Problem Set 5: Computing the Inventory Model in Alessandria et al (2010, AER)

Armen Khederlarian ECO862

Consider the inventory model seen in class in Alessandria, Kaboski and Midrigan (2010).

1. Download the code and understand the basics of quantitatively solving such models. Replicate the model figures in the paper (code description is added below).
   In the original model the idiosyncratic demand $\nu_{j,t}$ was *iid* across time and firms. Now assume that $\nu_{j,t}$ is follows and AR(1) process of the form:

   $$\nu_{j,t} = \rho \nu_{j,t} + \epsilon_{j,t}$$

   where $\epsilon \sim N(0, \sigma_\epsilon^2)$.

2. Modify the code so that it solves the model with $\nu_{j,t}$ following an AR(1) process. Choose σ_ϵ^2 such that Tauchen method generate the same grid and variance of shocks in AKM(2010) with *iid* shocks.

3. Replicate the ordering and price policy functions in Figures 2 and 3 of Alessandria, Kaboski and Midrigan (2010), in the *iid* and AR(1) process for the median demand shock. Compare the two policy functions. How are they different? Explain.

4. Go back to the benchmark model and now modify the code to consider two alternative formulations of depreciation rate: (1) goods don't depreciate in transit; and (2) the depreciation is convex function of the inventory level i.e. $\delta = \delta_0 \exp(\delta_1 s)$, set $\delta_0 = 0.01$ and δ_1 such that $\delta = 0.05$ if $s = \overline{S}$ in the benchmark case (with linear depreciation and $\delta = 0.025$). Again plot and compare the ordering and price policy functions.

Code Description: Table 5: Moments and Parameters, Figure 2 - Optimal Order Rule; Figure 3 - Optimal Price Rule; Figure 5a & 5b: Ergodic distribution before and after crisis; Transition dynamics (Figure 6; Figure 7 (anticipate drop in C prior to devaluation), Figure 8 (gradual devaluation), Figure 9 (Imports by date of Last pre-devaluation import); Figure 10 Effect of Depreciation and Inventory Holdings on Passthrough

- The codes for the simulations can be found in the attached folder
- Each folder contains files that perform computations described in the paper.
- The "Benchmark" folder contains the main (Benchmark) experiments, the rest perform additional robustness experiments.
- The structure of the "Benchmark" folder is described below, the other folders are similar.

  1. start.m computes the policy rules/value functions of firms in the Benchmark economy and reports moments. It also computes the expected value of a firm that is subject to the

frictions. This is used to compute the tariff-equivalent of the frictions. To compute the tariff-equivalent one must run start.m in the "Frictionless" folder which computes the value of a firm in a frictionless economy in which the tariff on importers is equal to tau. One then updates τ iteratively to equate the two valuations.

2. start.m calls simulate.m and statistics.m which simulate firm decision rule and compute statistics, respectively.

3. The code also calls Miranda and Fackler's Compecon library which must be installed on the matlab path before running start.m

4. start_calm.m computes the transition in response to a "devaluation". It first computes decision rules, then (in transition.m) computes the ergodic distribution prior to the devaluation, saves this data (in calm.mat) and then calls start_crisis.m which computes the post-crisis optimal decision rules. One can edit the parameters determining the magnitude of the devaluation, the behavior of the interest rate etc. in start_crisis.m. Finally, transition.m uses several auxiliary files (hist.m, hist2.m) in order to compute the ergodic distribution.

5. All other folders are identical. The only exception occurs in the "Gradual Devaluation anticipated" experiment which feeds a particular path for ω_t that is known at the moment of the devaluation. The code therefore no longer assumes that the firms use the post-crisis steady-state decision rules and use instead a shooting algorithm to solve the problem. That is, it guesses that the economy converges in K periods, and use backward iteration to compute value functions at each date during the transition. This is accomplished in start_crisis.m at lines 130-135.