Getting TEMOA running with CPLEX and SQLite

By Jeff Bennett, [jab6ft@virginia.edu](mailto:jab6ft@virginia.edu), 9/8/2017

The purpose of this document is to formally record how to get TEMOA running with CPLEX and SQLite so that it can be quickly transferred to another machine. The steps are based on <http://www.temoaproject.org/download/> and Joseph Saylor’s readMe file with elaborated steps to make it more straightforward for a user with little or no python experience. These steps were developed for a Windows machine.

Notes: anything in italics are code segments; skip installation of any software already present on the machine

1. Install **Git**
   1. Purpose: Version control software
      1. <https://www.slideshare.net/HubSpot/git-101-git-and-github-for-beginners>
      2. <http://product.hubspot.com/blog/git-and-github-tutorial-for-beginners>
   2. Download latest version from <https://git-scm.com/>
   3. Installation instructions
      1. Select the second option "Use Git from the Windows Comand Prompt" to ensure git is added to your computer's path
      2. Select "Use the OpenSSL library"
      3. Select “Checkout Windows-style, commit Unix-style line endings”
      4. Select “Use Windows’ default console window”
      5. Leave configuration options as-is
      6. Finish installing.
2. Install **Anaconda2 for Python 2.7**
   1. Purpose: Distribution of the Python language and common libraries
   2. Download the latest version of Anaconda2 for Python version 2.7 <https://www.anaconda.com/download/>
   3. Install using default settings
3. Install **Pyomo**
   1. Purpose: Library from Sandia National Labs for optimization modeling
   2. Installation steps:
      1. Start Anaconda2 terminal (Programs>Anaconda2>Anaconda Prompt)
      2. Verify version of Python

*python --version*

* + - 1. If Python 2.7.X does not appear, then troubleshooting is necessary
    1. Install Pyomo

*pip install pyomo*

*pip install pyomo.extras*

Note: If errors – may need to first install MS Visual C++ Compiler for Python https://www.microsoft.com/en-us/download/details.aspx?id=44266

* + 1. Exit Anaconda

*exit*

1. Install **CPLEX**
   1. Purpose: Powerful linear optimization solver
   2. Navigate to CPLEX website <https://www.ibm.com/developerworks/community/blogs/jfp/entry/CPLEX_Is_Free_For_Students?lang=en>
   3. Select the appropriate link either a) Students or b)Teachers, researchers and university staff
   4. Follow the steps to create an account using your institutional email address (i.e. [abc123@virgina.edu](mailto:abc123@virgina.edu)) and download CPLEX
   5. Install using default settings
2. Install **DB Browser for SQLite**
   1. Purpose: Viewer for SQLite files
   2. Download the latest version <http://sqlitebrowser.org/>
   3. Install with default settings
3. Install **TEMOA**
   1. Select the directory where TEMOA will be installed and record the pathname, i.e. “C:\”
   2. Decide what name to give the folder with temoa files, “temoa” is recommended and is used below
   3. Start Windows command prompt terminal (Start, type cmd, hit enter)
   4. Navigate to the path

*cd pathname*

* 1. Clone git (copies all of TEMOA’s files from the repository)

*git clone https://github.com/TemoaProject/temoa/*

1. Install **GLPK**
   1. Purpose: Required to prepare input files for CPLEX
   2. Download the latest version: <https://www.gnu.org/software/glpk/> or for Windows: <https://sourceforge.net/projects/winglpk/>
   3. Unzip and copy files to location of your choice
   4. Record directory to w32 folder (for 32 bit OS) or w64 folder (for 64 bit OS)
   5. Add directory to path
   6. Test glpsol works
      1. Start windows command prompt (Start, type “cmd”, enter)

*glpsol --help*

* + 1. Output should resemble this:



1. Optional: Install **Notepad++**
   1. Purpose: Text editor with increased abilities for searching and highlighting code syntax
   2. Download latest version from <https://notepad-plus-plus.org/>
   3. Install using default settings
   4. It is recommended to make this the default program for opening \*.txt files
2. Create test cases
   1. Navigate to the directory “temoa\_model” in the directory where TEMOA is installed
   2. Open config\_sample
   3. Save As a new file (if using Notepad++)
      1. File name: “config\_sample\_cplex”
      2. Save as type: “All Types(\*.\*)”
   4. *Confirm the below lines are set as follows*
      1. Line 11: *--input=db\_io/temoa\_utopia.sqlite*
      2. Line 15: *--output=db\_io/temoa\_utopia.sqlite*
   5. *Update the code as follows*
      1. Line 19: *--scenario=test\_run\_cplex*
      2. Line 35: *--solver=CPLEX # Optional, indicate the solver*
   6. *Save*
   7. Save As a new file (if using Notepad++)
      1. File name: “config\_sample\_glpk”
   8. *Update the code as follows*
      1. Line 19: *--scenario=test\_run\_glpk*
      2. Line 35: *--solver=GLPK # Optional, indicate the solver*
3. Run test case
   1. Start Anaconda2 terminal (Programs>Anaconda2>Anaconda Prompt)
   2. Verify version of Python

*python --version*

* + 1. If Python 2.7.X does not appear, then troubleshooting is necessary
  1. Change directory to the location of temoa (shown as “C:\temoa”)

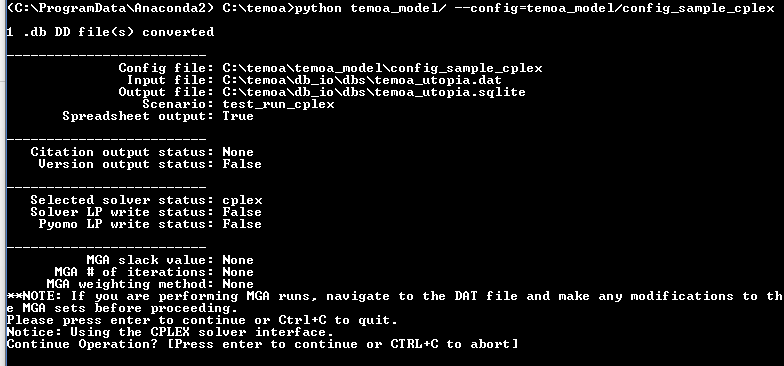
*cd C:\temoa*

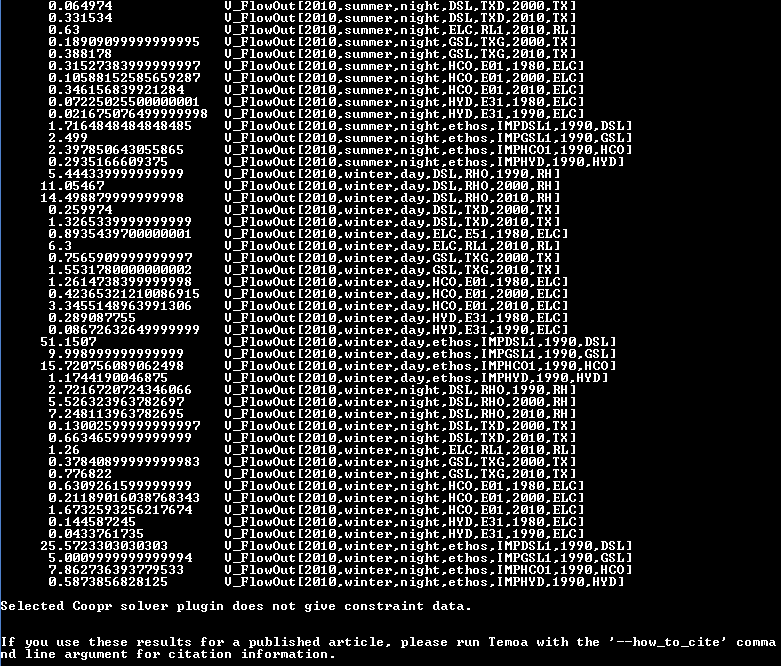
* 1. Run test case 1

*python temoa\_model/ --config=temoa\_model/config\_sample\_cplex*

* 1. *Expected Results*

After submitting command:



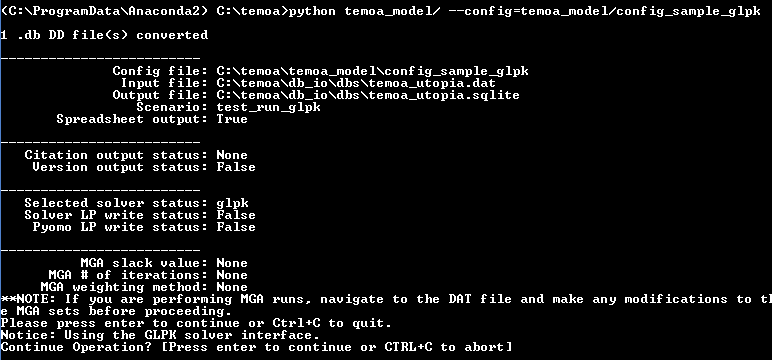
After hitting enter to continue: 

* 1. Run test case 2

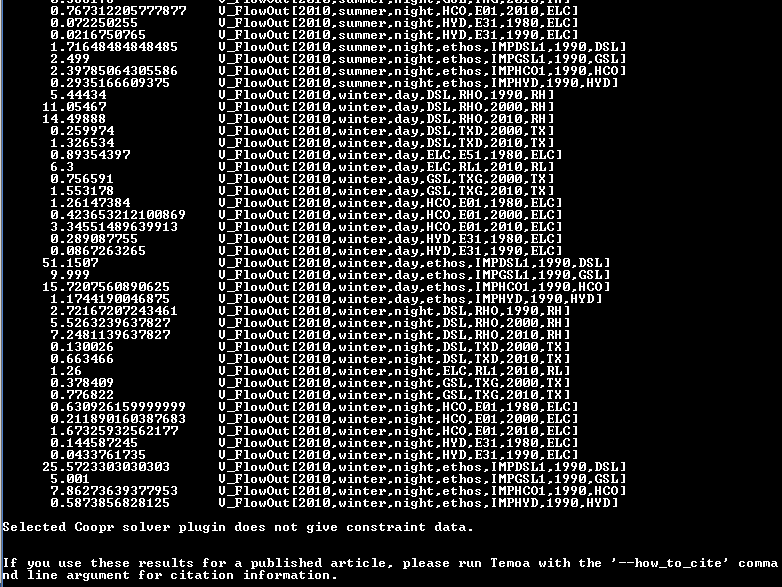
*python temoa\_model/ --config=temoa\_model/config\_sample\_glpk*

* 1. Expected Results

After submitting command:



After hitting enter to continue:



* 1. Exit Anaconda

*Exit*

Addendum – Background on TEMOA

DeCarolis, J., Hunter, K., Sreepathi, S., “The TEMOA Project: Tools for Energy Model Optimization and Analysis,” International Energy Workshop 2010, 21-23 June 2010, Stockholm, Sweden.

Hunter, K., Sreepathi, S., DeCarolis, J., “Modeling for insight using Tools for Energy Model Optimization Analysis (Temoa)”, Journal of Energy Economics, 40 (2013), 339-349.

Addendum – Background on creating TEMOA configuration files

* Example located in temoa\_model\config\_sample
* Config files can have any file extension, .txt makes it easy to read in
* There are a number of flags that you need to specify (mandatory and optional)
* first two are source of input and source of output: where to direct
* if running from database, input and output will be there same
* there will be a set of database tables that contain input data that you specify for your energy system
* blank tables will contain output data
  1. two different flags to specify text file as input and database as output if wanted
* name the scenario: model stores results in output tables and indexes by scenario name so results get aggregated an you can query by name of scenario (mandatory)
* need to specify path to db\_io folder (database input/output) for necessary files and scripts
* optional files:
  1. save excel: saves output to excel workbook where each sheet contains different sets of data, dividing by sectors with each sector having a different sheet for activity and capacity
  2. save text file: saves log of information from shell
  3. solver: specify solver if needed
  4. keep pyomo lp file: lp file gets generated by pyomo and set to solver to solve – helps with debugging
  5. modeling to generate alternatives: for uncertainty analysis
     1. if choosing to run modeling to generate alternatives (mga): need to specify slack file, iteration number, how many mga iterations you want to generate, and the method by which you want to update the objective function: either integer or normalized weighting