Installing CEA on an Intel Mac

Jonah Zimmerman*

December 11, 2013

1 Introduction

Chemical Equilibrium with Applications (CEA) is a computer program published by NASA Glenn with the purpose of calculating chemical equilibrium product concentrations from any set of reactants and determining thermodynamic and transport properties for the product mixture [1].

The reason that you're reading this document is that you'd like to install CEA on an Intel-based Apple computer, which is an unfortunately complicated procedure. Before continuing I suggest checking out the computer program Rocket Propulsion Analysis [2], which provides many of the same capabilities of CEA but in a user-friendly interface that installs and runs easily on modern Macs and PCs. There is also a web-based version of CEA accessible at `http://cearun.grc.nasa.gov/`.

The process presented in this document was compiled from the notes of Professor Brian Cantwell, Siina Haapanen, Minna Chao, Ashley Chandler, and from the help of Matthew Giarra, Mikhail Ozhibesov, Osa Igbinosun, Tom Colvin, and from references [3] and [4]. It has been tested in OS X 10.9 (Mavericks) in December of 2013. This tutorial is written for Mavericks users but an older version for Mountain Lion, Lion, Leopard, and Snow Leopard users is available from `http://spase.stanford.edu/CEA.html`.

2 Dramatis Personae

The characters involved in this play include:

CEA - a Fortran program coupled with a Java user interface

gfortran - GNU compiler for Fortran programs (free alternative to UNIX's f77)

gcc - GNU compiler collection - a set of compilers for various programming languages

terminal - OS X's porthole to the UNIX command line

XCode - a set of tools for developing software on OS X

root - the "super-user" on a UNIX machine

emacs - a UNIX text editing program

---

*Graduate Student, Stanford University, jonah.e.zimmerman@gmail.com

3   Terminal

There are many ways of getting to the UNIX command line which is buried under the shiny beautiful surface that is the OS X user interface. One of the simpler, no-frills methods is using the program Terminal which is located in the utilities folder inside of the applications folder. This opens the `bash` shell by default, and if it's not you can switch back through Terminal's preferences menu (if it's running `bash` already, it will say so at the top of the window):

Open Terminal preferences, and find the option under "startup" that will automatically enter a command each time Terminal is opened. Make this command `/bin/bash`. Alternatively, for a single window you can just type the command `bash` to change it for that window only.

4   XCode

You will need to install XCode to obtain the `gcc`. It can be obtained for free online via Apple's App Store. Just search for XCode and it'll come up. Once you've downloaded and installed it, run the program from your Applications folder.

Then, go to `Preferences` → `Downloads` → `Components` and install the `Command Line Tools`. This step may require you to register as a developer on Apple's website. It's a bit of a pain but unlike most things from Apple it's free.

5   gfortran

To make CEA function, you need to recompile some of the Fortran programs. To do that, you of course need some way of compiling Fortran programs! With XCode `Command Line Tools`, you've installed `gcc`, a compiler for a wide range of programming languages that by default doesn't include Fortran. To obtain a Fortran compiler, `gfortran`, you must head to `http://hpc.sourceforge.net/`.

There you can find under the heading "**GCC 4.7, 4.8, 4.9 (auto-vectorizing gcc with OpenMP)**" the links for downloading the gfortran files, specific to various versions of OS X. For this installation, only the `gfortran` file for Mavericks is needed: `gfortran-4.9-bin.tar.gz`. Once you've downloaded the `gfortran` package, you'll need to move it to somewhere useful, unzip it, and install it (we'll go through how to do that next).

Depending on the setup of your computer and user profile, you may need to switch to the root user of your system in order to complete some of the following commands. To switch to root, simply type in the terminal window `su root`, and then enter the appropriate password.

If you are not the owner of your computer, you should probably contact the owner before trying to continue. If you are the owner of the computer and you don't know the root password, first try your own user's password. If that doesn't work, the root password may never have been set. You can set it using the command `sudo passwd root`. Then enter your own password, and set the root password to whatever you'd like.

Continuing with the installation, first you'll need to make a directory in which to put `gfortran`:

```
mkdir /usr/local/src/gfortran
```

However, if `/usr/local/src` doesn't exist (you'll get an error when you try to type the above command), you'll first have to make it:

```
mkdir /usr/local/src
```

Then to move the downloaded file to the target directory, first navigate to the location where it was downloaded, and then use the following commands:

```
cp gfortran-4.9-bin.tar.gz /usr/local/src/gfortran
```

Now move the working directory to that folder:

```
cd /usr/local/src/gfortran
```

and then unzip it:

```
gunzip gfortran-4.9-bin.tar.gz
```

and then install it:

```
tar -xvf gfortran-4.9-bin.tar -C /
```

to test if it installed correctly, simply type `gfortran -v`. If it returns something similar to:

```
sh-3.2# gfortran -v
Using built-in specs.
COLLECT_GCC=gfortran
COLLECT_LTO_WRAPPER=/usr/local/libexec/gcc/x86_64-apple-darwin13.0.0/4.9.0/lto-wrapper
Target: x86_64-apple-darwin13.0.0
Configured with: ../gcc-4.9-20130929/configure --enable-languages=c,c++,fortran
Thread model: posix
gcc version 4.9.0 20130929 (experimental) (GCC)
```

then you're all set (some of it will be different depending on your computer and version of OS X).

If it returns `command not found`, then something's wrong. One possible reason is that the installed directory isn't in the normal search path, so your computer isn't seeing it. To see if this is the case, type: `echo $PATH`, and look at the list of directories that it outputs. If `/usr/local/bin` was not one of the directories, then you need to add it to the path. This can be done by typing

```
echo 'export PATH=/usr/local/bin:$PATH' >> ~/.profile
```

and then closing your terminal window, quitting the program, and then restarting terminal with a new window. Type `echo $PATH` to make sure it worked. If this also didn't work, see the notes at the end of the next section about the alternate method for updating the `PATH` variable.

If for whatever reason you decide to uninstall **gfortran**, go to the directory that you installed it *from* (ie `cd /usr/local/src/gfortran/`) and run this code:

```
tar -tf gfortranfile | sort -r | (cd /; xargs -t -n 1 rm -d)
```

6 CEA Files

To download the CEA files themselves, go to the following page on NASA Glenn's site:

   `http://www.grc.nasa.gov/WWW/CEAWeb/ceaguiDownload-unix.htm`

From here, download the following three files:

- CEAgui JAR (CEAgui-jar.tar.Z)

- CEA+Fortran Package (CEA+Fortran.tar.Z)

- CEAexec Package (CEAexec-mac.tar.Z)

Make a new directory and save all of these files in it This will be where CEA is installed, so place it wherever you'd like the final program to be. Then, open a terminal window and navigate to the folder where the files are, and use the commands below to uncompress the files. This can be accomplished using a program like Stuffit Expander or The Unarchiver or Archive Utility, but these programs by default will usually create new folders and place each file's uncompressed contents into them. So, if you're set on using one of them just make sure you place all the uncompressed files back into your CEA directory.

```
zcat CEA+Fortran.tar.Z | tar xvf -
zcat CEAexec-mac.tar.Z | tar xvf -
zcat CEAgui-jar.tar.Z | tar xvf -
```

When you're finished, you should have the following six files in this folder: `CEAgui.jar, thermo.lib, trans.lib, syntax, b1b2b3, FCEA2, runCEA.sh`. If not all six are present, check that you have correctly downloaded and unpacked all three packages. Three of these files have been previously compiled on a different OS and won't function. Later we will recompile them, but for now we just delete them to make sure we don't end up using the wrong versions:

```
rm syntax b1b2b3 FCEA2
```

Now, the permissions of one of the files have to be changed. This can be done using the following command:

```
chmod a+x runCEA.sh
```

Finally the environment variable `PATH` must be changed to include the CEA installation directory. Do this by typing:

```
echo 'export PATH=/directory/where/CEA/is:$PATH' >> ~/.profile
```

And then quitting and restarting terminal (make sure you replace `/directory/where/CEA/is` with the actual directory!). Type `echo $PATH` and make sure that your CEA installation directory is on the list.

For some people this method of updating `PATH` does not work, and while I have been unable to determine the reason, there is another way to do it. It involves opening a text document that contains a list of directories, and then adding the CEA directory to that list. The text editor of choice is `emacs`, and opening the file into it can be done by typing (note that you will need to be logged in as root for this step):

```
emacs /etc/paths
```

Now, navigate to the bottom of the file with the arrow keys, and then add on the next line the directory where CEA is installed. For example, once I added the directory to my PATH file, it looked like this:

```
/usr/bin
/bin/
/usr/sbin
/sbin
/Applications/CEA
/usr/local/bin
```

Note that I didn't add it to the end. I've found that when it's the last item in the PATH file it can sometimes cause some problems.

To exit emacs hit ctrl-x and then ctrl-c. When it asks you if you want to save the file, type y.

7 Update CEA Files

Now, some of the CEA Fortran files need to be recompiled. To do that, move to the CEA installation directory and type the following code:

```
gfortran -o FCEA2 cea2.f
gfortran -o b1b2b3 b1b2b3.f
gfortran -o syntax syntax.f
```

If during this process you get a long string of warning messages about feature deleted: goto ... it's bogus and safe to ignore. Now you should have new executable files, but you'll still need to get new library files. To do that, type:

```
FCEA2
trans
FCEA2
thermo
FCEA2
cea2
```

Note that each time you type FCEA2 you should get a statement asking for the input filename. You should now be all set. To try it out, type ./runCEA.sh in the CEA installation directory, and try loading an example file and executing it.

8 Troubleshooting

This section is still in progress. If you have any problems, let me know and I'll add it here.

There are many ways to go wrong installing this software, so here is a list of common errors and what they mean:

File not found! Runtime.getRuntime().exec(b1b2b3) not complete due to the missing thermo.lib or trans.lib or BAD data! This means that you didn't update the CEA files correctly. Make sure that you didn't get any error messages when completing that part of the installation.

ERROR: Missing Fortran Executable File: FCEA2.exe or FCEA2 ! Please Exit! This means that the PATH environment variable doesn't include the CEA installation directory.

`as: assembler (/usr/bin/../libexec/gcc/darwin/x86_64/as or /usr/bin/../local/libexec/gcc/darwin/x86_64/as) for architecture x86_64 not installed`

`as: no assemblers installed`

I don't entirely know the details behind this error, but it seems to be related to the installation of the `gcc`. Reinstalling XCode's `Command Line Tools` and then proceeding with the installation from there has solved this error in the past.

gfortran: error trying to exec 'as': execvp: No such file or directory This means that the `gcc` was not installed correctly. Make sure that the `Command Line Tools` have been installed via XCode.

References

[1] http://www.grc.nasa.gov/WWW/CEAWeb/ceaWhat.htm

[2] http://www.propulsion-analysis.com

[3] http://hpc.sourceforge.net/

[4] http://www.webmo.net/support/fortran_osx.html