**Read me file on the replication package for**

**“Foreign Shocks as Granular Fluctuations”**

**Julian di Giovanni, Andrei A. Levchenko, and Isabelle Mejean**

This file contains a detailed description of the code written to generate the analysis in the above-mentioned paper. A detailed description of the data is in the main text, Section II. The data are not included in this replication materials package because access to these data is restricted. To obtain the data, one must apply for access through the Comité du Secret Statistique of the Conseil national de l’information statistique (CNIS), the National Counsel for Statistical Information. More information on the application procedure can be found here: <https://cdap.casd.eu/>. In case the application is successful, we will be happy to transmit the entirety of the replication package through the server which the researchers are asked to use to access the data. All the code has been run on the CASD server configurated with the fourth package (6vCPU, 32 Go RAM).

**0. GENERAL ORGANIZATION**

The organization of folders for the Stata part of the code is defined in the Master program. The corresponding code must be saved in a folder later referred to as “$dopath”. The data are saved in a number of dedicated folders depending on the nature and origin of the data. Results are saved in $outputpath, $outputpathcsv, $outputpathpdfs, and $datacsvmatlab. $datacsvmatlab is also the “CSV” folder described below.

The organization of folders for the Matlab part of the code is the following. A master folder contains all the codes as well as four folders:

* CSV contains the .csv files exported from Stata
* MAT contains the .mat files produced by the code
* Tables contains the tables produced by the “master\_results.m” program.
* Figures contains the figures produced by the “master\_results.m” program.

The next two sections describe the structure of the code.

**I. STATA PROGRAMS**

**00\_Master.do**

* Master program for the Stata part of the empirical analysis.
* The first lines define a number of global, including the paths to the programs and data.
* Lines 71-78 sequentially run the do files that construct all datasets and run the regressions in Tables 1, A2 and A3.

1. **01\_0\_balancesheet\_customs.do**

This file uploads and cleans the balance sheet and customs data. The file itself calls three programs which are described next.

It runs the following files:

* 01\_1\_balancesheet.do

Input and clean balance-sheet data

01\_1\_1\_balancesheet\_cleaning.do

* Build the panel of balance-sheet data.
* Input files are brn1993.csv to brn2007.csv, which are the raw datafiles exported from SAS.
* Convert all data into euros
* Drop Finance, Insurance and Agriculture
* Harmonize sector codes over time
* Output is brn\_wof\_$firstyearbrn$lastyearbrn.dta

01\_1\_2\_balancesheet\_cleaning\_growth\_rates.do

* Rename the most important balance-sheet variables
* Create growth rates and dlog variables
* Keep observations with non-missing value added growth rates
* Keep firms with at least two growth rates
* Create a variable for the weight of the firm in aggregate value added (lagged)
* 01\_2\_0\_trade\_import.do
* Input and clean import data

01\_2\_1\_build\_panels.do

01\_2\_2\_insheet.do

* Insheet all customs files

01\_2\_3\_flux24Export.do

* Build the panel of export data
* Drop wrong ids, country codes and product codes
* Aggregate transactions by firm\*year(\*product)

01\_2\_3\_flux24Import.do

* Build the panel of Import data
* Drop wrong ids, country codes and product codes
* Collapse imports from Belgium and Luxembourg (the countries are associated with a single iso code until 1998)
* Aggregate transactions by firm\*year\*WIOT sector
* 01\_3\_0\_trade\_export.do

- Use data created in 01\_2\_3\_flux24Export.do

- Collapse exports to Belgium and Luxembourg (the countries are associated with a single iso code until 1998)

- Aggregate transactions by firm\*year

1. **02\_0\_Firm\_Level.do**

This file collects and organizes the relevant firm-level data for i) the motivation regression and ii) the calibration of the model.

It uses as input the panels created in 01\_0\_balancesheet\_customs.do

It runs the following steps:

- Bin firms in balance-sheet data by WIOT sectors based on their NAF sector (00\_1\_apetoWIOT12.do)

- Aggregate import and export data for countries outside of the WIOD dataset into a “Rest of the World”

- Add total imports and its growth rate to balance-sheet data

- Keep the lag of all balance-sheet data (sales, employment, wage bill, etc)

- Transform the long dataset of firm\*destination export data into a wide form to be merged with balance-sheet data.

1. **03\_0\_WIOT.do**

Input and organize the WIOD data. In particular:

- Creates the matrix of technical coefficients at the country\*sector level

- Adjust the technical coefficients to fit with our assumption regarding tradable / non-tradable goods.

- Compute the labor shares as the ratio of value added over gross output (our model has one factor which is “equipped” labor)

1. **04\_0\_Calibration.do**

Merge the firm-level and WIOD data and calculate the three sets of parameters necessary to calibrate firm-level production functions.

This is done for each separate year but we then use a single cross-section

The following steps are implemented:

* Reshape import data into a wide form. An observation is now a firm(\*year) and sector\*country of origin pairs are in columns
* Compute the labor share at the firm level (value added over total sales) and rescale to fit to the WIOD labor shares at the sectoral level. The rescaling factor is computed vis-à-vis a sales-weighted average of firm-level labor shares.
* Compute intermediate consumption at the firm level (input purchases on raw material and merchandises) as well as the share of foreign inputs (value of imports over intermediate consumption)
* Compute technical coefficients at firm level using information on bilateral imports (by sector\*origin) and intermediate consumption. Technical coefficients are then rescaled so as to match the sector-level values from WIOD.
* Compute sales shares at the firm-level using total sales and bilateral exports.

1. **05\_0\_VA\_GDP\_Trade.do**

Finish preparing the panel for the regressions in Tables 1, A2 and A3.

The following steps are implemented:

* Import data on nominal and real GDP per country.
* Merge with balance sheet
* Use GDP deflator for France to transform nominal variables into real variables
* Merge balance sheet with import and export data (in their wide format)
* Create dummies for exporting to or importing from any country as well as measure of export/import intensities

1. **06\_0\_Export\_CSV.do**

Export data for baseline year into CSV (to be later used in Matlab).

1. **07\_0\_Lifi.do**

Import the LiFi Survey on the location of firms’ headquarters / foreign affiliates (used in 08\_0\_Motivational\_Regressions.do)

1. **08\_0\_Motivational\_Regressions.do**

Run the regressions for Tables 1, A2 and A3.

The following steps are implemented:

* Use the balance-sheet panel and merge the additional data (GDPs, Lifi)
* Construct the control variables
* Identify extreme values for all the variables used as left-hand side variable (ie VA growth rates as well as the growth rate of the wage bill, intermediate consumption, imports)
* Run the regressions in Tables 1, A2 and A3
* Store results

**II. MATLAB PROGRAMS**

**Master\_run.m**

This master Matlab program runs the various version of the model used in the paper.

For the baseline calibration as for each robustness, the structure is the same:

* Define the parameters
* Run the “master\_baseline\_CES\_approx” program which solves the model using the baseline calibration and computes the wedges needed to clear all product markets
* Run the “master\_counterfactuals\_CES\_approx” program which simulates the various shocks and stores the results
* Run the “master\_counterfactuals\_CES\_Homogeneous\_approx” program which simulates the same shocks in the homogeneous version of the model

Specific robustness exercises:

* In “master\_counterfactuals\_CES\_approx\_countryloop”, the counterfactuals are simulated country by country.
* “master\_counterfactuals\_CES\_idiosyncraticprodshock\_byfirm” simulates shocks to various percentiles of the productivity distribution to compute the vector of influences in Figure 8
* “master\_counterfactuals\_CES\_approx\_Olig” is the version of the model with endogenous markups
* “master\_counterfactuals\_CES\_approx\_Profit” corresponds to the version in which the contribution of profits to aggregate demand is affected by the shock

**The “baseline” program is organized as follows**:

1. Step\_1\_Import\_WIOD\_CES.m (baseline only)

This program prepares the sectoral data. It imports the WIOD and organize them by country and sector as in the paper.

1. Step\_2\_Real\_Firms\_CES.m (baseline only)

This program imports the firm-level data and organizes the information to match the WIOD ordering.

1. Step\_3\_Market\_Clearing\_Expenditure\_CES.m (baseline only)

This program finds the market-clearing wedges.

1. Step\_4\_Algorithm\_baseline\_CES\_approx.m

This program implements the convergence algorithm using hat algebra. The algorithm itself involves four steps that are run within the EquilibriumCES\_fun\_approx.m function.

At the end of the program, the wedges are saved and all variables of the baseline calibration are updated so as to fit with the wedges.

**The “counterfactuals” and “counterfactuals\_Homogeneous” versions** display the same structure, the only difference being that “counterfactuals\_Homogeneous” has the calibration of the homogeneous model (starting from the parameters in the heterogeneous model):

* Load the data
* Run a loop on the different shocks: Productivity or preference shock
* Define the shock
* Run the algorithm for solving the model under the counterfactual: Step\_4\_Algorithm\_counterfactual\_CES\_approx.m. The algorithm itself involves four steps that are run within the EquilibriumCES\_fun\_approx.m function.
* Run Step\_5\_counterfactual\_output\_CES\_approx.m which produces the output of the algorithm we need to store. This program notably computes the various deflators used in the analysis (DoubleDeflation\_deflatorCES\_fun\_approx.m and GDP\_deflatorCES\_fun\_approx.m)
* Run Step\_6\_Elasticity\_output\_CES.m which further computes various elasticities

**The structure of the EquilibriumCES\_fun\_approx.m algorithm** is as follows:

* PriceLoopCES\_fun\_approx.m takes wage and price guesses and loops until prices converge
* TradeShareCES\_fun.m given the new prices, adjust the trade and labor shares for the next period
* ExpenditureCES\_fun.m creates a guess for the change in expenditures and loop until convergence
* LabourMarketConditionCES\_fun.m uses the labor market condition to update the wage guess
* Iterate until wages no longer change

**Master\_results.m**

Master file for all the results in the paper. Each table and figure are associated with a program which name corresponds to the labels in the paper.