Supporting information to:

“PTRAJ and CPPTRAJ: Software for the processing and analysis of molecular dynamics trajectory data” by DR Roe and TE Cheatham, III.

Submitted to Journal of Chemical Theory and Computation, April 2013.

Included:

1. Summary of RDPARM functionality and output.
2. Summary of PTRAJ/CPPTRAJ commands
3. PTRAJ/CPPTRAJ atom mask selection syntax
4. PTRAJ/CPPTRAJ Matrix/Vector Analysis scripts
5. Example CPPTRAJ scripts/commands
6. Example PTRAJ scripts
1) Summary of RDPARM Functionality and Output

RDPARM, available in the AmberTools suite of programs at [http://ambermd.org](http://ambermd.org), is software that provides a means to look at the information that is stored in Amber parameter/topology files. An example of its usage is below. From the UNIX prompt, one specifies the executable name and an Amber parameter/topology file. A summary of the information is listed for the 18-mer DNA duplex topology read in.

% rdparm GAAC-nowater.topo

\/-
/- Welcome to RDPARM (an interactive utility for reading AMBER topology files)
/- Type "help" or "?" for more information
\/- Version: "AMBER 12.0 integrated" (4/2012)
\/-

Residue labels:

DG5  DC  DA  DC  DG  DA  DA  DC  DA  DG  DA
DA  DC  DG  DA  DA  DC  DC3  DG5  DC
DG  DT  DT  DC  DT  DT  DT  DC  DG  DT
DT  DC  DT  DG  DT  DG  DT  DG  DT  DG
K+  K+  K+  K+  K+  K+  K+  K+  K+  K+
... K+  Cl-  K+  Cl-  K+  Cl-  Cl-  K+  Cl-

RDPARM MENU. Please enter commands. Use "?" or "help"
for more info. "exit" or "quit" to leave program...

A variety of commands are supported as shown by specifying "help".

RDPARM MENU: help

The following commands are currently available:

atoms, printAtoms
bonds, printBonds
angles, printAngles
dihedrals, printDihedrals
pertbonds, perturbedBonds
pertangles, perturbedAngles
pertdihedrals, perturbedDihedrals
printExluded
printLennardJones
printTypes
parmInfo
checkcoords
delete  <bond || angle || dihedral> <number>
delperturbed <bond || angle || dihedral> <number>
restrain <bond || angle || dihedral>
openparm  <filename>
writeparm  <filename>
analyze  <AMBER trajectory || AMBER coordinates>
rms  <AMBER trajectory>
stripwater
transform  <AMBER trajectory>
translateBox  <AMBER coords>
modifyBoxInfo
modifyMolInfo
quit, exit
Using RDPARM, a user can display information about atoms, bonds, Lennard-Jones parameters, etc. Atoms are selected via a “mask” string which is equivalent to the syntax used by Midas/Chimera and recently within Amber with “:” referring to residues, and “@” to atom names or numbers.

**RDPARM MENU: atoms @C5**

Mask [:@C5'] represents 1 atoms

Number:  Atom  Charge  Mass  ( Residue )  Type  Tree

3:  C5'  -0.00690  12.0 ( 1:DG5 )  CI  M

**RDPARM MENU: bonds @C5**

Mask [:@C5'] represents 1 atoms

<table>
<thead>
<tr>
<th>Bond</th>
<th>Kb</th>
<th>Req</th>
<th>atom names   (numbers)</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td>340.00</td>
<td>1.090</td>
<td>:1@C5' :1@H5'1 (3,4)</td>
</tr>
<tr>
<td>11</td>
<td>340.00</td>
<td>1.090</td>
<td>:1@C5' :1@H5'2 (3,5)</td>
</tr>
<tr>
<td>428</td>
<td>310.00</td>
<td>1.526</td>
<td>:1@C5' :1@C4' (3,6)</td>
</tr>
<tr>
<td>429</td>
<td>320.00</td>
<td>1.410</td>
<td>:1@O5' :1@C5' (2,3)</td>
</tr>
</tbody>
</table>

**RDPARM MENU: dihed @5**

Mask [@5] represents 1 atoms

<table>
<thead>
<tr>
<th>Dihedral</th>
<th>pk</th>
<th>phase</th>
<th>pn</th>
<th>atoms</th>
</tr>
</thead>
<tbody>
<tr>
<td>41</td>
<td>0.156</td>
<td>0.00</td>
<td>3.0</td>
<td>:1@H5'2 :1@C5' :1@C4' :1@H4' (5,3,6,7)</td>
</tr>
<tr>
<td>42</td>
<td>0.250</td>
<td>0.00</td>
<td>1.0</td>
<td>:1@H5'2 :1@C5' :1@C4' :1@O4' (5,3,6,8)</td>
</tr>
<tr>
<td>43</td>
<td>0.156</td>
<td>0.00</td>
<td>3.0</td>
<td>:1@H5'2 :1@C5' :1@C4' :1@C3' (5,3,6,26)</td>
</tr>
<tr>
<td>50</td>
<td>0.167</td>
<td>0.00</td>
<td>3.0</td>
<td>:1@H5' :1@O5' :1@C5' :1@H5'2 (1,2,3,5)</td>
</tr>
</tbody>
</table>

**RDPARM MENU: printtypes**

NOTE: if either A or C is zero (*), we cannot infer the value of r* or epsilon and assume zero...

<table>
<thead>
<tr>
<th>Type</th>
<th>r*</th>
<th>eps</th>
</tr>
</thead>
<tbody>
<tr>
<td>* HO</td>
<td>0.0000</td>
<td>0.0000</td>
</tr>
<tr>
<td>OH</td>
<td>1.7210</td>
<td>0.2104</td>
</tr>
<tr>
<td>CT</td>
<td>1.9080</td>
<td>0.1094</td>
</tr>
<tr>
<td>H1</td>
<td>1.3870</td>
<td>0.0157</td>
</tr>
<tr>
<td>OS</td>
<td>1.6837</td>
<td>0.1700</td>
</tr>
<tr>
<td>H2</td>
<td>1.2870</td>
<td>0.0157</td>
</tr>
<tr>
<td>NC</td>
<td>1.8240</td>
<td>0.1700</td>
</tr>
<tr>
<td>C</td>
<td>1.9080</td>
<td>0.0860</td>
</tr>
<tr>
<td>H5</td>
<td>1.3590</td>
<td>0.0150</td>
</tr>
<tr>
<td>O</td>
<td>1.6612</td>
<td>0.2100</td>
</tr>
<tr>
<td>H</td>
<td>0.6000</td>
<td>0.0157</td>
</tr>
<tr>
<td>HC</td>
<td>1.4870</td>
<td>0.0157</td>
</tr>
<tr>
<td>P</td>
<td>2.1000</td>
<td>0.2000</td>
</tr>
<tr>
<td>H4</td>
<td>1.4090</td>
<td>0.0150</td>
</tr>
<tr>
<td>HA</td>
<td>1.4590</td>
<td>0.0150</td>
</tr>
<tr>
<td>K+</td>
<td>1.8687</td>
<td>0.1000</td>
</tr>
<tr>
<td>C1-</td>
<td>2.4700</td>
<td>0.1000</td>
</tr>
</tbody>
</table>
## 2) Summary of PTRAJ/CPPTRAJ Commands

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
<th>Supported By</th>
</tr>
</thead>
<tbody>
<tr>
<td>2drms</td>
<td>Calculate two-dimensional RMSD (i.e. RMSD of each frame to every other frame).</td>
<td>Both</td>
</tr>
<tr>
<td>analyze matrix</td>
<td>Diagonalize a previously calculated symmetric matrix to obtain eigenmodes.</td>
<td>Both</td>
</tr>
<tr>
<td>analyze modes</td>
<td>Analyze previously calculated eigenmodes.</td>
<td>Both</td>
</tr>
<tr>
<td>analyze timecorr</td>
<td>Calculate time cross or auto correlation functions for previously calculated vectors.</td>
<td>Both</td>
</tr>
<tr>
<td>angle</td>
<td>Compute the angle in degrees between the center of mass of the three atom specifications listed.</td>
<td>Both</td>
</tr>
<tr>
<td>atomicfluct</td>
<td>Compute the atomic positional fluctuations (RMSF) or B-factors: Note this does not implicitly perform a RMS fit prior.</td>
<td>Both</td>
</tr>
<tr>
<td>atommap</td>
<td>Attempt to map atoms between two reference structures based on overall structure and bonding.</td>
<td>CPPTRAJ</td>
</tr>
<tr>
<td>autoimage</td>
<td>Automatically center and image a trajectory with periodic boundary conditions.</td>
<td>CPPTRAJ</td>
</tr>
<tr>
<td>average</td>
<td>Compute the straight coordinate average of the coordinate sets read in.</td>
<td>Both</td>
</tr>
<tr>
<td>center</td>
<td>Move all the coordinates to the origin or box center.</td>
<td>Both</td>
</tr>
<tr>
<td>overlap</td>
<td>Look for atoms that are closer (and optionally further separated than) a specified distance.</td>
<td>PTRAJ</td>
</tr>
<tr>
<td>checkstruct</td>
<td>Check for close atomic overlaps and/or unusual bond lengths in structures.</td>
<td>CPPTRAJ</td>
</tr>
<tr>
<td>closest</td>
<td>Strip the trajectories such that only the closest N (chosen by the user) solvent molecules are retained.</td>
<td>Both</td>
</tr>
<tr>
<td>cluster</td>
<td>Group configurations from the MD trajectory into distinct sets.</td>
<td>Both†</td>
</tr>
<tr>
<td>clusterdihedral</td>
<td>Group configurations from the MD trajectory by binning dihedral angles; output can be used with RREMD in SANDER.</td>
<td>Both</td>
</tr>
<tr>
<td>contacts</td>
<td>Find the number of atom interactions (and native contacts) within a given distance.</td>
<td>Both</td>
</tr>
<tr>
<td>corr</td>
<td>Calculate correlation coefficient between data sets.</td>
<td>CPPTRAJ</td>
</tr>
<tr>
<td>diffusion</td>
<td>Calculate the mean-squared displacements vs. frame number.</td>
<td>Both</td>
</tr>
<tr>
<td>dihedral</td>
<td>Calculate the dihedral values from the center of mass of the four specified atom masks.</td>
<td>Both</td>
</tr>
<tr>
<td>distance</td>
<td>Calculate the distance in Å between the center of mass of the two atom specifications listed.</td>
<td>Both</td>
</tr>
<tr>
<td>drms</td>
<td>Calculate distance RMSD (also known as DME).</td>
<td>CPPTRAJ</td>
</tr>
<tr>
<td>grid</td>
<td>Create a grid in X-Plor density format that has a count of the number of selected atoms in each grid cell.</td>
<td>Both</td>
</tr>
<tr>
<td>hbond</td>
<td>Keeps track of distances and angles between triplets of atoms.</td>
<td>Both*</td>
</tr>
<tr>
<td>histogram</td>
<td>Histogram previously calculated data in N-dimensions.</td>
<td>CPPTRAJ</td>
</tr>
<tr>
<td>image</td>
<td>With periodic boundary conditions, bring molecules outside the periodic box into the central unit cell.</td>
<td>Both</td>
</tr>
<tr>
<td>jcoupling</td>
<td>Calculate j-coupling values from dihedral angles.</td>
<td>CPPTRAJ</td>
</tr>
<tr>
<td>Command</td>
<td>Description</td>
<td>Tool(s)</td>
</tr>
<tr>
<td>----------</td>
<td>-----------------------------------------------------------------------------</td>
<td>----------</td>
</tr>
<tr>
<td>mask</td>
<td>Write information on selected atoms for each frame; works with distance-based masks.</td>
<td>CPPTRAJ</td>
</tr>
<tr>
<td>matrix</td>
<td>Calculate matrices of various types from input coordinates.</td>
<td>Both</td>
</tr>
<tr>
<td>molsurf</td>
<td>Calculate the Connolly surface area.</td>
<td>CPPTRAJ</td>
</tr>
<tr>
<td>nastruct</td>
<td>Calculate various nucleic-acid structure parameters.</td>
<td>CPPTRAJ</td>
</tr>
<tr>
<td>outtraj</td>
<td>Similar to 'trajout', except coordinates can be written during action processing instead of only at the end.</td>
<td>CPPTRAJ</td>
</tr>
<tr>
<td>principal</td>
<td>Align to the principal coordinate axis.</td>
<td>Both</td>
</tr>
<tr>
<td>projection</td>
<td>Project coordinates along previously calculated eigenmodes.</td>
<td>Both</td>
</tr>
<tr>
<td>pucker</td>
<td>Compute the pucker for 5-membered rings.</td>
<td>Both</td>
</tr>
<tr>
<td>radgyr</td>
<td>Determine the radius of gyration for each frame.</td>
<td>Both</td>
</tr>
<tr>
<td>radial</td>
<td>Compute a radial distribution function.</td>
<td>Both</td>
</tr>
<tr>
<td>randomizeions</td>
<td>Move single atom ion residues to new positions by swapping with random water molecules.</td>
<td>Both</td>
</tr>
<tr>
<td>rms</td>
<td>Compute the RMSd to the first frame or a reference frame.</td>
<td>Both</td>
</tr>
<tr>
<td>rmsavgcor</td>
<td>Calculate RMSD correlation by from the average RMSD of running-averaged coordinates over increasing window sizes.</td>
<td>CPPTRAJ</td>
</tr>
<tr>
<td>runningavg</td>
<td>Calculate the running average of coordinates over a given window size.</td>
<td>CPPTRAJ</td>
</tr>
<tr>
<td>secstruct</td>
<td>Determine the secondary structure of proteins.</td>
<td>Both</td>
</tr>
<tr>
<td>strip</td>
<td>Remove atoms from the trajectory.</td>
<td>Both</td>
</tr>
<tr>
<td>surf</td>
<td>Calculate the LCPO surface area.</td>
<td>CPPTRAJ</td>
</tr>
<tr>
<td>translate</td>
<td>Move the entire system in the x, y and/or z directions.</td>
<td>Both</td>
</tr>
<tr>
<td>trunctoct</td>
<td>Create an old style AMBER truncated octahedron.</td>
<td>PTRAJ</td>
</tr>
<tr>
<td>unstrip</td>
<td>Return to original topology/coordinates prior to action processing.</td>
<td>CPPTRAJ</td>
</tr>
<tr>
<td>unwrap</td>
<td>With periodic boundary, unwrap molecules to generate continuous trajectories.</td>
<td>Both</td>
</tr>
<tr>
<td>vector</td>
<td>Calculate vector quantities of various types from input coordinates.</td>
<td>Both</td>
</tr>
<tr>
<td>watershell</td>
<td>Determine how many solvent models are within a given radius of the solute.</td>
<td>Both</td>
</tr>
</tbody>
</table>

†The cluster command has many more options in PTRAJ.
*The hbond command functions very differently in PTRAJ and CPPTRAJ.
3) PTRAJ/CPPTRAJ atom mask selection syntax

Both PTRAJ and CPPTRAJ use the same mask syntax for atom selection (and this is similar to that present in UCSF's MidasPlus and Chimera visualization programs). Mask expressions are case-sensitive. Either names or numbers can be used. Masks can contain both ranges (denoted with '-' and comma separated lists. The logical operands ' & (and), '|' (or), and '!' (not) are also supported.

The syntax for elementary selections is the following:

{})residue numlist} e.g. [:1-10] [:1,3,5] [:1-3,5,7-9]
{}residue namelist} e.g. [:LYS] [:ARG,ALA,GLY]
@)atom numlist} e.g. [@012,17] [@054-85] [@012,54-85,90]
@)atom namelist} e.g. [@CA] [@CA,C,O,N,H]

Several wildcard characters are supported:

'*' -- zero or more characters.
'?.' -- one character.
'=' -- same as '*'

The wildcards can also be used with numbers or other mask characters, e.g. ':?0' means ":10,20,30,40,50,60,70,80,90", ':*" means all residues and '@*" means all atoms.

Compound expressions of the following type are allowed:

{})residue numlist | namelist}@)atom namelist | numlist}

and are processed as:

{})residue numlist | namelist} & @)atom namelist | numlist}

e.g. ':1-10@CA' is equivalent to ":1-10 & @CA".

More examples:

All residues named 'ALA' and 'TRP' (alanine and tryptophan):

:ALA,TRP

Atoms named 'CA' (alpha carbon) in residues 5 and 10:

:5,10@CA
All non-hydrogen atoms:

*:!*@H=

!*@H=

All protein backbone atoms (AMBER naming convention):

@CA,C,O,N,H

All protein non-backbone atoms (i.e. sidechain atoms):

!*CA,C,O,N,H

All backbone oxygens in residues 1-500 but not in water, lysine or arginine residues:

:1-500@O&!(:WAT|:LYS,ARG)

Both PTRAJ and CPPTRAJ support distance-based masks. For speed, distances are calculated only once based on a reference frame. Distance-based masks that update each frame are currently only supported by the mask action in CPPTRAJ.

The syntax for selection by distance is

{mask}[<|>][:|@]{distance}

Where {mask} is the mask of atoms to calculate distances to, ‘<’ means within, ‘>’ means without, ‘:’ means residues, ‘@’ means atoms, and {distance} is the cutoff. So for example,

:11-17<@2.4

means all atoms within 2.4 Å distance to residues 11-17. A practical use of this could be e.g. stripping all residues farther than 3.0 Å from residue 4 of a certain system using reference coordinates:

```
reference mol.rst7 # reference coordinates for distance-based mask
trajin mol.crd     # trajectory to be stripped
strip !(:4<:3.0)
```
4) PTRAJ/CPPTRAJ Matrix/Vector Analysis Scripts

1) Principal component analysis; calculate the first two principal components of a system using just C-alpha atoms:

#Script 1
trajin mdcrd
matrix @CA covar name MAT
analyze matrix MAT out modes.dat vecs 2

#Script 2
trajin mdcrd
projection modes modes.dat out PCA1-2.dat beg 1 end 2 @CA

In the first script, the CA atom coordinate covariance matrix is calculated from coordinates in trajectory 'mdcrd' and stored as 'MAT'. The matrix 'MAT' is then diagonalized and the first two eigenmodes are calculated and stored in the file 'modes.dat'. In the second script, the eigenmodes from 'modes.dat' are read back in and the CA atom coordinates from mdcrd are projected along them, with results written to PCA1-2.dat.

2) Calculating time correlation functions for vectors. Vectors between atoms 5 and 6 as well as 7 and 8 are calculated below, for which the auto time correlation and cross time correlation functions are obtained respectively.

vector v0 @5 corr @6 order 2
vector v1 @7 corr @8 order 2
analyze timecorr vec1 v0 tstep 1.0 tcorr 100.0 out v0.out
analyze timecorr vec1 v1 tstep 1.0 tcorr 100.0 out v1.out
analyze timecorr vec1 v0 vec2 v1 tstep 1.0 tcorr 100.0 out v0_v1.out

Similarly, a vector perpendicular to the plane through atoms 18, 19, and 20 is obtained and the auto time correlation function calculated.

vector v2 @18,@19,@20 corrplane order 2
analyze timecorr vec1 v3 tstep 1.0 tcorr 100.0 out v2.out
5) Example CPPTRAJ Scripts/Commands:

1) Select the last frame of a trajectory as a reference:

   reference solvated.nc lastframe

2) Select a range of frames from 1 to the last frame, skipping every 10:

   trajin solvated.nc 1 last 10

3) Use of outtraj to write coordinates during action processing:

   trajin solvated.nc
   strip :WAT
   outtraj nowat.crd
   rms first :2-11
   trajout nowat.rms.crd

   This will read frames from 'solvated.nc', remove residues named WAT (the solvent), write out a trajectory with no solvent named 'nowat.crd', RMS fit the coordinates to the first frames, then write out a second trajectory named 'nowat.rms.crd' which has no solvent and has the coordinates RMS-fit.

4) Use of 'trajin remdtraj' to process frames at a specific temperature from an ensemble of REMD trajectories:

   trajin rem.crd.000 remdtraj remdtrajtemp 492.2 1 11 2

   This will search for other replica trajectories named rem.crd.XXX, and process each in parallel, choosing only frames at 492.2 K for processing. In addition, only frames numbered 1 to 11, skipping every 2 will be used.

5) Same as script 2, except explicitly naming additional replica trajectories:

   trajin rem.crd.000 remdtraj remdtrajtemp 492.2 1 11 2 \
   trajnames rem.crd.001,rem.crd.002,rem.crd.003

6) Convert replica trajectories to temperature trajectories in one step:

   trajin rem.crd.000 remdtraj remdtrajtemp 492.20 remdout temp.crd

   This will search for other replica trajectories named rem.crd.XXX, and convert them to temperature trajectories named temp.crd.<Temperature>. 

7) **Automatic imaging:** DNA duplex, 30 base pairs, 58 ions, 13110 solvent molecules. Imaging will be performed based on centering the first molecule (strand 1) to the box center. The imaged trajectory will be written to ‘reimage.nc’ in NetCDF format.

```
parm dna30.parm7.gz
trajin split.duplex.nc
autoimage
trajout reimage.nc netcdf
```

8) **CPPTRAJ Hydrogen Bonding:** 22 residue beta sheet peptide. Searching for hydrogen bonds between all backbone oxygen (atom name 'O') acceptors and “standard” donors (i.e. F-H, O-H, N-H) in residues 2-20; hydrogen bond distance cut-off criterion is set to 3.3 Å. The number of hydrogen bonds vs. time will be written in xmgrace-compatible format to ‘nhb.agr’ due to the file name extension ‘.agr’.

```
parm DPDP.parm7
trajin DPDP.nc
hbond :2-20 acceptormask @O out nhb.agr avgout avghb.Odonor.dat \  
dist 3.3
```

9) **Nucleic acid structure analysis with ‘nastruct’:** Protein-DNA complex with 8-oxoguanine-damaged base. Residues 1-273 are protein, residues are 274-305 nucleic, residue 297 is 8-oxoguanine (8OG), and residue 306 is zinc. Base pair, base pair step, and helical step results will be written to BP.nastruct.dat, BPstep.nastruct.dat, and Helix.nastruct.dat respectively. Since 8OG is not a standard nucleic acid base it is mapped to ‘G’ (guanine) with the ‘resmap’ keyword.

```
parm nowat.8OG.parm7
trajin nowat.WT.DIST.500psTrajs.mdcrd.gz
nastruct naout nastruct.dat resrange 274-305 resmap 8OG:G
```

10) **Rotational diffusion:** The input trajectory (‘traj1.nc’) is 2000 frames. Rotation matrices are generated using an RMS-fit to a previously generated average structure using protein backbone atoms (Amber naming convention, CA, C, N, and O). 1000 random vectors (with random seed taken from wallclock time) will be rotated according to these rotation matrices; time correlation functions will be generated using the first 1000 points (ncorr) and a Langevin order of 2. Correlation functions will be integrated from 0.0 to 1.5 ns (time step 0.002 ns). The initial random vectors will be output to ‘rvecs.dat’, rotation matrices to ‘matrices.dat’, effective diffusion constants to ‘deffs.dat’, and all other command output to ‘rotdif.out’.

```
reference average.pdb
trajin traj1.nc 1 2000
rotdif rseed -1 nvecs 1000 reindex 0 @CA,C,N,O \  
dt 0.002 ti 0.0 tf 1.5 ncorr 1000 order 2 \  
itmax 1000 rvecout rvecs.dat rmout matrices.dat \  
deffout deffs.dat ncorr 1000 outfile rotdif.out
```
11) Atom mapping:
Re-order the atoms in 'xtallig.mol2' based on reference structure 'start.mol2'. Also illustrates use of parameter and reference file 'tagging'.

# Load structure to be remapped.
parm xtallig.mol2 [xtal_parm]
reference xtallig.mol2 parm [xtal_parm] [xtal]
# Load reference structure to be mapped to.
parm start.mol2 [start_parm]
reference start.mol2 parm [start_parm] [start]
# Create map from xtallig.mol2 to start.mol2, results to atommap.dat
atommap [xtal] [start] mapout atommap.dat
# Now that map is created, atoms can be reordered;
# load structure to be reordered.
# Remapping will happen automatically
trajin xtallig.mol2 parm [xtal_parm]
# Now that atoms in xtallig.mol2 have been remapped,
# start.mol2 can be used as reference.
rms 1g9v ref [start] out rmsd.dat
# Output remapped xtallig.mol2.
trajout fit.mol2 mol2
6) Example PTRAJ scripts:

1) Centering and imaging double-stranded DNA with 18 base pairs:

```bash
trajin traj.1.gz
center :1-18 mass origin
image origin center familiar
center :1-36 mass origin
image origin center familiar
rms first mass out rms :1-36
```

2) The PTRAJ hbond command; calculating solute-solute and ion-solute hydrogen bonds for double-stranded DNA with 18 base pairs:

```
# Imaging should be performed prior to these commands
# The prnlev command is used to give detailed info on selected atoms
#
prnlev 3
#
# backbone + sugar
#
donor mask @O?P
donor mask @O?'
acceptor mask :18@O3' :18@H3T
acceptor mask :36@O3' :36@H3T
acceptor mask :1@O5' :1@H5T
acceptor mask :19@O5' :19@H5T
#
# minor groove
#
donor DG N3
donor DG5 N3
donor DG3 N3
donor DA N3
donor DA5 N3
donor DA3 N3
donor DC O2
donor DC5 O2
donor DC3 O2
donor DT O2
donor DT5 O2
donor DT3 O2
acceptor DG N2 H22
acceptor DG3 N2 H22
acceptor DG5 N2 H22
#
# base pairing
#
donor DC N3
donor DC5 N3
donor DC3 N3
donor DA N1
donor DA5 N1
donor DA3 N1
acceptor DA N6 H61
```
acceptor DA3 N6 H61
acceptor DA5 N6 H61
acceptor DC N4 H41
acceptor DC3 N4 H41
acceptor DC5 N4 H41
acceptor DG N1 H1
acceptor DG3 N1 H1
acceptor DG5 N1 H1
acceptor DT N3 H3
acceptor DT3 N3 H3
acceptor DT5 N3 H3
#
#  major groove
#
donor DG O6
donor DG5 O6
donor DG3 O6
donor DG N7
donor DG5 N7
donor DG3 N7
donor DA N7
donor DA5 N7
donor DA3 N7
acceptor DA N6 H62
acceptor DA3 N6 H62
acceptor DA5 N6 H62
donor DT O4
donor DT5 O4
donor DT3 O4
acceptor DC N4 H42
acceptor DC3 N4 H42
acceptor DC5 N4 H42
#
#  Turn off detailed info
#
prnlev 0
#
#  keeping track of ion interactions (e.g. treating ions like solvent)
#  angle -1.0 turns off the angle criterion
#
hbond series hbond out hbond_k.out \ 
solventdonor K+ K+ \ 
solventacceptor K+ K+ K+ \ 
time 1.0 angle -1.0 distance 3.5

hbond series hbond out hbond_cl.out \ 
solventdonor Cl- Cl- \ 
solventacceptor Cl- Cl- Cl- \ 
time 1.0 angle -1.0 distance 3.5