Water Software/Code Workshop Series
Thursdays, 1 - 2 pm, CEE Conference Room (ENLAB 211)

Topic #2: Interfacing GAMS and Matlab GUI

April 7, 2016
(Updated April 8, 2016)
David E. Rosenberg

Why Interface Matlab and GAMS?

The General Algebraic Modeling System (GAMS) provides a flexible high-level programming language to formulate and solve optimization models with numerous commercial and academic solvers. GAMS specifies inputs and displays results in text file format. In contrast, Matlab formulates optimization models in matrix format, offers a limited set of optimization solvers (through the linprog, fmincon, etc. functions), but has numerous matrix processing, plotting, and graphical user interface (GUI) tools and controls. The GAMS data exchange (GDX) format allows users to interface and automate data exchange between the two software environments to leverage both program's strengths. This workshop demonstrates the data exchange for an example linear programming problem that maximizes profit from planting two crops subject to land and water constraints. Workshop participants will build a GUI in Matlab to enter input data on crop profitability, use the *wgdx*, *gams*, and *rgdx* functions to pass the entry to GAMS, run the optimization model four times to examine model sensitivity to water use by tomato crops, and read GAMS results and plot them in Matlab.

Requirements

- GAMS version 24.1 or later
- Matlab version 2015a or later

Before the Workshop

A. Download and install GAMS from http://gams.com
B. Install Matlab (see Paul Rew for a license)
C. Download the GitHub repository (https://github.com/dzeke/GAMS-Matlab-GUI) and unzip
D. Open the GAMS IDE (double click **IDE icon** created in step A or select from Program files)
    i. Open a new project in the folder created in step C (**File**=>**Project**=>**New Project**),
    ii. Open the file *Ex2-1-parametric.gms* (File=>Open) and select **File**=>**Run** (or press **F9**),
    iii. A new window will open and after a few seconds, the screen will show ***** Status: Normal completion** in blue at the bottom. The crop optimization model has solved.
E. Add the folder where you installed GAMS (e.g., C:\GAMS\win64\24.7) to your Windows path (to both run GAMS from the command prompt and allow Matlab to call GAMS). For more specific directions to modify the Windows Path, see Omar Alminagorta's SWAMPS Model Workshop Guide, Appendix 1, Page 9, Part A, Steps 1-5 (http://tinyurl.com/hk5uajo).
F. Back in Matlab, also add the GAMS folder to your Matlab path (**Home** ribbon => **Environment**=>**Set path**=>click **Add with subfolders…**). Then navigate to the GAMS folder.

Workshop Activities

1. In Matlab, open the Graphical User Interface for model input (file *PlantModelDemo.m*). Navigate to the folder you created in Initial Step C and double click the file. On the **Editor** ribbon, press the green arrow above **Run**. The GUI will open. Modify some profit data and press **Run GAMS**.

2. The data is written to the file *PlantProfit.gdx*. To confirm, use the GAMSIDE to view the file (**File** menu=>**Open**; in the Open window, select **Files of type**: *GDX files (*.gdx)* and select *PlantProfit.gdx*). Select the parameter *c_inp* to view values.
3. Add code to the GAMS model file *Ex2-1-parametric.gms* to read in the data. Below the declarations of parameters *c(plnt)*, *b(res)*, and *c_inp(plnt)*, add 6 lines (4 code, 2 comments):

```
*Load user-specified profit data from Matlab (GDX file)
$GDXIN PlantProfit.gdx
$LOAD   c_inp
$GDXIN
*Assign input to model parameter
c(plnt) = c_inp(plnt);
```

4. Run the *Ex2-1-parametric.gms* model (press **F9**) to verify the code entry. The model will create the output file *Ex2-1-parametric.gdx*. Still in the GAMSIDE, open this file (see Step 2) to examine the contents (all model inputs plus model outputs *DecVars*, *ObjFunc*, *ShadowVals*, *TomWatReq*).
5. Return to Matlab to complete the Matlab GUI => GAMS => Matlab workflow.
    a. Open the PlantModelDemo.m file (code associated with the GUI), and scroll down to line 81 (start of **Run GAMS** pushbutton event)
    b. Comment out lines 102 and 103 (add % before) and **Save** the file.
    c. Note that line 106 calls GAMS and lines 109+ read and plot the GAMS results
    d. Back in the GUI, click **Run GAMS** to run the entire workflow. A plot will appear.
6. How does the plot change by entering different crop profitability values?

Helpful Materials

- GAMS Development Corp, GDXMRW – Interfacing GAMS and Matlab (http://tinyurl.com/z4cokvg)
- Additional Software & Tools for interfacing GAMS with Excel, R, and Python, etc. (https://www.gams.com/contrib/contrib.htm)
- Mathworks, Building a GUI with GUIDE in Matlab (http://tinyurl.com/keodevz)
- Alminagorta (2015), SWAMPS model - more sophisticated GUI for a nonlinear wetlands water allocation model (http://tinyurl.com/hf8syow)