# Supplementary Materials for Replication: Worker Flows and Job Flows: A Quantitative Investigation

Shigeru Fujita[*] and Makoto Nakajima[†]

May 2016

## 1 Data

## 2 Codes

The followings, included in the folder `code-final`, are the codes and associated input files to replicate the results of the paper. The steady-state code (`jobwork-ss.f90`) has to be implemented first before implementing the code with aggregate shocks (`jobwork-ag.f90`), in order to obtain the stationary type distribution of firms, and a (steady-state) guess for the marginal value function. However, those outputs from the steady-state code are already included in this package, so there is no need to run the steady-state code first.

We implemented the codes using Intel Fortran Compiler for Linux (version 13.0.1) with MPI Library for Linux (version 4.1.0). Although the actual codes we used for the paper are parallel version with MPI (Message Passing Interface), we include the serial version of the same codes, which can be more easily and widely used. We made sure the two versions generate the same results. The operating system is CentOS 6, running as a part of Rocks Cluster Suite. All the seeds for the random number generator are specified in the codes. We also used -O3 optimization option when compiling the codes. Expected computational time for the serial steady-state code is about 20 minutes on 2.4GHz Intel CPU, using the correct guess for everything (prices, marginal value function, and type distribution). Expected computational time for the code with aggregate shocks is 5 hours with the same CPU. Using a parallel code with 128 nodes, computational time is less than 10 minutes.

1. `module`: This folder contains Fortran 90 modules called by the following codes.
2. `jobwork-ss.f90`: Fortran 90 program to compute a steady-state equilibrium. All parameters are set at the baseline values. The following input files can be used to speed up the convergence.

[*]Research Department, Federal Reserve Bank of Philadelphia. Ten Independence Mall, Philadelphia, PA 19106. E-mail: shigeru.fujita@phil.frb.org.

[†]Research Department, Federal Reserve Bank of Philadelphia. Ten Independence Mall, Philadelphia, PA 19106. E-mail: makoto.nakajima@phil.frb.org.

   (a) `jobwork-ss-in-fin.dat`: Input file containing a guess for the marginal value function in the steady state, $D(x, n')$. Use the option `finopt=1` in the code to use this input file.

   (b) `jobwork-ss-in-din.dat`: Input file containing a guess for the stationary type distribution of heterogeneous firms, $m(x, n)$. Use the option `dinopt=1` in the code to use this input file.

3. `jobwork-ag.f90`: Fortran 90 program to compute an equilibrium with aggregate uncertainty. All parameters are set at the baseline values. The following input files can be used to speed up the convergence.

   (a) `jobwork-ag-in-pin.dat`: Input file containing a guess for the forecasting functions of aggregate law of motion, $U' = \Phi_U(z, U)$ and $\theta = \Phi_\theta(z, U)$. Use the option `pinopt=2` in the code to use this input file.

   (b) `jobwork-ag-in-fin.dat`: Input file containing a guess for the marginal value function, $D(x, n', z, U')$. Use the option `finopt=2` in the code to use this input file.

   (c) `jobwork-ss-in-din.dat`: See above. This file is necessary to run the code.

4. In order to generate impulse response functions of the model, the same code as the one for the model with aggregate uncertainty (`jobwork-ag.f90`) is used but some options must be changed, as shown below. We show below how to construct the impulse response function with a negative one standard-deviation shock. You need all the input files as specified in 3 above.

   (a) `ecoopt=0` (since we are not solving for an equilibrium)

   (b) `zinopt=2` (with this option, the sequence for aggregate shocks is not generated by the random number generator but instead consistent with the impulse response function)

   (c) `impz=-0.003040_8` (size of the initial impulse, which is equal to (negative of) the size of the standard deviation of aggregate shocks in the current case)