaded from the MobyWat web site.

## 9.1.2 Preparation of the trajectory for MobyWat

### 9.1.2.1 Preparation of the trajectory for prediction

Once you have your trajectory in an **md.trr** file fast conversions are recommended using **trjconv**. Such conversions handle periodic boundary effects, center the system in the box and fit target molecules in subsequent frames on the top of the first frame (Section 3.2.2).

```
gmx trjconv -f md.trr -s md.tpr -o pbc_1.xtc -pbc whole <<EOF
0
EOF
```

```
gmx trjconv -f pbc_1.xtc -s md.tpr -o pbc_2.xtc -pbc cluster <<EOF
1
0
EOF
```

```
gmx trjconv -f pbc_2.xtc -s md.tpr -o pbc_3.xtc -center -pbc mol -ur
compact <<EOF
1
0
EOF
```

```
gmx trjconv -f pbc_3.xtc -s md.tpr -o system.xtc -fit progressive <<EOF
3
0
EOF
```

In the last command line, instead of **-o system.xtc**, a switch **-o system\_md1.pdb** or **-sep -o system\_pdb** can be specified for an NMR-type PDB file or separate PDB files (Section 4.1), respectively.

A topology file `system_tpy.pdb` can be easily produced for MobyWat by `trjconv`.

```
gmx trjconv -f pbc_3.xtc -s md.tpr -o system_tpy.pdb -b 0 -e 0 -fit
progressive <<EOF
3
0
EOF
```

The `system.xtc` (or the corresponding `pdb` files) and `system_tpy.pdb` can be used as input for MobyWat.

#### 9.1.2.2 Preparation of the trajectory for analysis or validation

If you want to compare the trajectory in `md.trr` to a reference crystallographic structure (Sections 3.1 and 3.3) named as e.g. `system_ref.pdb`, you will need to fit your trajectory onto the initial structure with waters using `confrms` (Section 3.1.2). The following command specifies multi-frame fit using  $C_{\alpha}$  atoms of the protein backbone.

```
gmx confrms -one -f1 system_ref.pdb -f2 md.tpr -o fit.pdb <<EOF
3
3
EOF
```

A chain ID “A” can be added using `editconf`.

```
gmx editconf -label A -f fit.pdb -o fit.pdb
```

```
gmx trjconv -f md.trr -s md.tpr -o pbc_1.xtc -pbc whole <<EOF
0
EOF
```

```
gmx trjconv -f pbc_1.xtc -s md.tpr -o pbc_2.xtc -pbc cluster <<EOF
1
0
EOF
```

```
gmx trjconv -f pbc_2.xtc -s md.tpr -o pbc_3.xtc -center -pbc mol -ur
compact <<EOF
1
0
EOF
```

```
gmx trjconv -f pbc_3.xtc -s fit.pdb -o system.xtc -fit progressive <<EOF
3
0
EOF
```

Preparation of topology file is also similar to Section 9.2.1.

```
gmx trjconv -f pbc_3.xtc -s fit.pdb -o system_tpy.pdb -b 0 -e 0 -fit
progressive <<EOF
3
0
EOF
```

A simulation example package can be downloaded from the MobyWat web site.

## 9.2 Interface hydration

Methods of interface hydration are described in Figs. S7-S9 of the Supporting Information of our paper (Jeszenői et al. 2016a). The Supporting Information is freely available at

[http://pubs.acs.org/doi/suppl/10.1021/acs.jcim.5b00638/suppl\\_file/ci5b00638\\_si\\_001.pdf](http://pubs.acs.org/doi/suppl/10.1021/acs.jcim.5b00638/suppl_file/ci5b00638_si_001.pdf)

Method 3 is recommended for void-free hydration.

Application packages for interface methods can be downloaded from the MobyWat web site.

## 10 References

- Afonine, P.V., Grosse-Kunstleve, R.W., Adams, P.D. and Urzhumstev, A. (2013) Bulk-solvent and overall scaling revisited: faster calculations, improved results. *Acta Cryst.*, **D69**, 625-634.
- Badger, J. (1997) Modeling and refinement of water molecules and disordered solvent. *Meth. Enzymol.*, **277**, 344-352.
- Baron, R., Setny, P. and McCammon, J.A. (2012) Hydrophobic association and volume-confined water molecules. In: Protein-ligand interactions, Ed. by Gohlke, H. Wiley-VCH Verlag GmbH & Co. KGaA, Weinheim.
- Berman, H.M., Henrick, K. and Nakamura, H. (2003) Announcing the worldwide Protein Data Bank *Nat. Struct. Biol.*, **10** (12), 98.
- Carugo, O. and Bordo, D. (1999) How many water molecules can be detected by protein crystallography? *Acta Cryst.*, **D55**, 479-483.
- Dror, R.O., Dirks, R.M., Grossman, J.P., Xu, H. and Shaw, D.E. (2012) Biomolecular simulation: A computational microscope for molecular biology. *Annu. Rev. Biophys.*, **41**, 429-52.
- Finney, J.L. (1977) The organization and function of water in protein crystals. *Phil. Trans. R. Soc. Lond. B*, **278**, 3-32.
- Halle, B. (2004a) Protein hydration dynamics in solution: a critical survey. *Phil. Trans. R. Soc. Lond. B*, **359**, 1207-1224.
- Halle, B. (2004b) Biomolecular cryocrystallography: Structural changes during flash-cooling. *Proc. Natl. Acad. Sci. USA*, **101**, 4793-4798.
- Henchman, R.H. and McCammon, J.A. (2002) Extracting hydration sites around proteins from explicit water simulations. *J. Comput. Chem.* **23**, 861-869.
- Huang, H.-C., Jupiter, D., Qiu, M., Briggs, J.M. and VanBuren, V. (2008) Cluster analysis of hydration waters around the active sites of bacterial alanine racemase using a 2-ns MD simulation. *Biopolymers*, **89**, 210-219.
- Islam, S.A. and Weaver, D.L. (1990) Molecular interactions in protein crystals: solvent accessible surface and stability. *Proteins*, **8**, 1-5.
- Israelachvili, J. and Wennerström, H. (1996) Role of hydration and water structure in biological and colloidal interactions. *Nature*, **379**, 219-225.
- Jeszenői N, Horváth I, Bálint M, van der Spoel D, Hetényi C. (2015) Mobility-based prediction of hydration structures of protein surfaces. *Bioinformatics*, **31**, 1959-65.
- Jeszenői N, Bálint M, Horváth I, van der Spoel D, Hetényi C. (2016a) Exploration of interfacial hydration networks of target-ligand complexes. *J Chem Inf Model*, **56**, 148-58.
- Jeszenői N, Bálint M, Horváth I, Hetényi C. (2016b) *Manuscript*.
- Ladbury, J.E. (1996) Just add water! The effect of water on the specificity of protein-ligand binding sites and its potential application to drug design. *Chem. Biol.*, **3**, 973-980.
- Lounnas, V., Pettitt, B.M. and Phillips, Jr., G.N. (1994) A global model of the protein-solvent interface. *Biophys. J.*, **66**, 601-614.
- Madhusudhan, M.S. and Vishveshwara, S. (2001) Deducing hydration sites of a protein from molecular dynamics simulations. *J. Biomol. Struct. Dynam.*, **19**, 105-114.
- Makarov, V.A., Andrews, B.K., Pettitt, B.M. (1998) Reconstructing the protein-water interface. *Biopolymers*, **45**, 469-478.
- Michel, J., Tirado-Rives, J. and Jorgensen, W.L. (2009) Prediction of the water content in protein binding sites. *J Phys Chem B*, **113**, 13337-13346.

- Petsko,G.A. and Ringe,D. (2009) Protein structure and function. Oxford University Press Inc., New York.
- Pettitt,B.M. and Karplus,M. (1987) The structure of water surrounding a peptide: a theoretical approach. *Chem. Phys. Lett.*, **136**, 383-386.
- Pitt,W.R. and Goodfellow,J.M. (1991) Modelling of solvent positions around polar groups in proteins. *Protein Engng.*, **4**, 531-537.
- Ross,G.A., Morris,G.M. and Biggin,P.C. (2012) Rapid and accurate prediction and scoring of water molecules in protein binding sites. *PLoS ONE*, **7**(3), e32036.
- Rossky,P.J. and Karplus,M. (1979) Solvation. A molecular dynamics study of a dipeptide in water. *J. Am. Chem. Soc.*, **101**, 1913-1937.
- Savage, H. and Wlodawer, A. (1986) Determination of water structure around biomolecules using x-ray and neutron diffraction methods. *Meth. Enzymol.*, **127**, 162-183.
- Schoenborn,B.P., Garcia,A. and Knott,R. (1995) Hydration in protein crystallography. *Prog. Biophys. Molec. Biol.*, **64**, 105-119.
- Schymkowitz,J.W.H., Rousseau,F., Martins,I.C., Ferkinghoff-Borg,J., Stricher,F. and Serrano,L. (2005) Prediction of water and metal binding sites and their affinities by using the Fold-X force field. *Proc. Natl. Acad. Sci. USA*, **102**, 10147–10152.
- van Gunsteren,W.F., Berendsen,H.J.C., Hermans,J., Hol,W.G. and Postma,J.P.M. (1983) Computer simulation of the dynamics of hydrated protein crystals and its comparison with x-ray data. *Proc. Natl. Acad. Sci. USA*, **80**, 4315-4319.
- Vedani,A. and Huhta,D.W. (1991) An algorithm for the systematic solvation of proteins based on the directionality of hydrogen bonds. *J. Am. Chem. Soc.*, **113**, 5860-5862.
- Virtanen,J.J., Makowski,L., Sosnick,T.R. and Freed,K.F. (2010) Modeling the hydration layer around proteins: HyPred. *Biophys. J.*, **99**, 1611–1619.

