Installation Karlsberg

Leonardo Preuss

August 2022

1 Prerequisites

First compilers for fortran and c++ need to be installed, using the following command:

```
$ sudo apt install g++ gfortran
```

Then three different shells need to be installed: bash, csh and tcsh

```
$ sudo apt install bash csh tcsh
```

For installing charmm, cmake and make are needed:

```
$ sudo apt install cmake make
```

Now a directory called kb2plus_package should be created where with provided tars should be placed. To untar the provided tar-files every use the following command in the directory containing the tars:

```
$ for file in ./*;do tar -xvzf "$file"; done
```

2 charmm installation

To install charmm you have to use the following commdands

```
$ cd charmm
$ ./configure
$ cd build/cmake
$ sudo make
$ sudo make install
```

3 Configuring Python

In order to use karlsberg python 2.7 and a few libraries are needed. This is best done using a virtual enviroment with conda. In a directory of your choice perform the following commands:

```
$ wget https://repo.anaconda.com/miniconda/Miniconda3-latest-Linux-x86_64.sh
$ bash Miniconda3-latest-Linux-x86_64.sh
```

Then a conda enviroment for karlsberg can be created and activated like this:

```
$ conda create --name kbp2 python=2.7
$ conda activate kbp2
```

To install the python packages with the correct version pip can be used to install them inside our enviroment:

```
$ pip install numpy==1.8.2 matplotlib==1.5.3 Biopython==1.65
  MDAnalysis==0.15.0 pyparsing==2.2.0 python-dateutil==2.4.2
  six==1.10.0
```

Since kbp2, the central package containing all the python modules, is not installable via pip, it needs to be added to the users pythonpath. For this the following needs to be added to the .bashrc file in the users home directory (replace <path> with the location of your kb2plus_package)

```
export PYTHONPATH="<path>/kb2plus_package"
```

4 Dependencies

The precompiled binaries in the bin directory provided need certain libraries to run. This can be taken care of by adding the following to the .bashrc file (replace <path> with location of your kb2plus_package):

```
export LD_LIBRARY_PATH="<path>/kb2plus_package/APBS-1.5-linux64/lib:<path>/
kb2plus_packe/lib_fort_gnu:$LD_LIBRARY_PATH"
```

To add these to the in the path of the current session:

```
$source ~/.bashrc
```

5 Testing Karlsberg

To test that Karlsberg is installed correctly, a directory called test-example is provided. Inside this directory there is the file get_pkas_example.py. This file needs to be edited, because it contains relatives paths and python2 can only work with absolute path. In order to do this Ctrl + H can be used to replace every ".." by the path to the python package.
Also the paths to the charmm binary and the working directory need to be changed. For this Ctrl + H can be used to search for:
`<user folder of choice unrelated to the package>`
and replaced with the path to the test_example directory.

Also `<charmm binary filepath>` should be replaced by the actual path tom the charmm binary (<path to charmm directory>/charmm/bin/charmm). Finally inside the kb2-package the file karlsberg.py needs to be changed. In line 165 the string:

```
../kb2plus_package/binaries_karlsberg/bin/karlsberg2.x86_64_fixed
```

needs to be changed to:

```
<pathw>/kb2plus_package/bin/karlsberg2.x86_64_fixed
```

Karlsberg can be tested inside the test_example directory:

```
$python2.7 get_pkas_example.py
```

6 Other Problems

A common problem source when installing karlsberg is when karlsberg interacts with the different shells (csh, tcsh, bash). One of the locations in the code where this happens is in the karlsberg.py module in line 326:

```
process = subprocess.Popen("csh", \
                           shell=True, \
                           stdin=subprocess.PIPE, \
                           stdout=subprocess.PIPE, \
                           stderr=subprocess.PIPE, \
     )
```

Here one fix could be trying to change "csh" to "tcsh".

Another problem that does not occur in the test example, but when karlsberg uses apbs, is found in the apbs_manager.py module in line 212 and 338 it should also be tested if changing the "csh" to "tcsh" or "bash" fixes the problem.