Heterogeneous Agents in Macro Models

Solving Hetero-Models *With* Aggregate Uncertainty
Alternatives & Extensions

Petr Sedláček

Lake Como School of Advanced Studies
July 2018

Plan overall

Basic tools for solving heterogeneous agent models

1. motivation
2. refresher: solution methods for rep agent models
3. Aiyagari model: hetero-model without aggregate uncertainty
4. Krusell-Smith algorithm: introducing aggregate uncertainty
5. alternative solution methods: XPA, hybrid, S-S
6. continuous time: basic techniques

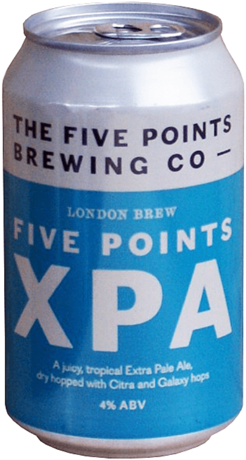
Plan for today

Alternatives and extensions to Krusell-Smith algorithm

- explicit aggregation
- hybrid method (projection+perturbation)
- solving models with ex-ante heterogeneous agents

Explicit Aggregation

Explicit Aggregation (XPA)



Main idea

Use version of Ayiagari model with employed and unemployed

- $e_i = e_e = 1$ if employed and $e_i = e_u = 0$ if unemployed

Guess functional form of individual policy rules (e.g. linear, quadratic)

- aggregate individual decisions rules
- check implied aggregate law of motion
- and whether more terms need to be added

Guess for individual policy rules

Suppose policy rules for (un-)employed are given by

$$k'_u = \Psi_{0,u}(\mathbf{S}) + \Psi_{1,u}(\mathbf{S})k \quad \text{when } \mathrm{e}_i = \mathrm{e}_u = 0$$

$$k'_e = \Psi_{0,e}(\mathbf{S}) + \Psi_{1,e}(\mathbf{S})k \quad \text{when } \mathrm{e}_i = \mathrm{e}_e = 0$$

- **S** is aggregate state (possibly including distribution)
- policy rules assumed linear in k, but general otherwise

So what?

If we can aggregate and compute K', then

- conditional on Z', we can compute r and w
- and thus solve individual problem

Notice that we're not talking about simulation anywhere

- $\rightarrow$ explicit aggregation!

Some notation

$\widetilde{K}_i$: end of period aggregate capital of $i = u, e$

K_i': beginning of period aggregate capital of $i = u, e$

$\widetilde{K}_i \neq K_i'$! Why?

Aggregation

Compute end of period capital holdings as

$$\widetilde{K}_i = \widetilde{M}_i(1) = \int k_i'(k, \mathbf{S}) dF(k) = \Psi_{0,i}(\mathbf{S}) + \Psi_{1,i}(\mathbf{S}) M_i(1)$$

- $M_i(j)$ is the j-th moment of capital distribution
- entire distribution is taken into account
- policy rules can be complex, non-linear in aggregate states!

Aggregation cont.

But we need to compute K' in order to get prices

$$K'_u = \frac{\omega_{euZZ'}\widetilde{K}_e + \omega_{uuZZ'}\widetilde{K}_u}{\omega_{euZZ'} + \omega_{uuZZ'}}$$

$$K'_e = \frac{\omega_{eeZZ'}\widetilde{K}_e + \omega_{ueZZ'}\widetilde{K}_u}{\omega_{eeZZ'} + \omega_{ueZZ'}}$$

$$K' = ur'K'_u + (1 - ur')K'_e$$

- $\omega_{ijZZ'}$ are transition probabilities between $i = u, e$ and $j = u, e$
- ur is the unemployment rate

So what?

What have we done?

1. assumed a functional form for policy rules
2. implies a particular law of motion for aggregates
3. can use that l.o.m. to compute prices
4. $\rightarrow$ can compute individual policy rules
    - we'll talk about how later on

But first, what if policy rules are quadratic?

Policy rules

$$k_i' = \Psi_{0,i}(\mathbf{S}) + \sum_{j=1}^{2} \Psi_{j,i}(\mathbf{S}) k^j \quad \text{for i=u,e}$$

Aggregation

$$\widetilde{K}_i = \widetilde{M}_i(1) = \int k_i'(k, \mathbf{S}) dF(k) = \Psi_{0,i}(\mathbf{S}) + \sum_{j=1}^{2} \Psi_{j,i}(\mathbf{S}) M_i(j)$$

What's the problem?

Aggregation of quadratic policy rules

We need laws of motion for aggregate moments

$$\widetilde{M}_i(2) = \int k_i'(k, \mathbf{S})^2 dF(k)$$

Fine, but what's the problem?

- the above includes cubic and quadratic terms of k
- would need l.o.m. for those aggregates too
- but those would include even higher-order terms etc!

Way out of infinite regress?

Come up with different *approximation* of $\widetilde{M}_i(2)$

$$\widetilde{M}_i(2) = \Psi_{0,M(2),i}(\mathbf{S}) + \sum_{j=1}^{2} \Psi_{j,M(2),i}(\mathbf{S}) M_i(j)$$

Importantly

- no direct relationship between coefficients
  - $\Psi_{0,i}(\mathbf{S})$ and $\Psi_{0,M(2),i}(\mathbf{S})$ or $\Psi_{j,i}(\mathbf{S})$ and $\Psi_{j,M(2),i}(\mathbf{S})$!

Summary of quadratic policy rule setup

$$k_i' = \Psi_{0,i}(\mathbf{S}) + \sum_{j=1}^{2} \Psi_{j,i}(\mathbf{S}) k^j \quad \text{for i=u,e}$$

$$\widetilde{K}_i = \widetilde{M}_i(1) = \int k_i'(k, \mathbf{S}) dF(k) = \Psi_{0,i}(\mathbf{S}) + \sum_{j=1}^{2} \Psi_{j,i}(\mathbf{S}) M_i(j)$$

$$\widetilde{M}_i(2) = \Psi_{0,M(2),i}(\mathbf{S}) + \sum_{j=1}^{2} \Psi_{j,M(2),i}(\mathbf{S}) M_i(j)$$

Implementing XPA

How to find coefficients of policy rules?

1. guess aggregate law of motion
   - i.e. make stand on functional form of individual policy rules
   - gives law of motion for prices
2. solve individual problem in accordance to 1
3. explicitly aggregate individual decisions
4. compare initial guess with resulting aggregate law of motion
5. update and go back to 2

Interim summary

Explicit aggregation

- realize that specific functional forms of policy rules
- aggregate up in particular ways
  - can track aggregate laws of motion directly
  - while keeping lots of non-linearity in certain dimensions

Hybrid method

Possible issue with Krusell-Smith

Aggregate uncertainty?

- typically relatively small
- will not move individuals far from steady state
- $\rightarrow$ perturbation likely to work relatively well

Idiosyncratic uncertainty?

- typically relatively large
- likely to move individuals far from steady state
- $\rightarrow$ perturbation unlikely to work well!

Way out?

Krusell-Smith setup of individual problem

$$k_{i,t+1} = a_0 + a_1(k_{i,t} - \overline{k}) + a_2(e_{i,t} - \overline{e}) + a_3(Z_t - \overline{Z}) + a_4(K_t - \overline{K})$$

Replace above with projection?

$$k_{i,t+1} = P_n(k_{i,t}, e_{i,t}, Z_t, \mathbf{m}_t; \lambda_k)$$

In best-case scenario we have

- 4 state variables
- however, in many applications
  - there are more than 2 individual states
  - first moment of distribution is not enough
- moreover, above formulation also somewhat restrictive
  - how?

Beautiful insight in Reiter (2008)

Because aggregate shocks are relatively small

- combine perturbation solution to aggregate dynamics
- with non-linear solution of individual problem

In other words

- solve non-linearly around "stationary" steady state
  - i.e. $\sigma_Z = 0$, but $\sigma_e > 0$
- perturb entire system around this stationary equilibrium

How is this going to help?

1. Magically reduce curse of dimensionality of projection solution
   - how?!
2. Make the non-linear solution more general
   - how?!

Details

The key insight

$$k_{i,t+1} = P_n(k_{i,t}, e_{i,t}, Z_t, \mathbf{m}_t; \lambda_k)$$

Details

The key insight

$$k_{i,t+1} = P_n(k_{i,t}, e_{i,t}; \lambda_{k,t})$$

With

$$\lambda_{k,t} = \lambda_k(Z_t, \mathbf{m}_t)$$

Intuition

A brute force extension

- include aggregate shocks
- and moments of distribution as state variables
- $\rightarrow$ policy rules affected
  - in possibly non-linear ways
  - but coefficients on individual states independent of aggregates

Reiter's hybrid method

- make coefficients on individual states vary with aggregates!
- allows individual policy rules to drop aggregates as states!
  - can track aggregate distribution in spectacular detail!
  - while keeping non-linear approximation of individual problem!

Implementation

1. solve individual problem non-linearly in stationary equilibrium
   - how?
2. aggregate individual decisions
   - how?
3. perturb entire system, including parameters of policy rules
   - how?

Implementation: 1. non-linear approximation

This is the easy bit, we already know how

- use projection to non-linearly approximate individual problem

$$k_{i,t+1} = P_n(k_{i,t}, e_{i,t}; \lambda_{k,t})$$

But how to make coefficients time-varying?

- worry about that in a few steps
- for now, as if solving model *without* aggregate uncertainty

Implementation: 2. aggregate individual decisions

This is also relatively easy, we already know how

- $\rightarrow$ similar to non-stochastic simulation
  - given a grid for state variables $[k_j, e_j]$
  - and given decision rules $k = P_n(k_j, e_j; \lambda_{k,t})$
  - $\rightarrow$ can compute mass of agents moving across grid points
  - $\rightarrow$ aggregate simply by summing over grid points

Implementation: 3. perturb entire system

This is the last tricky bit, but we also know how

Recap of setup

1. know how to perturb rep-agent model (i.e. with only aggregates)
   - know how to get aggregates given distribution!
2. can perturb distribution in exactly the same way
   - know how to get distribution given individual choices!
3. what are the "variables" at individual level?
   - not $k_{i,t}$ or $c_{i,t}$! But $\lambda_{k,t}$ and $\lambda_{c,t}$!
   - what are the respective "steady states"?

Implementation: 3. perturb entire system cont.

But how to "perturb" coefficients of policy approximations?

- e.g. imagine $P_n(k_{i,t}, e_{i,t}, \lambda_{k,t})$ is a 2nd order polynomial
- $\rightarrow 6$ coefficients to perturb
- what are the "equations" determining these coefficients?

$$\frac{1}{r_t k_j - w_t e_j - P_n(k_j, e_j; \lambda_{k,t})} =$$
$$\beta \mathbb{E}\left[\frac{r_{t+1}}{r_{t+1} P_n(k_j, e_j; \lambda_{k,t}) + w_{t+1} e_{+1} - P_n(k_j, e_j; \lambda_{k,t})}\right]$$

- $\rightarrow$ 6 grid points are enough to conduct all this!
- what if we have more grid points?

Interim summary

Reiter's Hybrid method

- combines the benefits of both worlds
  - non-linear solution to individual problem
  - aggregate shocks with very accurate tracking of distributions
- relatively easy to implement (can use Dynare for parts)

Solving models with ex-ante heterogeneity

Main idea

Krusell-Smith algorithm is computationally costly

- simulation step takes time
- iterative updating takes time
- accuracy checks and moment selection take time

Why do we need to do all the above?

- need to know the joint distribution of capital and productivity
  - distribution of infinitely-lived agents
  - hit by persistent idiosyncratic shocks
- can we get around the above?

Basic idea

Agent heterogeneity?

- what if we had (many) ex-ante heterogeneous agents (I)?

Infinitely-lived agents?

- what if we consider an OLG framework instead (T)?

What have these chagnes given us?

- ex-ante heterogeneity: finite number of types of agents
- finite life-time: finite number of periods to track each type
- $\rightarrow T \times I$ types of agents

Basic idea

This is great, because $T \times I$ is finite!

- moreover, the $T \times I$ types describe the *entire* distribution!

What about the curse of dimensionality?

- $\rightarrow$ do perturbation

No need for iterative procedure

- entire distribution described by transitions between types
  - i.e. aging of agents

Details

Specific application: firm dynamics model

- there are I *types* of firms
  - productivity/demand heterogeneity (different long-run sizes)
- firms live for (at most) A periods
  - life-cycle heterogeneity within firm types

Details cont.

Each age-type has a different value function!

$$V_{i,a}(S_t) = \pi_{i,a}(S_t) + \beta(1-\delta_a)\mathbb{E}V_{i,a+1}(S_{t+1})$$

With

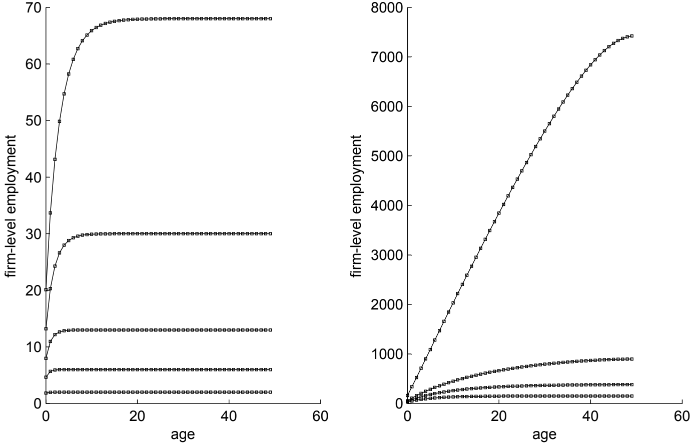
- δ is age-specific death rate
- S_t being the aggregate state
  - aggregate shocks, but also entire distribution of firms!

Keeping track of entire distribution is easy!

$$\omega_{i,a} = (1-\delta)\omega_{i,a-1} \quad a \in (1, A]$$

$$\omega_{i,0} = \text{free entry condition}$$

Resulting economy has lots of heterogeneity



Source: Sedláček and Sterk (2016)

Resulting economy has lots of heterogeneity

Table: Employment share distribution by size and age

	data			model		
	small	medium	large	small	medium	large
0 to 5 years	50	41	8	52	40	8
6 to 10 years	36	46	18	36	48	16
11 to 15 years	29	45	26	27	46	27
16 to 20 years	24	42	34	22	44	34
21 to 25 years	18	39	43	18	39	43

Notes: Employment shares in percentages of small (1-19 employees), medium-sized (20-499) and large (500 and over) firms, by age. Data (averages) and models (steady states).

Quantitative implementation

- results in more than 900 state variables
    - include the entire joint distribution of
    - firm employment and their masses
- solved with first-order perturbation
    - along steady state growth path
- solution takes several seconds
    - fast enough that we can estimate parts of the model

Implementation in Dynare

Macro-language in Dynare

- can define loops over certain criteria (e.g. types)
- ideal for heterogeneous-agent setups
  - structure of first-order conditions the same across types
  - they differ in (some) parameter values
- other examples of its use include
  - multi-country models

Interim summary

Ex-ante heterogeneity and finite lives

- replace ex-post heterogeneity with ex-ante (permanent)
- cut distribution (where it doesn't matter quantitatively)
  - can keep track of entire distribution
  - $\rightarrow$ many state-variables, but manageable with perturbation!

Summary

What did we do?

Alternatives and extensions to Krusell-Smith algorithm

- hybrid method
- explicit aggregation
- Sedláček, Sterk (2017)

What's next?

Continuous time

- basic setup and tools