Introductory Guide for SMB Optimizer on Windows 10

Version 1.3

SMB Optimizer: https://github.com/suzuki1969/Python-based_SMB_Optimizer

Kensuke Suzuki
suzuki.kensuke@f.mbox.nagoya-u.ac.jp
Department of Materials Process Engineering
Nagoya University

March, 03th 2021

Original code and document developed by Zeyu Yang, Georgia Tech, USA

This documentation shows how to install the SMB Optimizer software package for optimization of simulated moving bed process with IPOPT.

The code was originally developed by Suriya Arulselvan in December 2015, and later modified by Zeyu Yang in July 2016 at Georgia Institute of Technology in the US as their undergraduate research projects. Modifications for parameter estimation were made by Hideki Harada and Kensuke Suzuki at Nagoya University, Japan in 2019-2021.

Software and Package Installation

The following installation steps are confirmed for the following environment:

OS: Windows 10 pro ver.1909
MSYS2: msys2-x86_64-20190524
Ipopt: ver.3.13.3.

We have not confirmed installation for other operating systems such as Mac and Unix.

1. Install MSYS2/MinGW

MSYS2 is a UNIX-like environment for Windows. Go to the following website and download `i686` (for 32 bits) or `x86_64` (for 64 bits) depending on your operating system.

https://www.msys2.org/

In the following we assume that we use `x86_64`.

1) After installation, run 'MSYS2 MinGW 64-bit' from the Windows Start menu.
2) Run the following commands ($ indicates the command line prompt, do not type $):

```
$ pacman -S binutils diffutils git grep make patch pkg-config
$ pacman -S mingw-w64-x86_64-gcc
$ pacman -S mingw-w64-x86_64-gcc-fortran
$ pacman -S mingw-w64-x86_64-lapack mingw-w64-x86_64-metis
```

2. IPOPT **Compilation**

IPOPT is an open source software package for large-scale nonlinear optimization. The instruction manual of the latest version can be found here: https://coin-or.github.io/Ipopt/index.html

You can compile IPOPT following the official instruction “Download, build, and install dependencies”: https://coin-or.github.io/Ipopt/INSTALL.html.

When you finish the compilation of IPOPT, follow the instruction in the Section 3. “**Python and library installation**”.

NOTE: IPOPT ver.3.13.x (x means a number of version) did not work with our code (March 25, 2020). Although, currently, our program does not have any problem with IPOPT version 3.13.3, if you have similar problem above, we suggest we use an older version, 3.12 following the steps below:

1) Download IPOPT version 3.12.13 as a zip file from:

   http://www.coin-or.org/download/source/Ipopt

2) Unpack Ipopt-3.12.13.zip and put the folder ‘Ipopt-3.12.13’ in $MSYS2\home*your_user_name* where ‘*your_user_name*’ is your user name, and ‘$MSYS2’ represents the folder in which you installed MSYS2 on your computer (if MSYS2 is installed with the default setting, ‘$MSYS’ is ‘c:\msys64’).

   In the MSYS64 terminal, run following command:

```
$ cd Ipopt-3.12.13
```

3) Run the following commands to download some external packages, including, LAPACK ASL, and Metis.:

```
$ cd ThirdParty/Blas
$ ./get.Blas
$ cd ../Lapack
$ ./get.Lapack
$ cd ../ASL
$ ./get.ASL
$ cd ../Metis
$ ./get.Metis
```

   In addition, the Harwell Subroutine Library (HSL) is required. Follow the steps below:

   1) Go to http://hsl.rl.ac.uk/ipopt
   2) We need the “Coin-HSL Full (RC)” code. To download, click the ‘source’ link.
   3) Follow the instructions on the HSL website, read the license agreement, and submit the registration form.
   4) Wait for an email with a download link (this should take no more than one working day).
   5) Download the zip file of HSL.
   6) Unpack the zip file, then move and rename the resulting directory as follows:
   `$IPOPTDIR/ThirdParty/HSL/coinhsl`

where '$IPOPTDIR' is the directory in which you unpacked this document (here, note that , '$IPOPTDIR' corresponds to '/home/*your_user_name*/Ipopt-3.12.13' in MSYS2 terminal)

4) Compile Ipopt. Type the following lines:

```
$ mkdir Ipopt-3.12.13/build
$ cd Ipopt-3.12.13/build
$ ../configure
```

The above commands create a directory to configure and perform the configuration. After typing these commands, if the last output line shows 'configure: Main configuration of Ipopt successful' then everything worked fine. Otherwise, look at the screen output, check the config.log output files and/or consult:
http://www.coin-or.org/Ipopt/documentation/node19.html#ExpertInstall

5) Type the following lines,

```
$ make
$ make install
```

If the installation is completed, the IPOPT solver executable 'ipopt.exe' can be found in "/mingw64/bin".

6) Finally, we install sIPOPT to run our program

```
$ cd Ipopt/contrib/sIPOPT
$ make
$ make install
```

Further information about sIPOPT installation : https://projects.coin-or.org/Ipopt/wiki/sIpopt

3. Python and library installation

There are two ways to run the SMB Optimizer in Python: **A)** within the MSYS2 terminal, **B)** within the windows platform (recommended).

A). With MSYS2 terminal

1) Type the following lines in MSYS2 terminal

```
$ pacman -S mingw-w64-x86_64-python
$ pacman -S mingw-w64-x86_64-python-pip
$ pacman -S mingw-w64-x86_64-python-numpy
$ pacman -S mingw-w64-x86_64-python-matplotlib
$ pacman -S mingw-w64-x86_64-python-Pillows
$ pip install pyomo
```

2) Put all python scripts and CSV files of the SMB Optimizer above to $MSYS2\home\your_user_name

3) Type the following line to ensure the program works

```
$ python ##.py
```

Here, '`##`' refers to the name of the python run file of the SMB Optimizer, for instance;
'`Run_SMB_ParamEst`' in the following directory: https://github.com/suzuki1969/Python-based_SMB_Optimizer/tree/master/SMB/Model_Parameter_Estimation

B). With windows platform

1)Install Python (https://www.python.org/downloads/) and its libraries on your computer. The libraries you need are numpy:
https://numpy.org/

and matplotlib:

https://matplotlib.org/.

Recommended: It is enough to just install ANACONDA python platform from the official website : https://www.anaconda.com/
While installation of ANACONDA platform, it installs libraries including numpy and matplotlib automatically.

2) Make a folder on your computer as 'c:\ipopt', or any arbitrary folder of your choice.

3) Put the following files in the folder you made above: 'ipopt_sens.exe' and 'libgfortran-5.dll' in '$MSYS2\mingw64\bin' ('c:\msys64\mingw64\bin' in the default setting).

4) Add the directory 'c:\ipopt' to Windows PATH environmental variables referring to the instruction: https://helpdeskgeek.com/windows-10/add-windows-path-environment-variable/.
Please be careful to change environmental variables correctly; the wrong setting may affect some other software on your computer.

5) Check if the program works by typing the following line on the Windows command prompt.

```
C:\Users\your_user_name> ipopt -v
```

If the terminal shows the version of Ipopt, go to the next step. If not, check the above steps again. In the case that the terminal returns the error message about absence of some dll files, copy the relevant dll file to the folder ('c:\ipopt') from '$MSYS2\mingw64\bin'.

6) Install pyomo referring to the official manual:
https://pyomo.readthedocs.io/en/stable/installation.html

After following the above steps, all software and packages should have been successfully installed and ready to use.