1 Purpose of *files4amber*

files4amber is used to generate the parameter-topology and coordinate files required for simulations with various Amber modules (mainly *sander*, *pmemd*, and variants thereof). It requires AmberTools17 or later and does probably work correctly with earlier versions. It can prepare the required files for smaller organics ("ligands") alone, receptors ("proteins") alone, and protein/ligand complexes. The files can include explicit TIP3P water (default) or just generate the required files for implicit-solvent simulations.

2 File types required or generated by *files4amber*

***.pdb: input** standard PDB file;

***.sdf: input** standard SDF (MDL) file with all hydrogens, bond orders, tautomeric and protonation state;

***.leap.crd: output** Amber format coordinate files;

***.leap.prm: output** Amber parameter-topology files;

***.leap.pdb: output** PDB file (the ".leap" indicates that they were created via *tleap*; such files have all hydrogen atoms;

***.frcmod: output** file (ligand-related) that may contain additional force field parameters (generated automatically) not part of the original parameter files;

***.ac.mol2: output** file (ligand-related, generated internally via the *antechamber* module); it has SYBYL mol2 format, but atoms have GAFF/GAFF2 force field atom types; it also contains the partial charges generated via the AM1/BCC method;

sqm.*: output left overs from the *sqm* module used to compute partial charges in the ligand;

***.cmd: output** file containing the commands for the *tleap* module; this file can also be submitted to *tleap* via the command `tleap -f filename.cmd`; useful for editing further (which possibly requires reading some of the LeaP documentation in the Amber documentation);

3 Running *files4amber*

For help, just type files4amber and RETURN. You then get this:

```
-------------------------------------------
 files4amber version 1.0
 Romain M. Wolf (February 2019)
-------------------------------------------
Usage: files4amber [options]

Options:
  -h, --help      show this help message and exit
  --prot=FILE     protein PDB file                          (no default)
  --pfrc=STRING   protein force field                 (default: ff14SB)
  --disul=FILE    file with S-S definitions in protein    (no default)
  --lig=FILE      ligand MDL (sdf) or SYBYL file (mol2)   (no default)
  --chrg=INTEGER  formal charge on ligand                 (default: 0)
  --colig=FILE    co-ligand file (no extension)           (no default)
  --lfrc=STRING   ligand force field                   (default: gaff2)
  --cplx=FILE     name for complex files                  (no default)
  --solv=STRING   explicit [exp] or implicit [imp]      (default: exp)
  --buffer=FLOAT  solvent box buffer zone             (default: 12.0 A)
  --neut          neutralize with ions    (default: don't neutralize)
  --rad=STRING    radius type for PB/GB              (default: mbondi2)
  --ctrl=FILE     leap command file name            (default: leap.cmd)
```

4 Command line options

`--prot` must be followed by a protein PDB file ("Amber-clean");

`--pfrc` specifies the protein force field to be used. The default *ff14SB* is a good choice at this point and selecting another FF needs good reasons;

`--disul` can be used to create disulfide bonds; the option specifies a text file that contains on each line the residue numbers of the two cysteines to be connected; these residues must also be renamed CYX in the original PDB file;

> **note** that this option is not required if the disulfide bonds are specified by CONECT records in the original PDB file (connecting the respective SG, i.e. γ sulfure atoms); but in any case, cysteines involved in disulfides must be renamed CYX; see the sheet entitled "*cleanprotein*" for further details;

`--lig` must be followed by an SDF file, including all hydrogens and reflecting the correct bond orders, tautomeric and protonation state (and coordinates corresponding to a reasonable geometry, i.e., embedding the ligand into the receptor in the desired starting configuration);

> **note:** the residue name for ligand in all generated files will always be **LIG**!

`--chrg` must be specified if the ligand has a formal charge; omitting this option with a charged ligand leads to a failure of the partial charges computations via AM1/BCC and the routine stops;

`--colig` must be followed by the "co-ligand" name (no extension); this feature is described in more detail below and and also details in the separate sheet about the routine *lmw4amber*;

`--lfrc` selects the ligand (small organic) force field; *gaff2* is the default, the only other option would be *gaff*;

`--cplx` is followed by a name for the complex;

> **if not used**, or if only a ligand or only a protein have been specified, **no complex will be formed**

`--solv` specifies the solvation; the default is `exp` for explicit and will create a TIP3P water box around each solute (ligand, protein, and/or complex); specifying `imp` will prepare the files for implicit solvent (GB) simulations, i.e., no water is added;

`--buffer` is used in the case of explicit water to specify the zone around the solute that is filled with water; the default of 12 Å is OK (a) if a cutoff of non-bonded interactions 12 Å or less is used, and (b) if it cannot be expected that the solute will unfold to assume at least one much larger dimension, which could lead to periodic boundary problems (e.g., the original interacting with one of its own images); note that for a smaller organic alone (if defining only `--lig` without forming complexes with a protein), the surrounding solvent box will be isometric instead of just orthorhombic;

`--neut` will neutralize the system by adding the adequate number of Na+ or Cl- ions, depending on the overall formal charge of the system. The position of the counter-ions is controlled via the simplified electrostatic potential method of the "`addions`" command in *tleap*; ions are added **before** the explicit solvation by TIP3P waters, i.e., they become parts of the system to be solvated; counter-ions can also be added in the case of implicit solvent approaches, when using `--solv imp` (up to the user to decide if this is useful);

`--rad` can be used to change the default selection for radii used in GB and PB computations; the `mbondi2` default is a good choice is the GB settings `igb=5` are used; keep the default if in doubt what to use;

`--ctrl` allows to specify the name (default is *leap.cmd*) for the generated command file for *tleap*, the routine that actually generates the Amber-related files; for additional or changed *tleap* commands, you may edit this file and then re-process it through *tleap* with the command `tleap -f leap.cmd`; if you intend to do that, it is a good idea to use this option to change the default name, in order to avoid later overwriting of this file or at least to clearly identify it.

5 Remarks

1. In order not to overwrite files with similar names, always create a new subfolder to generate Amber files for a specific simulation.

2. Files for the same system of the types `*.crd` and `*.prm` are inseparable pairs. Coordinates might be changed (in general, **don't**), but the order of atoms **must not** be altered in both files.

3. Complexes are formed using the initial coordinates. There is **no docking** whatsoever. In other words, forming complexes between receptors, ligands, co-ligands, works on the **original coordinates** of each species.

   > Always make sure that all entities are in their correct "world" coordinates.

   The resulting `*.leap.pdb` files are a good control. If in doubt, open them in some compatible visualisation software to verify that everything is in place before running lengthy simulations.

6 Clarifications to the option `--colig`

In many cases, a second ligand GDP, ATP, co-factor, and alike, must be included. To create specific `.ac.mol2` and `.frmod` files, use the *lmw4amber* routine, described on a separate sheet. The *lmw4amber* ("low-molecular-weight for amber") routine generates the required files, but in addition, also creates other special files useful for other specific purposes.

Since *files4amber* has to read both the `.ac.mol2` and `.frcmod` files for this second ligand, do NOT include any extension. Just enter the global file name.

Since the standard ligand gets automatically assigned the "residue" name "LIG", choose another name for the additional low-molecular-weight compound to be included! This is explained in the separate sheet concerning *lmw4amber* and is handled by the command line option `--name` in that routine.