

This zip file contains all the files necessary to compile and run the program for the Gustman-Steinmeier retirement model. The main program for the C++ code is GSModel.cpp. Associated files which should be in the same directory are GSModel.inc, winshell.h, winshell.ico, wishell.inc, winshell.rc, winshell64.h, and winshell64.inc. The files coh0015.95T, coh1655.95T, coh5690.95T, and SurvTabs.dat should be placed in the same directory as the executable file when the program is run. The program has been compiled and linked with Microsoft Visual Studio 9.0 and will run on 64 bit operating systems. It will run on 32 bit systems, but the number of threads may be limited.

The program has several modes determined by a variable in the main routine called mode. When mode is set to a value of zero, as it currently is, the model operates in test mode on a synthetic individual whose characteristics are given in the subroutine setData. To run the program, double click on the executable file, choose from the pull-down menu whether you want the output to go to the screen, file, or both, and whether you want to replace or append to the file, and choose start from the menu. When started, the program will ask for a range of thread numbers. The number of threads is given by nthreads in the main routine and should be equal to the lesser of the number of observations (one in the test case) or the number of cores in the machine (twice that number if the cores are multithreaded). For instance, if the machine has four multithreaded cores, nthreads should be set equal to 8, and the range of threads should be entered as 0 7. But in test mode the range should be entered as 0 0. If the program is run on multiple machines, the program should be started on the slave machines first and the range of threads on those machines will be something like 8 11, numbered consecutively after the range of threads on the master machine. Each slave machine should have a unique range of thread values. The master machine will always have a range of threads starting with 0. The output for the test individual is given in the file GSModel.out, which is a text file.

Mode 2 does an estimation (with mode 1 doing a single function evaluation) and mode 3 does a simulation. For actual data, the form of the input file is a binary file called GSRetirementData.dat. The first four bytes are an integer for the number of observations, and the rest of the file is that number of observations of a data structure called ORIGDATA. This data structure is found in the inpt routine. Note that the input file is not required for the test mode using synthetic data. The routine calcPrep calculates several values for the structure CALCSTRUCT which is used to pass values among routines. The routine rhocalc calculates a value of the time preference variable consistent with observed wealth for retirement at the observed or expected retirement age. The routine calcModel solves the model for the decision variables as a function of the state variables, and the routine getMoments does the simulations.