Solving and Estimating Heterogeneous Agent Models with Aggregate Uncertainty by Perturbation Methods

Short Course CEMFI

R. Luetticke
UCL, CEPR & CfM

March 2021

Heterogeneous agents models with aggregate uncertainty

These models are computational demanding to solve

- The original Krusell and Smith (1997, 1998) algorithm is notoriously slow
- Therefore, many papers study transitions
- or are restricted to relatively simple household decisions

Heterogeneous agents models with aggregate uncertainty

These models are computational demanding to solve

- The original Krusell and Smith (1997, 1998) algorithm is notoriously slow
- Therefore, many papers study transitions
- or are restricted to relatively simple household decisions

- We depart from the Reiter (2002, 2009) perturbation method
- And (try to) provide an accessible algorithm that can deal with high-dimensional heterogeneity

Reiter (2002): Solve by perturbation

- Models can be written as a non-linear difference equation:
  $\mathbb{E}F(X_t, X_{t+1}, Y_t, Y_{t+1}, \varepsilon_{t+1}) = 0$

  The heterogeneous agent model:
- that is function valued and
- needs to be linearized around the stationary equilibrium (StE)

Reiter (2002): Solve by perturbation

- Models can be written as a non-linear difference equation:
  $\mathbb{E}F(X_t, X_{t+1}, Y_t, Y_{t+1}, \varepsilon_{t+1}) = 0$

  The heterogeneous agent model:
- that is function valued and
- needs to be linearized around the stationary equilibrium (StE)
- Functions need to be approximated by finite dimensional objects (e.g. coefficients of polynomials, splines, etc.)
- We show how to do this in a smart efficient way

Course outline

Basics

- Standard incomplete markets model
- Perturbation approach

Solving SIM by perturbation

- Our reduction method
- Application: Solving the Krusell&Smith model
- Comparison to MIT shock solution

HANK models

- Application: Estimation of HANK models

Resources

- Lecture slides
- Coding exercises
  - I provide templates for their solution in MATLAB or Julia.

Heterogeneous agents models with aggregate uncertainty

Available codes:

- Perturbation vs MIT shock for KS model (Matlab)
  https://github.com/ralphluet/KS_Perturbation_vs_MIT
- Perturbation with our reduction for KS and HANK models (Matlab)
  https://github.com/ralphluet/perturbation_codes
- Perturbation with our reduction for HANK models (Python)
  https://github.com/econ-ark/BayerLuetticke
- Perturbation with our reduction for estimating HANK models (Julia)
  https://github.com/BenjaminBorn/HANK_BusinessCycleAndInequality

Literature - Academic articles

Perturbation:

- Reiter (2002, 2009), Ahn et al. (2018), Bayer and Luetticke (2020), and Bayer et al. (2019)
- ...

MIT shock:

- Boppart et al. (2018) and Auclert et al. (2019)
- ...

Global:

- Carroll (2006) and Hintermaier and Koeniger (2010)
- ...

Literature - Textbooks

- Heer, B. and A. Maussner (2009), "Dynamic General Equilibrium Modelling", 2nd edition, Springer, Berlin.
  Getting started: Ch. 1, 4, 7, 8
- Adda, J. and R. Cooper (2004): "Dynamic Economics", MIT Press, Cambridge.
  Partial Equilibrium only, Many Applications with Non-Convex Budget Sets.
- Ljungqvist, L. und T. Sargent (2012): "Recursive Macroeconomic Theory", 3rd ed., MIT press, Cambridge.
  Economic Theory Background
- Stockey, N.L. and Lucas, R.E. with E.C. Prescott (1989): "Recursive Methods in Economic Dynamics", Chapters 4 and 9, Havard University Press, Cambridge.
  Mathematical Background to Dynamic Programming

- An excellent quantitative econ source (for Python and Julia though): lectures.quantecon.org by Tom Sargent et al.

SIM
Model Setup

Recursive Dynamic Planning Problem

Consider a household problem in presence of aggregate and idiosyncratic risk

- S_t is an (exogenous) aggregate state
- s_{it} is a partly endogenous idiosyncratic state
- μ_t is the distribution over s

- Bellman equation:

$$v(s_{it}, S_t, \mu_t) = \max_{x \in \Gamma(s_{it}, P_t)} u(s_{it}, x) + \beta \mathbb{E} v(s_{it+1}(x, s_{it}), S_{t+1}, \mu_{t+1})$$

Recursive Dynamic Planning Problem

Consider a household problem in presence of aggregate and idiosyncratic risk

- S_t is an (exogenous) aggregate state
- s_{it} is a partly endogenous idiosyncratic state
- μ_t is the distribution over s

- Euler equation:

$$u'[x(s_{it}, S_t, \mu_t)] = \beta R(S_t, \mu_t)\mathbb{E}u'[x(s_{it+1}, S_{t+1}, \mu_{t+1})],$$

No aggregate risk

Recall how to solve for a stationary equlibrium:

- Discretize the state space (vectorized)
- Optimal policy $\bar{h}(s_{it};P)$ induces flow utility $\bar{u}_{\bar{h}}$ and transition probability matrix $\Pi_{\bar{h}}$

No aggregate risk

- Discretized Bellman equation

$$\bar{v} = \bar{u}_{\bar{h}} + \beta \Pi_{\bar{h}} \bar{v} \tag{1}$$

holds for optimal policy (assuming a linear interpolant for the continuation value)

- and for the law of motion for the distribution (histograms)

$$d\bar{\mu} = d\bar{\mu} \Pi_{\bar{h}} \tag{2}$$

No aggregate risk

Equilibrium requires

- $\bar{h}$ is the optimal policy given P and v (being a linear interpolant)
- $\bar{v}$ and $d\bar{\mu}$ solve (1) and (2)
- Markets clear (some joint requirement on $\bar{h}, \mu, P$, denoted as $\Phi(\bar{h}, \mu, P) = 0$)

This can be solved for efficiently

- $d\bar{\mu}$ is vector corresponding to the unit-eigenvalue of $\Pi_{\bar{h}}$
- Using fast solution techniques for the DP, e.g. EGM
- Using a root-finder to solve for P

Equilibrium

- Market clearing depends on the specific model

Equilibrium

- Market clearing depends on the specific model
- In an Aiyagari model, we require that

$$r + \delta = F_K(K, L), w = F_L(K, L)$$

where in the most simple case aggregate labor supply is exogenously given. Then, prices are only a function of K and the equilibrium condition is simply

$$K^S(P(K)) = K$$

Equilibrium

- Market clearing depends on the specific model
- In an Aiyagari model, we require that

$$r + \delta = F_K(K, L), w = F_L(K, L)$$

where in the most simple case aggregate labor supply is exogenously given. Then, prices are only a function of K and the equilibrium condition is simply

$$K^S(P(K)) = K$$

- In a Huggett model, aggregate bond supply is zero and

$$K^S(r) = 0$$

is the equilibrium condition.

Computer Exercise 1

Aiyagari model

Exercise

Solve the Aiyagari model as spelled out in Bayer and Luetticke (2020). The production function is

$$F(K,N) = K^{\alpha}N^{1-\alpha},$$

where $N = n_i$ *is common across households (GHH preferences)*

Solve the steady state using the EGM and Young's method.

Excursus: Endogenous Grid Method

- Most solution methods involve root-finding.

Excursus: Endogenous Grid Method

- Most solution methods involve root-finding.
- This is numerically intense.

Excursus: Endogenous Grid Method

- Most solution methods involve root-finding.
- This is numerically intense.
- Carroll (2006) proposes a method to solve dynamic optimization problems without relying on root-solving.

Excursus: Endogenous Grid Method

- Most solution methods involve root-finding.
- This is numerically intense.

- Carroll (2006) proposes a method to solve dynamic optimization problems without relying on root-solving.
- This method makes the grid and not the policy "endogenous".

Endogenous Grid Method

- The idea is akin to forward induction: "Given the state I am in now, what must have been the state I was in before, if I behaved rational".

Endogenous Grid Method

- The idea is akin to forward induction: "Given the state I am in now, what must have been the state I was in before, if I behaved rational".
- The method deals with occasionally binding constraints, but requires

Endogenous Grid Method

- The idea is akin to forward induction: "Given the state I am in now, what must have been the state I was in before, if I behaved rational".
- The method deals with occasionally binding constraints, but requires
  - differentiability of the value function everywhere,

Endogenous Grid Method

- The idea is akin to forward induction: "Given the state I am in now, what must have been the state I was in before, if I behaved rational".
- The method deals with occasionally binding constraints, but requires
  - differentiability of the value function everywhere,
  - (in its basic form) first order conditions that are sufficient, and

Endogenous Grid Method

- The idea is akin to forward induction: "Given the state I am in now, what must have been the state I was in before, if I behaved rational".
- The method deals with occasionally binding constraints, but requires
  - differentiability of the value function everywhere,
  - (in its basic form) first order conditions that are sufficient, and
  - monotone policy function (isomorphisms)

Endogenous Grid Method

- Consider the dynamic programming problem

$$V(s,\xi) = \max_{s' \geq b} u\left((1+r)s - s' + \xi\right) + \beta E V\left(s', \xi'\right).$$

Endogenous Grid Method

- Consider the dynamic programming problem

$$V(s, \xi) = \max_{s' \geq b} u\left((1+r)s - s' + \xi\right) + \beta E V\left(s', \xi'\right).$$

- The first-order condition

$$u'\left((1+r)s - s' + \xi\right) = (1+r)\beta E u'((1+r)s' - s'' + \xi') + \lambda \tag{3}$$

characterizes the optimal solution. Where $\lambda = 0$ if $s' > b$

Endogenous Grid Method

The algorithm works iteratively and solves for the policy function directly:

1. Start with a policy guess (for simplicity for consumption).

Endogenous Grid Method

The algorithm works iteratively and solves for the policy function directly:

1. Start with a policy guess (for simplicity for consumption).
2. Thereby, obtain values for the RHS of (3) for each grid point.

Endogenous Grid Method

The algorithm works iteratively and solves for the policy function directly:

1. Start with a policy guess (for simplicity for consumption).
2. Thereby, obtain values for the RHS of (3) for each grid point.
3. Then solve for the current state s such that (3) holds for each s'.

Endogenous Grid Method

The algorithm works iteratively and solves for the policy function directly:

1. Start with a policy guess (for simplicity for consumption).
2. Thereby, obtain values for the RHS of (3) for each grid point.
3. Then solve for the current state s such that (3) holds for each s'.
4. This yields a set of inner solution pairs $\{s', (s, \xi)\}$ to be inverted.

Endogenous Grid Method

The algorithm works iteratively and solves for the policy function directly:

1. Start with a policy guess (for simplicity for consumption).
2. Thereby, obtain values for the RHS of (3) for each grid point.
3. Then solve for the current state s such that (3) holds for each s'.
4. This yields a set of inner solution pairs $\{s', (s, \xi)\}$ to be inverted.
5. It also identifies the current states s for which the constraint binds.

Endogenous Grid Method

The algorithm works iteratively and solves for the policy function directly:

1. Start with a policy guess (for simplicity for consumption).
2. Thereby, obtain values for the RHS of (3) for each grid point.
3. Then solve for the current state s such that (3) holds for each s'.
4. This yields a set of inner solution pairs $\{s', (s, \xi)\}$ to be inverted.
5. It also identifies the current states s for which the constraint binds.
6. Go back to 2. and iterate until convergence.

Endogenous Grid Method

- Say $c^{(n)}$ is the policy function in iteration n. Then, we can be calculate the necessary assets s as

$$
\begin{aligned}
(1+r)s^*(s',\xi) &= s' - \xi + c^*(s',\xi) \\
c^*(s',\xi) &= u'^{-1}\left\{(1+r)\beta E_{\xi} u'\left[c^{(n)}(s',\xi')\right]\right\}
\end{aligned}
$$

Endogenous Grid Method

- Say $c^{(n)}$ is the policy function in iteration n. Then, we can be calculate the necessary assets s as

$$\begin{aligned}(1+r)s^*(s',\xi) &= s' - \xi + c^*(s',\xi)\\ c^*(s',\xi) &= u'^{-1}\left\{(1+r)\beta E_{\xi}u'\left[c^{(n)}(s',\xi')\right]\right\}\end{aligned}$$

- For a given grid of points s', s^* is typically off-grid!

Endogenous Grid Method

- Say $c^{(n)}$ is the policy function in iteration n. Then, we can be calculate the necessary assets s as

$$\begin{aligned}(1+r)s^*(s',\xi) &= s' - \xi + c^*(s',\xi) \\ c^*(s',\xi) &= u'^{-1}\left\{(1+r)\beta E_{\xi} u'\left[c^{(n)}(s',\xi')\right]\right\}\end{aligned}$$

- For a given grid of points s', s^* is typically off-grid!
- However, we have solved a policy function for some asset levels:

$$(s^*,\xi) \rightarrow c^*(s',\xi)$$

Policy functions on a fixed grid

- For well behaved felicity functions u, s^* and c^* are monotone in s'. This allows us to move to the next iteration making use of the following two implications:

Policy functions on a fixed grid

- For well behaved felicity functions u, s^* and c^* are monotone in s'. This allows us to move to the next iteration making use of the following two implications:
  - For a given ξ we can obtain consumption policies $c^{(n+1)}(s,\xi)$ by interpolating c^* at $s^*(s'_1 = b, \xi) \leq s \leq s^*(s'_n, \xi)$.

Policy functions on a fixed grid

- For well behaved felicity functions u, s^* and c^* are monotone in s'. This allows us to move to the next iteration making use of the following two implications:
  - For a given ξ we can obtain consumption policies $c^{(n+1)}(s,\xi)$ by interpolating c^* at $s^*(s'_1 = b, \xi) \leq s \leq s^*(s'_n, \xi)$.
  - For any $s < s^*(s'_1, \xi)$ households would like to choose assets *lower* than the borrowing limit. Of course they cannot. Hence, we know their asset policy: $s'(s,\xi) = b$ and their consumption is $c^{(n+1)}(s,\xi) = (1+r)s + \xi - b$.

Policy functions on a fixed grid

- For well behaved felicity functions u, s^* and c^* are monotone in s'. This allows us to move to the next iteration making use of the following two implications:
  - For a given ξ we can obtain consumption policies $c^{(n+1)}(s,\xi)$ by interpolating c^* at $s^*(s'_1 = b, \xi) \leq s \leq s^*(s'_n, \xi)$.
  - For any $s < s^*(s'_1, \xi)$ households would like to choose assets *lower* than the borrowing limit. Of course they cannot. Hence, we know their asset policy: $s'(s,\xi) = b$ and their consumption is $c^{(n+1)}(s,\xi) = (1+r)s + \xi - b$.
- We then iterate $c^{(n)}$ until convergence.

EGM: Further issues

- One important issue is how to choose starting guesses for $c^{(0)}$. Here it is useful to recall that the infite horizon planning problem can be viewed as the limit of a finite horizon problem. Hence start with $c^{(0)} = (1+r)s + \xi$ (if possible).

EGM: Further issues

- One important issue is how to choose starting guesses for $c^{(0)}$. Here it is useful to recall that the infite horizon planning problem can be viewed as the limit of a finite horizon problem. Hence start with $c^{(0)} = (1+r)s + \xi$ (if possible).
- There are extensions on

EGM: Further issues

- One important issue is how to choose starting guesses for $c^{(0)}$. Here it is useful to recall that the infite horizon planning problem can be viewed as the limit of a finite horizon problem. Hence start with $c^{(0)} = (1+r)s + \xi$ (if possible).
- There are extensions on
  - **How to deal with multiple assets** (Hintermaier and Koeniger, 2010): Find the asset combinations in $t+1$ that can be optimal using FOCs, then map back from only these optimal points.

EGM: Further issues

- One important issue is how to choose starting guesses for $c^{(0)}$. Here it is useful to recall that the infite horizon planning problem can be viewed as the limit of a finite horizon problem. Hence start with $c^{(0)} = (1+r)s + \xi$ (if possible).
- There are extensions on
  - **How to deal with multiple assets** (Hintermaier and Koeniger, 2010): Find the asset combinations in $t+1$ that can be optimal using FOCs, then map back from only these optimal points.
  - **How to deal with non-convex setups** (Fella, 2015): Use the fact that FOCs are still necessary and compare potential solutions.

Excursus: Policy functions inducing a Markov chain (Young's method)

Suppose we have solved a dynamic programming problem

$$V(s,\xi) = \max_{s'} u(s,s') + \beta E_{\xi'} V(s',\xi')$$

at nodes $(s,\xi) \in S \times \Xi$ by **linearly** interpolating V off nodes, where S and Ξ are indexed sets $S = \{s_1, \ldots, s_n\}$, $\Xi = \{\xi_i, \ldots, \xi_m\}$.

(For exposition we assume that S is one dimensional, but everything extends to higher dimensional S.)

Policy functions and indexes

Then, we have obtained a policy function

$$s^*(s,\xi) = \arg\max_{s'} u(s,s') + \beta E_{\xi'} V(s',\xi').$$

Now let $i^*(s,\xi)$ be the index of the next smallest element in S relative to s^*, i.e. $s^* \in [s_i, s_{i+1})$.

Policy functions: Linear interpolation weights

Define weights $w(s,\xi) = \frac{s^* - s_{i^*}}{s_{i^*+1} - s_{i^*}}$.

- then the linearly interpolated value function is given by

$$V(s,\xi) = u\left[s, s^*(s,\xi)\right] + \left[1 - w(s,\xi)\right] E_\xi V(s_{i^*}, \xi') + w(s,\xi) E_\xi V(s_{i^*+1}, \xi')$$

- Observe that V' is now only on grid in the s dimension!

Policy functions: Linear interpolation weights

Now assume the evolution of ξ is given by a discrete Markov chain.

- then the linearly interpolated value function is given by

$$\begin{aligned} V(s_i,\xi_j) &= u\left[s_h, s^*(s_i,\xi_j)\right] \\ &+ \sum_{j'} p_{jj'}\left\{ \left[1 - w(s_i,\xi_j)\right] V(s_{i^*},\xi_{j'}) + w(s_i,\xi_j) V(s_{i^*+1},\xi_{j'}) \right\} \end{aligned} \tag{4}$$

- Observe that V' fully on grid!

Policy functions: Linear interpolation weights

Moreover, we can reinterpret the weights

- Define
$$\gamma_{(i,j)\to(i',j')} = \begin{cases} p_{jj'}\left[1 - w(s_i,\xi_j)\right] & \text{if } i' = i^*(i,j) \\ p_{jj'} w(s_i,\xi_j) & \text{if } i' = i^*(i,j)+1 \\ 0 & \text{else} \end{cases} \tag{5}$$
- Then $\Gamma := \left[\gamma_{(i,j)\to(i',j')}\right]_{(i,j)}^{(i',j')}$ is a stochastic (transition) matrix
- of a DMC on the vectorized state space.

Policy functions: Transition probability matrix

Why is this useful?

- We can reinterpret the linear interpolant:
  The decision maker chooses only (fair) lotteries over on-grid points.

$$V_t = \max_{s'} u(s,s') + \beta \Gamma_{s'} V_{t+1}$$

- This we can use to obtain the ergodic distribution of states the planning problem induces **without simulation** and therefore fast.
- If an ergodic distribution exists, it is given by the left unit eigenvector of $\Gamma = [\gamma(i,j)]_{i=1...N}^{j=1...N}$, as

$$\mu_{t+1} = \mu_t \Gamma.$$

Introducing aggregate risk

With aggregate risk

- Prices and distributions change over time

Yet, for the household

- Only prices and continuation values matter
- Distributions do not influence the decisions directly

Redefining equilibrium (Reiter, 2002)

A **sequential equilibrium with recursive individual planning**

- A sequence of discretized Bellman equation, such that

$$v_t = \bar{u}_{P_t} + \beta \Pi_{h_t} v_{t+1} \tag{6}$$

  holds for optimal policy, h_t (which results from v_{t+1} and P_t)
- and a sequence of histograms, such that

$$d\mu_{t+1} = d\mu_t \Pi_{h_t} \tag{7}$$

  holds given the optimal policy
- (Policy functions, h_t, that are optimal given P_t, v_{t+1})
- Prices, distributions and policies lead to market clearing

Compact notation (Schmitt-Grohé and Uribe, 2004)

The equilibrium conditions as a non-linear difference equation

- Controls: $Y_t = \begin{bmatrix} v_t & P_t & Z_t^Y \end{bmatrix}$ and
- States: $X_t = \begin{bmatrix} \mu_t & S_t & Z_t^X \end{bmatrix}$
  where Z_t are purely aggregate states/controls
- Define

$$F\left(d\mu_t, S_t, d\mu_{t+1}, S_{t+1}, v_t, P_t, v_{t+1}, P_{t+1}, \varepsilon_{t+1}\right) \tag{8}$$

$$= \begin{bmatrix} d\mu_{t+1} - d\mu_t \Pi_{h_t} \\ v_t - \left(\bar{u}_{h_t} + \beta \Pi_{h_t} v_{t+1}\right) \\ S_{t+1} - H(S_t, d\mu_t, \varepsilon_{t+1}) \\ \Phi(h_t, d\mu_t, P_t, S_t) \\ \varepsilon_{t+1} \end{bmatrix}$$

s.t.

$$h_t(s) = \arg \max_{x \in \Gamma(s, P_t)} u(s, x) + \beta \mathbb{E} v_{t+1}(s') \tag{9}$$

Compact notation (Schmitt-Grohé and Uribe, 2004)

In words

- First set of equations: Difference of one forward iteration of the distribution to assumed value.
- Second set of equations: Difference of one backward iteration of the value function (or policy functions in EGM) to assumed value.
- Last two sets of equations: Macro model.

Compact notation (Schmitt-Grohé and Uribe, 2004)

The equilibrium conditions as a non-linear difference equation

- Function-valued difference equation $\mathbb{E}F(X_t, X_{t+1}, Y_t, Y_{t+1}, \varepsilon_{t+1}) = 0$
- turns real-valued when we replace the functions by their discretized counterparts

- Standard techniques to solve by perturbation (Dynare etc)

Perturbation References: General

- General:
  1. A First Look at Perturbation Theory by James G. Simmonds and James E. Mann Jr.
  2. Advanced Mathematical Methods for Scientists and Engineers: Asymptotic Methods and Perturbation Theory by Carl M. Bender, Steven A. Orszag.

- This lecture:
  1. "Perturbation Methods for General Dynamic Stochastic Models" by Hehui Jin and Kenneth Judd.
  2. "Computational Methods for Economists" by Jesus Fernandez-Villaverde.
  3. "Solving Dynamic General Equilibrium Models Using a Second-Order Approximation to the Policy Function" by Martin Uribe and Stephanie Schmitt-Grohe.

Non-linear difference equation

- A large class of economic models can be written as a set of non-linear difference equations of the form

$$E_t f(s_{t+1}, s_t, c_{t+1}, c_t) = 0$$

where s are all state and c are all control variables now.

Perturbation methods

- More generally, functional equations of the form:

$$\mathcal{H}(d) = 0$$

for an unknown decision rule d.

- Perturbation solves the problem by specifying:

$$d^n(x,\theta) = \sum_{i=0}^{n} \theta_i (x - x_0)^i$$

- We use implicit-function theorems to find coefficients θ_i's
- Inherently local approximation.

Motivation

- Many complicated mathematical problems have:
  - either a particular case
  - or a related problem

  that is easy to solve.

- Often, we can use the solution of the simpler problem as a building block of the general solution.

- Sometimes perturbation is known as asymptotic methods.

A simple example

- Imagine we want to compute $\sqrt{26}$ by hand
- Note that:
$$\sqrt{26} = \sqrt{25 * 1.04} = 5 * \sqrt{25 * 1.04} \approx 5 * 1.02 = 5.1$$
- Exact solution: $\sqrt{26} = 5.09902$
- More generally:
$$\sqrt{x} = \sqrt{y^2 * (1+\epsilon)} = y * \sqrt{(1+\epsilon)} \approx y * (1+\epsilon)$$
- Accuracy depends on how big ϵ is

Applications in economics

- Judd and Guu (1993) showed how to apply it to economic problems
- Recently, perturbation methods have been gaining much popularity
- In particular, second- and third-order approximations are easy to compute and notably improve accuracy
- Perturbation theory is the generalization of the well-known linearization strategy
- Hence, we can use much of what we already know about linearization

Regular versus singular perturbations

- Regular perturbation: a small change in the problem induces a small change in the solution.
- Singular perturbation: a small change in the problem induces a large change in the solution.

- Example: excess demand function.
- Most problems in economics involve regular perturbations.
- Sometimes, however, we can have singularities.
  Example: introducing a new asset in an incomplete market model.

Perturbation methods

- Back to our economic model cast in the following form:

$$E_t f(s_{t+1}, s_t, c_{t+1}, c_t) = 0$$

where s are state and c are control variables.

- Rewrite f introducing a parameter for uncertainty :

$$E_t f(s_{t+1}, s_t, c_{t+1}, c_t; \sigma) = 0$$

.

Evolution of states

- Dynamic stochastic general equilibrium models in addition have a structure where a subset of state variables s_t^1 are predetermined and endogenous while the remainder are exogenously driven as
$$s_{t+1}^2 = H_2(s_t^2, \sigma) + \sigma\Sigma\epsilon_{t+1}$$
where ϵ_{t+1} are i.i.d. with zero mean, unit covariance and bounded support.
- Stacking all state variables, we can write
$$s_{t+1} = [H_1(s_t, \sigma); H_2(s_t, \sigma)] + \sigma\eta\epsilon_{t+1}$$
- with $\eta = [0; \Sigma]$

Evolution of controls

- That c_t is a control means that there is a function $c_t = G(s_t, \sigma)$
- The goal is to solve for the unknown H_1 and G.

Local approximation

- Take a Taylor series approximation of G and H:

$$
\begin{aligned}
G(s,\sigma) = {} & G(s^*,\sigma^*) + G_s(s^*,\sigma^*)(s-s^*) + G_\sigma(s^*,\sigma^*)(\sigma-\sigma^*) \\
& + 1/2G_{ss}(s^*,\sigma^*)(s-s^*)^2 + G_{s\sigma}(s^*,\sigma^*)(s-s^*)(\sigma-\sigma^*) \\
& + 1/2G_{\sigma\sigma}(s^*,\sigma^*)(\sigma-\sigma^*)^2 + \ldots
\end{aligned}
$$

$$
\begin{aligned}
H(s,\sigma) = {} & H(s^*,\sigma^*) + H_s(s^*,\sigma^*)(s-s^*) + H_\sigma(s^*,\sigma^*)(\sigma-\sigma^*) \\
& + 1/2H_{ss}(s^*,\sigma^*)(s-s^*)^2 + H_{s\sigma}(s^*,\sigma^*)(s-s^*)(\sigma-\sigma^*) \\
& + 1/2H_{\sigma\sigma}(s^*,\sigma^*)(\sigma-\sigma^*)^2 + \ldots
\end{aligned}
$$

Local approximation

- Replace s and c in F:

$$F(s,\sigma) \equiv E_t f\left(H(s_t,\sigma) + \sigma\eta\epsilon_{t+1}, s_t, G[H(s_t,\sigma) + \sigma\eta\epsilon_{t+1},\sigma], G(s_t,\sigma)\right)$$
$$= 0$$

- The goal is to solve for the unknown H_1 and G.
- Local approximation means that we solve for H_1, G by taking a Taylor expansion of F around the non-stochastic steady state s^*, for which $\sigma^* = 0$.

Local approximation

- Define non-stochastic steady state as vectors (s^*, c^*) :

$$f(s^*, s^*, c^*, c^*) = 0$$

- $c^* = G(s^*, 0)$ and $s^* = H(s^*, 0)$
- Note that if $\sigma = 0$, then $E_t f = f$

Local approximation

- Approximation of G and H around the point $(s,\sigma) = (s^*,0)$

$$G(s,0) = G(s^*,0) + G_s(s^*,0)(s - s^*) + G_\sigma(s^*,0)\sigma$$
$$H(s,0) = H(s^*,0) + H_s(s^*,0)(s - s^*) + H_\sigma(s^*,0)\sigma$$

- $G(s^*,0), H(s^*,0)$ identified by steady state values
- Remaining coefficients are identified by:

$$F_s(s^*,0) = 0$$
$$F_\sigma(s^*,0) = 0$$

Local approximation

- Take derivative of F w.r.t. uncertainty :

$$\begin{aligned} F_\sigma(s^*,0) &= E_t f_{s'}(H_\sigma + \eta\epsilon') + f_{c'}[G_s(H_\sigma + \eta\epsilon') + G_\sigma] + f_c G_\sigma \\ &= f_{s'} H_\sigma + f_{c'}[G_s H_\sigma + G_\sigma] + f_c G_\sigma \\ &= 0 \end{aligned}$$

- This is a system of n equations:

$$\begin{pmatrix} f_{s'} + f_{c'} G_s & f_{c'} + f_c \end{pmatrix} \begin{pmatrix} H_\sigma \\ G_\sigma \end{pmatrix} = 0$$

- This equation is linear and homogeneous in H_σ, G_σ. Thus we have that $H_\sigma = 0$ and $G_\sigma = 0$.

Local approximation

Important theoretical result:

- In words, up to first order, we do not need to adjust the steady state solution when changing aggregate risk σ.
- Expected values of s_t and c_t are equal to their non-stochastic steady-state values.
- In a first order approximation the certainty equivalence principle holds, i.e., the policy function is independent of the variance-covariance matrix of ϵ.
- Interpretation: no precautionary behavior.

Local approximation

- Differentiation w.r.t s yields:

$$F_s(s^*,0) = f_{s'}H_s + f_s + f_{c'}G_sH_s + f_cG_s = 0$$

- In matrix form:

$$\begin{pmatrix} f_{s'} & f_{c'} \end{pmatrix} \begin{pmatrix} I \\ G_s \end{pmatrix} H_s = -\begin{pmatrix} f_s & f_c \end{pmatrix} \begin{pmatrix} I \\ G_s \end{pmatrix}$$

Local approximation

- Let $A = [f_{s'} \quad f_{c'}]$ and $B = [f_s \quad f_c]$
- Let $\hat{s}_t \equiv s_t - s^*$, then postmultiply:

$$A \begin{pmatrix} I \\ G_s \end{pmatrix} H_s \hat{s}_t = -B \begin{pmatrix} I \\ G_s \end{pmatrix} \hat{s}_t$$

- Consider a perfect foresight equilibrium. In this case, $H_s \hat{s}_t = \hat{s}_{t+1}$

$$A \begin{pmatrix} I \\ G_s \end{pmatrix} \hat{s}_{t+1} = -B \begin{pmatrix} I \\ G_s \end{pmatrix} \hat{s}_t$$

Local approximation

- This leaves us with a system of quadratic equations that we need to solve for H_s, G_s.
- Procedures to solve rational expectations models:
  1. Blanchard and Kahn (1980).
  2. Uhlig (1999).
  3. Sims (2000).
  4. Klein (2000).

Local properties of the solution I

- Perturbation is a local method.
- It approximates the solution around the deterministic steady state of the problem.
- It is valid within a radius of convergence.

Local properties of the solution II

- What is the radius of convergence of a power series around x? An $r \in \mathcal{R}_{+}^{\infty}$ such that $\forall x', |x' - z'| < r$, the power series of x will converge.

- A Remarkable Result from Complex Analysis:
  The radius of convergence is always equal to the distance from the center to the nearest point where the decision rule has a (non-removable) singularity. If no such point exists then the radius of convergence is infinite.

- Singularity here refers to poles, fractional powers, and other branch powers or discontinuities of the functional or its derivatives.

Solution

- Using an eigenvalue decomposition of $H_s = P\Lambda P^{-1}$ we obtain,

$$\begin{aligned} AZ\Lambda &= BZ \qquad & Z &= [I; G_s]P \\ A &= [\partial F_{s'}, \partial F_{c'}] \qquad & B &= -[F_s, F_c] \end{aligned}$$

  which implies that the solution corresponds to a subset of the solutions to the generalized eigenvalue problem

$$AXD = BX$$

- But which?
- If we have a stable system, then $\lim_{t\to\infty} H_s^t = 0$. Therefore, we are searching for the exactly n_s eigenvalues smaller than unity.

Solution

- Splitting all solutions to the eigenvalue problem above and below eigenvalues of 1, we obtain

$$A \begin{bmatrix} X_{11} & X_{21} \\ X_{12} & X_{22} \end{bmatrix} \begin{bmatrix} D_1 & 0 \\ 0 & D_2 \end{bmatrix} = B \begin{bmatrix} X_{11} & X_{21} \\ X_{12} & X_{22} \end{bmatrix}$$

  and therefore

$$H_s = X_{11} D_1 X_{11}^{-1}$$

  and

$$G_s = X_{12} X_{11}^{-1}$$

Alternative Solution: Time Iteration

- Rendahl (2018) extends linear time iteration
- Intuitive, robust, and easy to implement algorithm. Now x is vector of all controls and states. F_n is transition matrix of states and controls.

- Rewrite difference equation in "end of last period" notation

$$Ax_{t-1} + Bx_t + Cx_{t+1} = 0$$

- Let F_n be a candidate solution:

$$Ax_{t-1} + Bx_t + CF_n x_t = 0$$
$$x_t = -(B + CF_n)^{-1} Ax_{t-1}$$

- Thus update guess F_{n+1} as

$$F_{n+1} = -(B + CF_n)^{-1} A$$

Higher order approximations

- Obtaining higher-order approximations to the solution of the non-linear system is a sequential procedure.
- The coefficients of the ith term of the jth-order approximation are given by the coefficients of the ith term of the ith order approximation, for $j > 1$ and $i < j$.
- More importantly, obtaining the coefficients of the ith order terms of the approximate solution given all lower-order coefficients involves solving a linear system of equations.

Notation: Tensors

- General trick from physics.
- An n^{th} -rank tensor in a m-dimensional space is an operator that has n indices and m^n components and obeys certain transformation rules.
- $[F_y]^i_\alpha$ is the (i, α) element of the derivative of F with respect to y:
  1. The derivative of F with respect to y is an n x n_y matrix.
  2. Thus, $[F_y]^i_\alpha$ is the element of this matrix located at the intersection of the i-th row and α-th column.
  3. Thus, $[F_y]^i_\alpha [G_x]^\alpha_\beta [H_x]^\beta_j = \sum_{\alpha=1}^{n_y} \sum_{\beta=1}^{n_x} \frac{\partial F^i}{\partial y^\alpha} \frac{\partial G^\alpha}{\partial x^\beta} \frac{\partial H^\beta}{\partial x^j}$
- $[F_{yy}]^i_{\alpha\gamma}$
  1. F^i_{yy} is a three dimensional array with n rows, n_y columns, and n_y pages.
  2. $[F_{yy}]^i_{\alpha\gamma}$ denotes the element at the intersection of row i, column α, and page γ

Second order approximation

- Derivatives of $F(s,\sigma)$:

$$[F_{ss}(s^*,0)]^i_{jk} = 0$$
$$[F_{\sigma\sigma}(s^*,0)]^i = 0$$
$$[F_{s\sigma}(s^*,0)]^i_j = 0$$

Second order approximation

- Cross derivatives are equal to zero when evaluated at $(s^*, 0)$:

$$[F_{\sigma s}(s^*,0)]^i_j = [F_{s'}]^i_\beta [H_{\sigma s}]^\beta_j + [F_{c'}]^i_\alpha [G_s]^\alpha_\beta [H_{\sigma s}]^\beta_j + [F_{c'}]^i_\alpha [G_{\sigma s}]^\alpha_\gamma [H_s]^\gamma_j + [F_c]^i_\alpha [G_{\sigma s}]^\alpha_j = 0$$

- This is a system of $n \times n_s$ equations in the $n \times n_s$ unknowns given by the elements of $G_{\sigma s}$ and $H_{\sigma s}$.
- The system is homogeneous in the unknowns. Thus, if a unique solution exists, it is given by $G_{\sigma s} = 0$ and $H_{\sigma s} = 0$.

Second order approximation

Important theoretical result:

- The coefficients of the policy function on the terms that are linear in the state vector do not depend on the size of the variance of the underlying shocks
- Uncertainty only affects the constant term in the policy function

Second order approximation

- Approximation of G and H around the point $(s, \sigma) = (s^*, 0)$

$$
\begin{aligned}
[G(s,\sigma)]^i = & [G(s^*,0)]^i + [G_s(s^*,0)]^i_a[(s-s^*)]_a \\
& + \frac{1}{2}[G_{ss}(s^*,0)]^i_{ab}[(s-s^*)]_a[(s-s^*)]_b \\
& + \frac{1}{2}[G_{\sigma\sigma}(s^*,0)]^i[\sigma^2]
\end{aligned}
$$

$$
\begin{aligned}
[H(s,\sigma)]^j = & [H(s^*,0)]^j + [H_s(s^*,0)]^j_a[(s-s^*)]_a \\
& + \frac{1}{2}[H_{ss}(s^*,0)]^j_{ab}[(s-s^*)]_a[(s-s^*)]_b \\
& + \frac{1}{2}[H_{\sigma\sigma}(s^*,0)]^j[\sigma^2]
\end{aligned}
$$

Second order approximation

- Approximation of G and H around the point $(s, \sigma) = (s^*, 0)$ ctd
- The unknowns of this expansion are $[G_{ss}(s^*, 0)]^i$, $[G_{\sigma\sigma}(s^*, 0)]^i$, $[H_{ss}(s^*, 0)]^j$, and $[H_{\sigma\sigma}(s^*, 0)]^j$
- Derivatives of $F(s, \sigma)$ yield as many equations as we have unknowns. Perfectly identified linear system!

Second order approximation

$$
\begin{aligned}
[F_{xx}(\bar{x},0)]^i_{jk} = & ([f_{y'y'}]^i_{\alpha\gamma}[g_x]^\gamma_\delta[h_x]^\delta_k + [f_{y'y}]^i_{\alpha\gamma}[g_x]^\gamma_k \\
& + [f_{y'x'}]^i_{\alpha\delta}[h_x]^\delta_k + [f_{y'x}]^i_{\alpha k})[g_x]^\alpha_\beta[h_x]^\beta_j \\
& + [f_{y'}]^i_\alpha[g_{xx}]^\alpha_{\beta\delta}[h_x]^\delta_k[h_x]^\beta_j \\
& + [f_{y'}]^i_\alpha[g_x]^\alpha_\beta[h_{xx}]^\beta_{jk} \\
& + ([f_{yy'}]^i_{\alpha\gamma}[g_x]^\gamma_\delta[h_x]^\delta_k + [f_{yy}]^i_{\alpha\gamma}[g_x]^\gamma_k + [f_{yx'}]^i_{\alpha\delta}[h_x]^\delta_k + [f_{yx}]^i_{\alpha k})[g_x]^\alpha_j \\
& + [f_y]^i_\alpha[g_{xx}]^\alpha_{jk} \\
& + ([f_{x'y'}]^i_{\beta\gamma}[g_x]^\gamma_\delta[h_x]^\delta_k + [f_{x'y}]^i_{\beta\gamma}[g_x]^\gamma_k + [f_{x'x'}]^i_{\beta\delta}[h_x]^\delta_k + [f_{x'x}]^i_{\beta k})[h_x]^\beta_j \\
& + [f_{x'}]^i_\beta[h_{xx}]^\beta_{jk} \\
& + [f_{xy'}]^i_{j\gamma}[g_x]^\gamma_\delta[h_x]^\delta_k + [f_{xy}]^i_{j\gamma}[g_x]^\gamma_k + [f_{xx'}]^i_{j\delta}[h_x]^\delta_k + [f_{xx}]^i_{jk} \\
= & 0; \quad i = 1,\ldots,n, \quad j,k,\beta,\delta = 1,\ldots,n_x; \quad \alpha,\gamma = 1,\ldots,n_y.
\end{aligned}
$$

- System of nxn_sxn_s linear equations in the nxn_sxn_s unknowns given by the elements of G_{ss} and H_{ss}.

Higher order approximations

- We can iterate this procedure as many times as we want.
- We can obtain n-th order approximations.
- Levintal (2017) uses tensor-unfolding to work with higher-order derivatives

- Problems:
  1. Existence of higher order derivatives.
  2. Numerical instabilities.
  3. Computational costs.

Example: Simple RBC

Stochastic neoclassical growth model

$$
\begin{aligned}
max \quad & E_0 \sum_{t=0}^{\infty} \beta^t \log c_t \\
\texttt{s.t.} \quad & c_t + k_{t+1} = e^{z_t} k_t^{\alpha} \\
& z_t = \rho z_{t-1} + \sigma \epsilon_t, \quad \epsilon \sim \mathcal{N}(0,1)
\end{aligned}
$$

- Note: full depreciation.
- Equilibrium conditions:

$$
\begin{aligned}
\frac{1}{c_t} =& \beta E_t \frac{1}{c_{t+1}} \alpha e^{z_{t+1}} k_{t+1}^{\alpha-1} \\
c_t + k_{t+1} =& e^{z_t} k_t^{\alpha} \\
z_t =& \rho z_{t-1} + \sigma \epsilon_t
\end{aligned}
$$

Solution and steady state

- Exact solution (found by "guess and verify"):

$$
\begin{aligned}
c_t =& (1-\alpha\beta)e^{z_t}k_t^{\alpha} \\
k_t =& (\alpha\beta)e^{z_t}k_t^{\alpha}
\end{aligned}
$$

- Steady state is also easy to find:

$$
\begin{aligned}
k =& (\alpha\beta)^{\frac{1}{1-\alpha}} \\
c =& (\alpha\beta)^{\frac{\alpha}{1-\alpha}} - (\alpha\beta)^{\frac{1}{1-\alpha}} \\
z =& 0
\end{aligned}
$$

The goal

- We are searching for decision rules:

$$
\begin{aligned}
c_t =& c(k_t, z_t) \\
k_{t+1} =& k(k_t, z_t)
\end{aligned}
$$

- Then we have:

$$
\begin{aligned}
\frac{1}{c(k_t, z_t)} =& \beta E_t \frac{1}{c(k(k_t, z_t), z_{t+1})} \alpha e^{z_{t+1}} k(k_t, z_t)^{\alpha-1} \\
c(k_t, z_t) + k(k_t, z_t) =& e^{z_t} k_t^{\alpha} \\
z_t =& \rho z_{t-1} + \sigma \epsilon_t
\end{aligned}
$$

- This is a system of functional equations (after substituting z_t)

A perturbation solution

- Add perturbation parameter σ
  - When $\sigma = 0$ deterministic case (with $z_0 = 0$ and $e^{z_t} = 1$)
  - When $\sigma > 0$ stochastic case
- Now we are searching for decision rules:

$$\begin{aligned} c_t &= c(k_t, z_t; \sigma) \\ k_{t+1} &= k(k_t, z_t; \sigma) \end{aligned}$$

Taylor's theorem

- We will build a local approximation around $(k^*, 0; 0)$
- Given equilibrium conditions:

$$\frac{1}{c(k_t, z_t; \sigma)} = \beta E_t \frac{1}{c(k(k_t, z_t; \sigma), \rho z_t + \sigma \epsilon_{t+1}; \sigma)} \alpha e^{\rho z_t + \sigma \epsilon_{t+1}} k(k_t, z_t; \sigma)^{\alpha - 1}$$
$$c(k_t, z_t; \sigma) + k(k_t, z_t; \sigma) = e^{\rho z_{t-1} + \sigma \epsilon_t} k_t^{\alpha}$$

- Take derivatives w.r.t. k_t, z_t, σ and evaluate them around $(k^*, 0; 0)$

Compact Notation

$$F(k_t, z_t, \sigma) = E_t \begin{pmatrix} \frac{1}{c(k_t,z_t;\sigma)} - \beta E_t \frac{\alpha e^{\rho z_t + \sigma \epsilon_{t+1}} k(k_t,z_t;\sigma)^{\alpha-1}}{c(k(k_t,z_t;\sigma),\rho z_t + \sigma \epsilon_{t+1};\sigma)} \\ c(k_t, z_t; \sigma) + k(k_t, z_t; \sigma) - e^{\rho z_{t-1} + \sigma \epsilon_t} k_t^{\alpha} \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix}$$

- Note that:

$$\begin{aligned} F(k_t, z_t, \sigma) &= \mathcal{H}(c_t, c_{t+1}, k_t, k_{t+1}, z_t; \sigma) \\ &= \mathcal{H}(c(k_t, z_t; \sigma), c(k(k_t, z_t; \sigma), z_t; \sigma), k_t, k(k_t, z_t; \sigma), z_t; \sigma) \end{aligned}$$

- Because $F(k_t, z_t, \sigma)$ must be equal to zero for any possible values of k , z , and σ, the derivatives of any order of F must also be zero.

First-order approximation

- Take first-order derivatives of $F(k_t, z_t, \sigma)$ around $(k^*, 0; 0)$

$$
\begin{aligned}
F_k(k, 0; 0) &= 0 \\
F_z(k, 0; 0) &= 0 \\
F_\sigma(k, 0; 0) &= 0
\end{aligned}
$$

Second-order approximation

- Take second-order derivatives of $F(k_t, z_t, \sigma)$ around $(k^*, 0; 0)$

$$
\begin{aligned}
F_{kk}(k, 0; 0) &= 0 \\
F_{kz}(k, 0; 0) &= 0 \\
F_{k\sigma}(k, 0; 0) &= 0 \\
F_{zz}(k, 0; 0) &= 0 \\
F_{z\sigma}(k, 0; 0) &= 0 \\
F_{\sigma\sigma}(k, 0; 0) &= 0
\end{aligned}
$$

- We substitute the coefficients that we already know.
- A linear system of 12 equations on 12 unknowns.
- Cross-terms on $k\sigma$ and $z\sigma$ are zero.
- More general result: all the terms in odd derivatives of σ are zero.

Correction for risk

- We have the term $1/2c_{\sigma\sigma}(k,0;0)$
- Captures precautionary behavior.
- We do not have certainty equivalence any more!
- Important advantage of second order approximation.
- Changes ergodic distribution of states.

Higher-order terms

- We can continue the iteration for as long as we want.
- Great advantage of procedure: it is recursive!
- Often, a few iterations will be enough.
- The level of accuracy depends on the goal of the exercise: e.g. Welfare analysis: Kim and Kim (2001).

Computer Exercise 2

Exercise

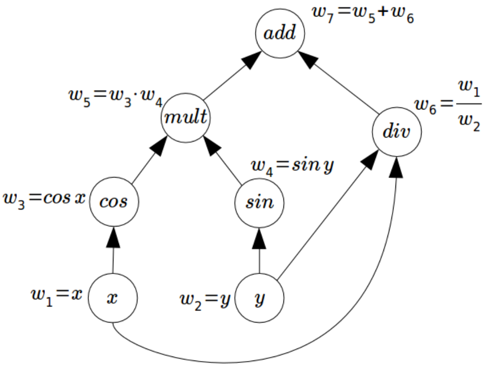
Solve the simple stochastic growth model using perturbation methods. For this purpose, first write a function that calculates the Euler equation errors, errors from capital accumulation, and the law of motion for productivity. Define consumption as control and capital and productivity as states. Compare the first and second order perturbation of $c(k, z, \sigma)$ *to the true solution.*

Excursus: Automatic differentiation

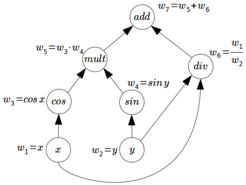
- Modern computer languages like Julia offer easy to implement automatic differentiation libraries.
- Automatic differentiation is neither:
  - Symbolic differentiation
    - Inefficient code
  - nor Numerical differentiation (the method of finite differences)
    - If you make h too small, then your accuracy gets killed by floating point roundoff
    - If h is too big, then approximation errors start ballooning
- AD avoids these problems: it calculates exact derivatives, so your accuracy is only limited by floating point error.

Excursus: Automatic differentiation

- AD applies the chain rule to your function
- Any complicated function f can be rewritten as the composition of a sequence of primitive functions
- Let $f(x, y) = \cos x \sin y + \frac{x}{y}$



Excursus: Automatic differentiation



$$Df = Dw_7 = D(w_5 + w_6) = Dw_5 + Dw_6$$

$$Dw_6 = D\frac{w_1}{w_2} = \frac{w_1 Dw_2 - w_2 Dw_1}{w_2^2}$$

$$Dw_5 = Dw_3 w_4 = w_3 Dw_4 + w_4 Dw_3$$

$$Dw_4 = D\sin w_2 = \cos w_2 \cdot Dw_2$$

$$Dw_3 = D\cos w_1 = -\sin w_1 \cdot Dw_1$$

$$Dw_2 = Dy$$

$$Dw_1 = Dx$$

Excursus: Automatic differentiation

- AD is implemented by a nonstandard interpretation of the program in which real numbers are replaced by dual numbers and the numeric primitives are lifted to operate on dual numbers.
- Dual numbers: Replace every number x with the number $x + x'\varepsilon$, where x' is a real number, but ε is an abstract number with the property $\varepsilon^2 = 0$

- Julia does this for you!
- ForwardDiff Package: `www.juliadiff.org/`
- Examples: `www.juliadiff.org/ForwardDiff.jl/stable/user/advanced.html`

Excursus: Julia

- Julia combines three key features for highly intensive computing tasks as perhaps no other contemporary programming language does: it is fast, easy to learn and use, and open source.

- Introduction by Fernandez-VillaVerde: `www.sas.upenn.edu/~jesusfv/Chapter_HPC_8_Julia.pdf`
- Introduction by QuantEcon: `https://lectures.quantecon.org/jl/`

Back to heterogeneous agent model

The equilibrium conditions as a non-linear difference equation

- Controls: $Y_t = \begin{bmatrix} v_t & P_t & Z_t^Y \end{bmatrix}$ and
- States: $X_t = \begin{bmatrix} \mu_t & S_t & Z_t^X \end{bmatrix}$
  where Z_t are purely aggregate states/controls
- Define

$$F\left(d\mu_t, S_t, d\mu_{t+1}, S_{t+1}, v_t, P_t, v_{t+1}, P_{t+1}, \varepsilon_{t+1}\right) \tag{10}$$
$$= \begin{bmatrix} d\mu_{t+1} - d\mu_t \Pi_{h_t} \\ v_t - \left(\bar{u}_{h_t} + \beta \Pi_{h_t} v_{t+1}\right) \\ S_{t+1} - H(S_t, d\mu_t, \varepsilon_{t+1}) \\ \Phi(h_t, d\mu_t, P_t, S_t) \\ \varepsilon_{t+1} \end{bmatrix}$$

s.t.

$$h_t(s) = \arg \max_{x \in \Gamma(s, P_t)} u(s, x) + \beta \mathbb{E} v_{t+1}(s') \tag{11}$$

So, is all solved?

The dimensionality of the system F is still an issue

- With high dimensional idiosyncratic states, discretized value functions and distributions become large objects
- For example:
  4 income states (grid points)
  $\times$ 100 illiquid asset states
  $\times$ 100 liquid asset states
  $\implies \geq 40,000$ control variables in F
- Same number of state variables

Our Reduction Method

Reiter (2009): Reduce dimensionality ex ante

Problem:

- Dimensionality of the difference equation is large

Proposal:

- Reduce dimensionality ex ante (before solving the StE)
- e.g. (sparse) splines to represent policy functions
- Then linearize

Winberry (2016) extends this to the distribution functions

What we do

Proposal:

- Reduce dimensionality after StE, but before linearization
- Extract from the StE the *important* basis functions to represent individual policies (akin to image compression)
- Perturb only those basis functions but use the StE as "reference frame" for the policies (akin to video compression)
- Similarly for distributions (details later)

Our idea

1.) Apply compression techniques as in video encoding

- Apply a discrete cosine transformation to all value/policy functions (Chebycheff polynomials on roots grid)
- Define as reference "frame": the StE value/policy function
- Write fluctuations as differences from this reference frame
- Assume all coefficients of the DCT from the StE close to zero do not change after shock

Our idea

2.) Transform joint-distribution μ into copula and marginals

- Calculate the Copula, $\bar{\Xi}$ of μ in the StE
- Perturb the marginal distributions
- Approximate changes in the Copula (via DCT) or use fixed Copula to calculate an approximate joint distribution from marginals

- Idea follows Krusell and Smith (1998) in that some moments of the distribution do not matter for aggregate dynamics

Copula

A distribution of probabilities

A *Copula* is a joint distribution function of univariate marginal probabilities for a multivariate stochastic variable. It maps $[0,1]^n \rightarrow [0,1]$

Sklar's theorem

Every distribution function F can be represented by the marginal distribution functions F_i and a *Copula*, Ξ, with $F(x_1,\ldots,x_n) = \Xi\left[F_1(x_1),\ldots,F_n(x_n)\right]$.

Details

1.) Apply compression techniques as in video encoding

- DCT yields the coefficients of the fitted (multi-dimensional) Chebyshev polynomial, where the polynomial is constructed such that the tensor grid for s is mapped to the Chebyshev knots.
  See Ahmed et al. (1974) for the seminal contribution.
- Importantly, the absolute value of the coefficients has an interpretation in terms of the R^2 contribution of the corresponding polynomial in fitting the data.
- This allows us to order and select the polynomial terms based on their importance.

Excursus: Global polynomial

- Express a function by the coefficients ψ of a polynomial

$$\hat{f}(x) = \sum_{j=1}^{n} \psi_j c_j(x)$$

where $c_i(x)$ are known basis functions such as $c_j(x) = x^j$.

- Better than ordinary polynomials are usually Chebyshev polynomials of which the baseline functions are

$$c_j(x) = \cos(j \ \arccos x)$$

- These are orthogonal on [-1,1], i.e.

$$\int_{-1}^{1} c_i(x) c_j(x) \frac{1}{\sqrt{1-x^2}} dx = 0 \forall i \neq j$$

Excursus: Global polynomial

- Since the evaluation points x_i are known ("grid"), we can compute $\mathbf{C} = [c_j(x_i)]_{i=1...M,j=1...n}$
- The vector of function values $\hat{\mathbf{f}} = [\hat{f}(x_i)]_{i=1...M}$ is then given by

$$\hat{\mathbf{f}} = \mathbf{C}\psi$$

- Therefore, we can obtain an optimal (minimal MSE) as

$$\psi^* = (\mathbf{C}'\mathbf{C})^{-1}\mathbf{C}'\ \mathbf{f}$$

- The big **advantage** of polynomials is that they can be integrated analytically and that they are differentiable of any order.

Excursus: Global polynomial: issues

- **Runge's Phenomenon:** Since polynomials tend to infinity as $x \to \infty$ it is not true that the overall fit of a global polynomial gets better, if more grid points and higher order polynomials are used (oscillating behavior).

- Choosing **Chebyshev polynomials** as basis functions and
- grid points as the **roots** $x_i = \cos(\frac{2i-1}{2N})$ for $i = 1 \ldots N$ of these polynomials minimizes approximation error.

Excursus: Discrete Cosine Transforms

A first observation

- Suppose Chebychev root grid-points are not suitable for our problem.
- Then, we can write $f(x) = f(g(y))$ and
- generate the grid x_i by applying g to the Chebyshev nodes y_i,
- with basis functions $c_j(x) = \cos\left(j \arccos g^{-1}(x)\right)$

Discrete Cosine Transform (DCT) and lossy compression

- In particular, if we do not intend to evaluate off-grid, we do not need to know g but just the nodes $y_i = \cos\left(\frac{2i-1}{2N}\pi\right)$ and grid values x_i
- and obtain an equivalent representation of f_i in terms of coefficients.
- Shrinking ≈ 0-coefficients to 0 leaves $\hat{f}_i$ close to unchanged.
- In addition $C'C = I$.

Details

1.) Apply compression techniques as in video encoding

- Let $\bar{\Theta} = dct(\bar{v})$ be the coefficients obtained from the DCT of the value function in StE

- A DCT expresses a finite sequence of data points in terms of sum of cosine functions at different frequencies
- Linear, invertible function $f = \Re^N - > \Re^N$ (equivalently: an invertible NxN matrix)
- x_n is transformed to X_n according to:

$$X_k = \sum_{n=0}^{N-1} x_n \cos\left[\pi/N(n+1/2)k\right], k = 0, ..., N-1$$

Details

1.) Apply compression techniques as in video encoding

- Define an index set $\mathcal{I}$ that contains the x percent largest (i.e. most important) elements from $\bar{\Theta}$
- Let θ be a sparse vector with non-zero entries only for elements $i \in \mathcal{I}$
- Define $\tilde{\Theta}(\theta_t) = \begin{cases} \bar{\Theta}(i) + \theta_t(i) & i \in \mathcal{I} \\ \bar{\Theta}(i) & else \end{cases}$

Details

Decoding

- Now we reconstruct $\nu_t = \nu(\theta_t) = idct(\tilde{\Theta}(\theta_t))$
- This means that in the StE the reduction step adds no additional approximation error as $\nu(0) = \bar{\nu}$ by construction
- Yet, it allows to reduce the number of derivatives that need to be calculated from the outset

Details

2) Analogously for the distribution function

- Define μ_t as $\Xi_t(\bar{\mu}_t^1, \ldots, \bar{\mu}_t^n)$ for n being the dimensionality of the idiosyncratic states
- The StE distribution is obtained when $\mu = \bar{\Xi}(\bar{\mu}^1, \ldots, \bar{\mu}^n)$

- We can treat the copula as an interpolant defined on the grid of steady-state marginal distributions, and also approximate Ξ_t as a sparse expansion around the steady-state copula $\bar{\Xi}$.
- The most extreme variant of this is to treat the copula as time fixed.

Details

2) Analogously for the distribution function

- Typically prices are only influenced through the marginal distributions
- The approach ensures that changes in the mass of one dimension, say wealth, are distributed in a sensible way across the other dimensions
- The implied distributions look “similar” to the StE one

Obtaining the copula function of the StE

To obtain an estimate of the Copula of the StE:

1. Accumulate the histogram along every dimension to obtain CDF estimate, M.
2. Add a leading zero to the CDF matrix, M, along every dimension.
3. Calculate marginal distributions, m_i, from the CDF (summing out other dims)
4. Obtain the Copula estimate as an interpolant of M on $\{m_1, \ldots, m_n\}$

$$\bar{\Xi} = \textsc{griddedInterpolant}(\{m_1, \ldots, m_n\}, M)$$

.

Details

Too many equations

- The system

$$F\left(\{d\mu_t^1,\ldots,d\mu_t^n\},S_t,\{d\mu_{t+1}^1,\ldots,d\mu_{t+1}^n\},S_{t+1},\right. \tag{12}$$
$$\left.\theta_t,P_t,\theta_{t+1},P_{t+1}\right)=$$
$$\begin{bmatrix} d\bar{\Xi}(\bar{\mu}_t^1,\ldots,\bar{\mu}_t^n)-d\bar{\Xi}(\bar{\mu}_t^1,\ldots,\bar{\mu}_t^n)\Pi_{h_t} \\ dct\left[idct(\tilde{\Theta}(\theta_t))-\left(\bar{u}_{h_t}+\beta\Pi_{h_t}idct(\tilde{\Theta}(\theta_{t+1}))\right)\right] \\ S_{t+1}-H(S_t,d\mu_t) \\ \Phi(h_t,d\mu_t,P_t,S_t) \end{bmatrix}$$

  has too many equations

- Use only difference in marginals and the differences on $\mathcal{I}$

Quality of approximation

- David Childers (2018), "Automated Solution of Heterogeneous Agent Models":
- Under some regularity conditions the solution algorithm is guaranteed to converge to the first derivative of the true infinite dimensional solution as the discretization is refined.
- Convergence rates for the approximation are provided as well, depending on the choices of interpolation method including polynomials, splines, histograms, and wavelets.

Application: Krusell-Smith model

A simple KS economy

Incomplete Markets and TFP

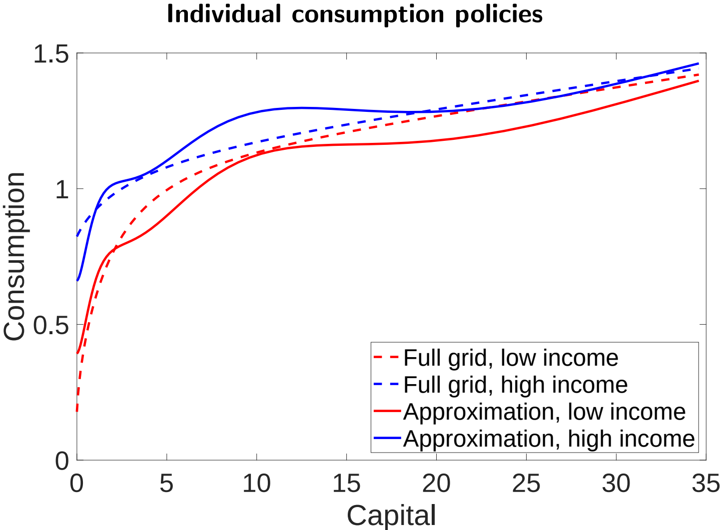
- Household productivity can be high or low
- No contingent claims
- Households save in capital goods (which they rent out)
- Households supply labor (disutility) and consume (utility)
- Aggregate productivity (TFP) follows a log AR-1 process
- Cobb-Douglas production function

A simple KS economy

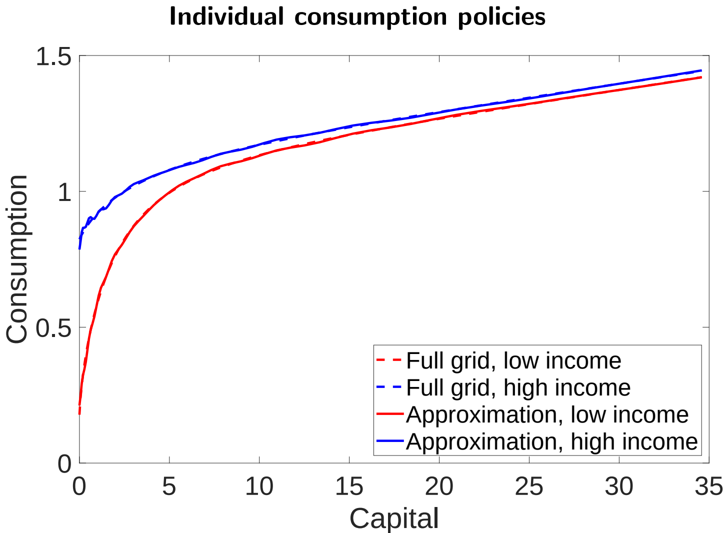
Numerical setup

- Asset grid has 100 points ($\Longrightarrow$ a total grid size of 200)
- Policies solved by EGM (instead of VFI)

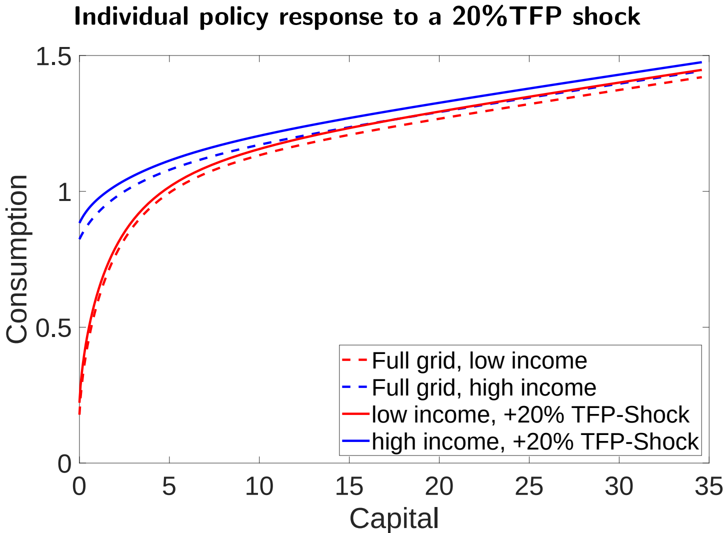
Different levels of “compression”



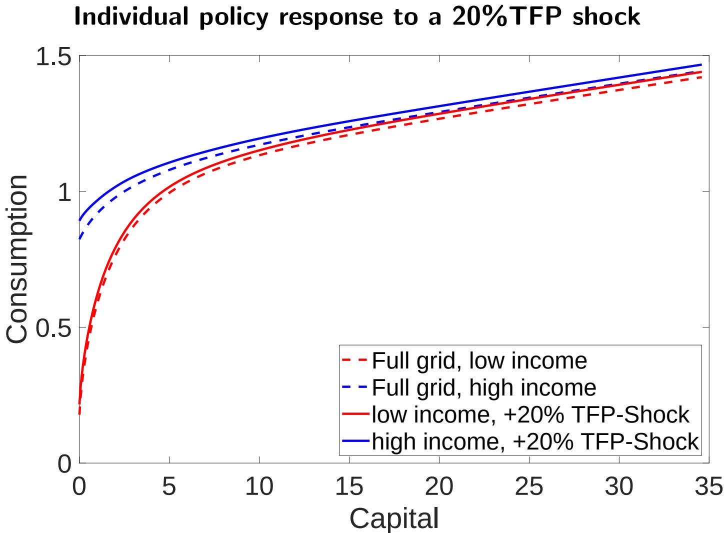
Different levels of “compression”



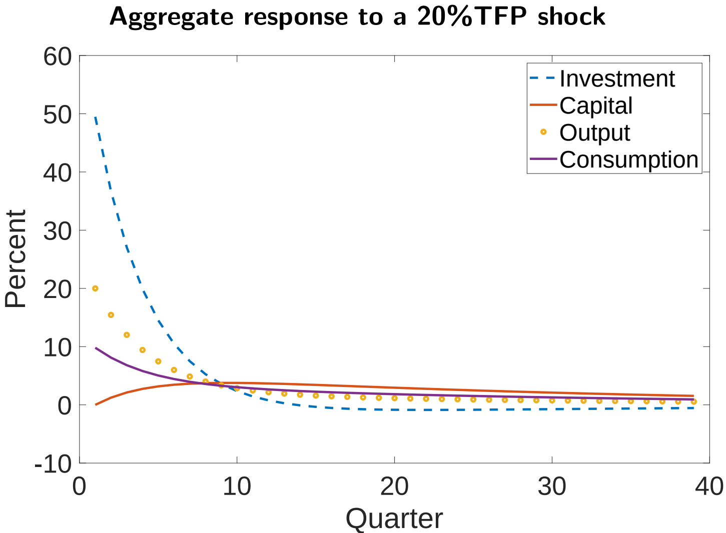
Different levels of "compression"



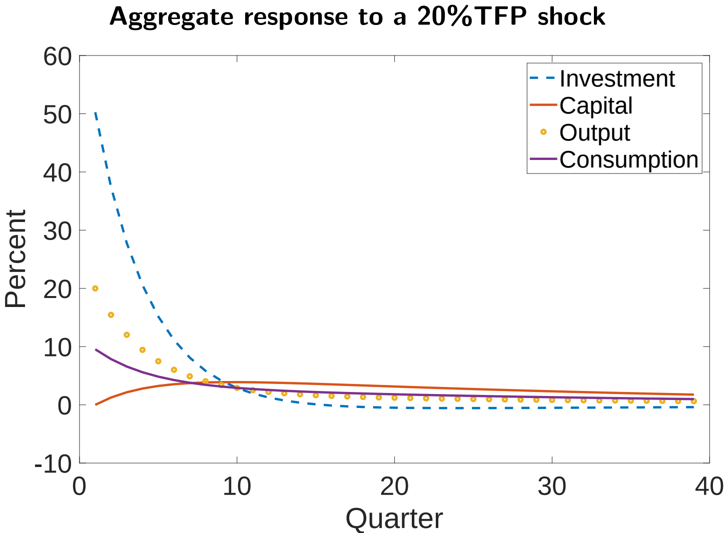
Different levels of “compression”



Different levels of “compression”



Different levels of “compression”

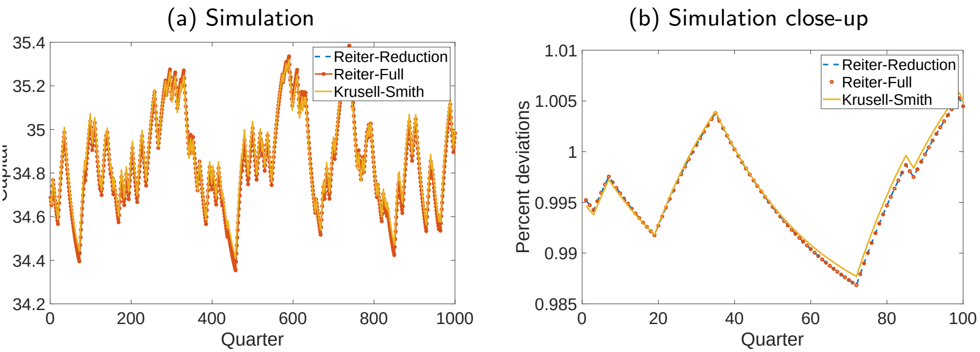


Taking stock

- When looking only at the StE policy function one concludes that roughly 50% of the information is needed to reconstruct the policies well
- This is roughly level of state reduction Reiter (2009) approach would achieve
- Using the StE as reference one can achieve much higher reduction
- For the aggregate dynamics maintaining only 3-6% of the basis functions suffices

Simulation performance

Figure: Simulations of Krusell & Smith model



Notes: Both panels show simulations of the Krusell & Smith (1998) model with TFP shocks solved with (1) the Reiter method with our proposed state-space reduction, (2) the original

Error Statistics

Table: Den Haan errors

	Absolute error (in %) for capital K_t		
	Reiter-Reduction	Reiter-Full	K-S
Mean	0.0119	0.0119	0.1237
Max	0.0152	0.0152	0.3491

Notes: Differences in percent between the simulation of the linearized solutions of the model and simulations in which we solve for the intratemporal equilibrium prices in every period and track the full histogram over time for $t = \{1, ..., 1000\}$; see Den Haan (2010)

Computing time

Table: Run time for Krusell & Smith model

	StE	K & S	Reiter-Reduction	Reiter-Full
in seconds	6.28	49.85	0.43	0.91

Notes: Run time in seconds on a Dell laptop with an Intel i7-7500U CPU @ 2.70GHz x 4. Model calibration and number of grid points as in Den Haan et al. (2010). Code in Matlab.

MIT shock solution

- See Kurt Mitman's slides.

Computer Exercise 3

Exercise

Solve the Krusell-Smith model using first order perturbation. For this purpose, first solve the steady state by either EGM or VFI. Then write a function that calculates the Euler equation errors, errors from capital accumulation, and the law of motion for productivity.

Computer Exercise 4

Exercise

Solve the Krusell-Smith model using MIT shock solution approach.

Computer Exercise 5

Exercise

Solve the Krusell-Smith model using first order perturbation and dimensionality reduction proposed by Bayer and Luetticke (2018). For this purpose, split the joint-distribution into Copula and marginals and define the marginals as state. Apply the DCT transformation to the policy function and keep only the most important basis functions as controls. Write the corresponding non-linear difference equations as a function Fsys.

Application: Estimating HANK models

Bayer, Born, Luetticke (2020): Shocks, Frictions, and Inequality in US Business Cycles

What we do

- Fuse two-asset HANK model with a Smets-Wouters-type medium scale DSGE model
- Estimate the model using (Bayesian) full-information approach
- IRF analysis and variance decompositions

- Research Question:
  What shocks and frictions drive the US business cycle and US inequality?

Overview of the model

Workers		Production Sector	Government
Trade Assets	**Obtain Income**	**Produce and Differentiate Consumption Goods**	**Monetary Authority, Fiscal Authority**
Bonds, b>B; and Illiquid Assets (trading friction)	Wages set by unions s.t. Rotemberg wage adjustment costs (Idiosyncratic Income Risk) Interest Dividends Profits	**Intermediate goods producers** Rent capital & labor Competitive Market for Intermediate Goods **Entrepreneurs** Monopolistic resellers s.t. Rotemberg price adjustment costs	Policy Rules: • Monetary authority sets nominal interest rate -> Taylor rule • Fiscal authority supplies government debt, consumes goods, taxes labor income and profits -> Spending rule

Introducing more macro structure

Linearization techniques easily allow for more structure

- Say, we want to add price stickiness, monetary, and fiscal policy.
- This requires additional extra state variables.
- This is numerical cheap when linearizing.

Recap: Reiter's method(s)

The starting point is the following observation:

- For the household, current prices and a sequence of value functions suffices to describe the decision problem. In discretized form this is

$$v_t = \bar{u}_{h_t} + \beta \Gamma_{h_t} v_{t+1} \tag{13}$$

- h_t is the optimal policy given prices (or other aggregate controls) P_t and continuation values v_t
- This induces payoffs $\bar{u}_{h_t}$ and a transition matrix Γ_{h_t}
- and this transition matrix also induces the law of motion

$$\mu_{t+1} = \mu_t \Gamma_{h_t} \tag{14}$$

- We can view (13) and (14) as the equation describing the idiosyncratic part of a sequential equilibrium with recursive individual planning.

Recap: Compact notation (Schmitt-Grohé and Uribe, 2004)

Allows to write equilibrium as non-linear difference equation

- Add P_t and S_t, purely aggregate controls and states, respectively.
- Define "market-clearing" conditions $\Phi(h_t, \mu_t, P_t, S_t)$
- and a mapping $\Xi(S_t, P_t, \sigma\Sigma\varepsilon_{t+1})$ of controls to $t+1$ states
- Define

$$F\left(\mu_t, S_t, \mu_{t+1}, S_{t+1}, v_t, P_t, v_{t+1}, P_{t+1}, \varepsilon_{t+1}\right) \tag{15}$$

$$= \begin{bmatrix} \mu_{t+1} - \mu_t \Gamma_{h_t} \\ S_{t+1} - \Xi(S_t, P_t, \sigma\Sigma\varepsilon_{t+1}) \\ v_t - \left(\bar{u}_{h_t} + \beta\Gamma_{h_t} v_{t+1}\right) \\ \Phi(h_t, \mu_t, P_t, S_t) \\ \varepsilon_{t+1} \end{bmatrix}$$

s.t.

$$h_t(s) = \arg\max_{x \in \Gamma(s, P_t)} u(s, x) + \beta \mathbb{E} v_{t+1}(s') \tag{16}$$

Observations

Changing the aggregate macro structure is easy

- As long as a change in the model does not affect what income is composed of and which choices households can make given prices and incomes, but only how prices are formed, we can change the aggregate part of the model without touching the micro part.

- Modular: Micro and Macro block: $F(\dots) = [F_1, F_2]'$

$$
\begin{aligned}
F_1 &= \begin{bmatrix} d\bar{\Xi}(\bar{\mu}_t^1, \dots, \bar{\mu}_t^n) - d\bar{\Xi}(\bar{\mu}_t^1, \dots, \bar{\mu}_t^n)\Gamma_t \\ dct\left[idct(\tilde{\Theta}(\theta_t)) - \left(\bar{u}_{h_t} + \beta\Gamma_t idct(\tilde{\Theta}(\theta_{t+1}))\right)\right] \end{bmatrix} \\
F_2 &= \begin{bmatrix} S_{t+1} - \Xi(S_t, P_t) \\ \Phi(h_t, \mu_t, P_t, S_t) \end{bmatrix}
\end{aligned}
$$

Fusing incomplete markets and price stickiness

A HANK model

- Introduce price stickiness (NKPC),
- introduce a central bank (Taylor rule),
- introduce a fiscal authority (Expenditure rule),
- introduce capital goods producers (quadratic adjustment costs).

Issues:

- We need to deal with assigning monopoly rents.
- We need to deal with the portfolio problem (Bonds vs. Capital).
- We need to make labor supply endogenous

A HANK model

See Bayer et al., 2019

Extending the household sector

1. Assume GHH preferences (for business cycles reasonable)

$$u(c,n) = \frac{\left(c - h\frac{n^{1+\gamma}}{1+\gamma}\right)^{1-\xi}}{1-\xi}$$

   Scaling with productivity h allows for easy aggregation w.l.o.g. if taxes are linear.

2. Assign profits to either to (a) a group of households, (b) the government, or (c) a profit-asset.

A HANK model

Modeling portfolio choice: easy version

- All households hold the same bonds-to-capital ratio.
- All assets can be traded without any friction.
- Choice is over total wealth.
- For first order approximation: Returns must equal in expectations, i.e. define a safe return on bonds R_t, prices of capital goods q_t and rental rates of capital r_t, then

$$\mathbb{E}_t \frac{r_{t+1} + q_{t+1}}{q_t} = R_{t+1}$$

Equilibrium conditions (idiosyncratic part)

This leaves us with the following equilibrium conditions:
(A) idiosyncratic part, using linear interpolations in micro problem:

1. Recursive planning. For the vectors of marginal utilities $\mathbf{u}_{c,t}$:

$$\mathbf{u}_{c,t} = \underbrace{\beta R_{t+1} \Gamma_t (\mathbf{u}_{c,t+1} + \lambda_{t+1})}_{\text{one EGM backwards step}}$$

   with Γ_t induced by optimal policies,
   - given future marginal utils $\mathbf{u}_{c,t+1}$, and expected returns R_{t+1}
   - and current incomes determined through wages w_t, dividends r_t, profits π_t, and capital prices q_t.

2. Law of motion for distribution of capital

$$\mu_{t+1} = \mu_t \Gamma_t$$

Equilibrium conditions (summary variables)

(B) summary variables, model free:

1. It is useful to introduce an aggregate control that summarize μ_t: $K_t := \sum_j k_j \mu_t^j$ where k_j is the capital grid.
2. Let $\phi_t := \frac{B_t}{K_t}$ be the bonds-to-capital ratio entering period t.
3. For any unit of capital households hold, they have $r_t + q_t + \phi_t R_t$ resources for consumption.
4. Every unit of capital for next period sells at $q_t + \phi_{t+1}$

Equilibrium conditions (macro model)

(C) prices:

1. Factor prices as controls from FOCs of firms

$$w_t = (1-\alpha) mc_t Z_t \left(\frac{K_t}{N_t}\right)^{\alpha}, \quad r_t = \alpha mc_t Z_t \left(\frac{N_t}{K_t}\right)^{1-\alpha} - \delta,$$

prices of undifferentiated goods, mc_t, and total profits accordingly

$$\hat{\pi}_t = \beta \mathbb{E}_t \hat{\pi}_{t+1} + \kappa \left(mc_t^{-1} - \bar{\mu}\right)$$

$$\Pi_t = (1 - mc_t) Y_t - \text{adjustment costs/profits}$$

2. Returns on government bonds from Taylor rule (state variable)

$$R_{t+1} = R_t^{\rho_R} \hat{\pi}_t^{(1-\rho_R)\theta_\pi} \hat{Y}_t^{(1-\rho_R)\theta_Y}$$

Observe that adjustment costs are zero up to first order around stationary equilibrium.

Equilibrium conditions (macro model)

(D) aggregate quantities:

1. Labor supply
$$(1-\tau)w_t = N_t^{\gamma}$$
2. Production of capital goods (ignore externality)
$$q_t = 1 + \phi \frac{K_{t+1} - K_t}{K_t}$$
3. Total output and components
$$Y_t = Z_t K_t^{\alpha} N_t^{1-\alpha}, \quad C_t = Y_t - G_t - K_{t+1} + (1-\delta)K_t$$
4. A fiscal rule (spending adjusts, B_t is a state, G_t a control)
$$\hat{G}_t = (\hat{B}_t \hat{R}_t / \hat{\pi}_t)^{\rho_B} \hat{\pi}_t^{\gamma_\pi} \hat{Y}_t^{\gamma_Y}, \quad B_{t+1} = G_t + B_t R_t / \pi_t - \tau w_t N_t$$
5. Goods market clearing is residual.

2 asset models

HANK models have more action with more assets

- The literature has highlighted the role of wealthy hand-to-mouth consumers (Kaplan et al., 2014).
- HANK models with more assets feature asset substitution as in the older Keynesian literature (see e.g. Tobin, 1969), which is supported by the data (see Bayer et al., 2019; Luetticke, 2020).

A tractable structure

- For many applications it suffices to assume that capital can only be traded from time to time randomly (Calvo shock).

2 asset models

2 marginal values of assets

- Marginal value of liquid assets results from usual envelop condition

$$V_b(h,b,k) = Ru_c$$

- Marginal value of illiquid assets results from usual envelop condition

$$V_k(h,b,k|adjust) = (q+r)u_c^a$$

  when trade is possible and from the marginal value of the dividend payment plus discounted marginal value if no trade is possible

$$V_k(h,b,k|not) = ru_c^n + \beta \mathbb{E} V_k'(h',b',k)$$

- Thus, $V_k(h,b,k) = \lambda(q+r)u_c^a + (1-\lambda)(ru_c^n + \beta \mathbb{E} V_k'(h',b',k))$
- Optimal asset choices require $q\mathbb{E}V_b(h',b',k') = \mathbb{E}V_k(h',b',k')$ which allows us to trace out potentially optimal $(b',k')(h)$ pairs

Computational aspects of HANK-2

- requires both assets (100 points each)
- 22 income states
- uses both value functions and consumption policies as controls
- Full set would have $> 600,000$ variables
- Reduction to 219 distribution states and ca 700 controls for value functions and policies

Exercise 6: Krusell-Smith-model with nominal rigidity

Exercise

Take the setup from last exercise: *and add a government that runs a central bank, a fiscal authority and owns all profit incomes. Households have GHH preferences over labor and consumption, but still unemployment shocks.*

1. *Solve for the steady state without aggregate risk.*
2. *Solve using Bayer and Luetticke's refinement.*

Assume the central bank only reacts to inflation and past interest rates $\rho_R = 0.95$ *and* $\theta_\pi = 1.25$. *The fiscal side only reacts to the level of debt* $\rho_B = -0.1$. *Assume steady state profits are* 10% *and the Phillips Curve reflects price adjustment of roughly once a year if it was from Calvo. Assume steady state labor taxes are 25%.*

Sources of Fluctuations

Standard in complete markets model

- total factor productivity
- gov. bond spread (a.k.a. "risk premium")
- price markup
- wage markup
- monetary policy
- government spending

Sources of Fluctuations

Standard in complete markets model

- total factor productivity
- gov. bond spread (a.k.a. "risk premium")
- price markup
- wage markup
- monetary policy
- government spending

New in the incomplete markets model

- idiosyncratic income risk
- tax progressivity

Solution and Estimation

Solution method

- The distribution Θ over b, k, h is a state variable
- First-order perturbation of the non-linear difference equation $EF(x_t, x_{t+1}, \epsilon_t) = 0$ around the stationary equilibrium to obtain a local approximation to the solution
- We approximate the policy functions as sparse polynomials around their stationary equilibrium values and approximate the distribution functions by histograms of their marginals and a time-varying Copula as in Bayer and Luetticke (2020).

Estimation: Overview

- i.e., method linearizes the resulting non-linear difference equation
- Write as $Ax_t = Bx_{t+1}$ and solve using standard methods
- State-space representation of the model solution

- Use Kalman filter to evaluate likelihood
- Maximize posterior likelihood
- Draw from posterior

Estimation: Numerical details

For each new draw of the parameter vector:

1. Update of Jacobian of $F[.]$ (less then 1sec)
2. Solve linear state space model
   - State&Control vector has roughly 1000 entries
   - Klein's method via schur decomposition (ca. 5sec)
3. Run Kalman Filter to obtain log-likelihood (ca. 1sec)

Estimation: Update of Jacobian

Changing parameters does not mean we have to update everything

- Since estimated parameters do not directly show up in household problem, they change only small parts of A, B during estimation.
- This means that writing $F(\dots) = [F_1, F_2]'$,

$$F_1 = \begin{bmatrix} d\bar{\Xi}(\bar{\mu}_t^1, \dots, \bar{\mu}_t^n) - d\bar{\Xi}(\bar{\mu}_t^1, \dots, \bar{\mu}_t^n)\Gamma_t \\ dct\left[idct(\tilde{\Theta}(\theta_t)) - \left(\bar{u}_{h_t} + \beta\Gamma_t idct(\tilde{\Theta}(\theta_{t+1}))\right)\right] \end{bmatrix}$$
$$F_2 = \begin{bmatrix} S_{t+1} - \Xi(S_t, P_t) \\ \Phi(h_t, \mu_t, P_t, S_t) \end{bmatrix}$$

  implies we only need to update the Jacobian of F_2 when changing model parameters and $\Phi(h_t, d\mu_t, P_t, S_t) = \bar{\Phi}(P_t, S_t)$ except for summary equation.

- Number of derivatives to be calculated same as in RANK
- Convenient also in terms of coding: Only dynare-like macro model block needs to be edited to change the macro model.

Estimation: Solve linear state space model

- Write as $Ax_t = Bx_{t+1}$ and solve using Klein's method to obtain G and H.
- Uses generalized Schur decomposition (computational efficiency)
- Algorithm cost $\mathcal{O}(n^3)$ floating point operations

- We also experimented with the Anderson and Moore (1985) algorithm. While it is more than twice as fast as Klein's method for the HANK model with two assets in many cases, it appears to produce less numerically stable results in a setting such as ours, where the Jacobians are not very sparse.

Estimation: Solve linear state space model

Alternative: Speed up Linear Time Iteration (Papp&Reiter, 2020)

- $F_{n+1} = -(B + CF_n)^{-1}A$
- F need not be initialized to zero if an estimate of F is available from an earlier calculation with similar parameter values.
- The linear equation system can be solved making use of a variant of the Sherman-Morrison-Woodbury formula (blockwise matrix inversion).
- Computational complexity only depends on number of states (and not controls)!

Estimation: Kalman filter

Similar to An and Schorfheide (2007) and Fernández-Villaverde (2010)

1. Kalman filter to obtain the likelihood from the state-space representation of the model solution.
2. Advantage of State-Space: Deal with mixed frequency and missing observations.
3. Roughly one evaluation of the Kalman filter every other second.
4. Maximize posterior likelihood

Estimation: Kalman filter

- For a one-frequency data set without missing values, one can speed up the estimation by employing so-called "Chandrasekhar recursions" for evaluating the likelihood.
- These recursions replace the costly updating of the state variance matrix by multiplications involving matrices of much lower dimension (see Herbst, 2014, for details).
- This is especially relevant for the two-asset HANK model as the speed of evaluating the likelihood is dominated by the updating of the state variance matrix, which involves the multiplication of matrices that are quadratic in the number of states.

Estimation: Posterior

- Random Walk Metropolis Hastings algorithm to draw from posterior
- Standard to draw 200k times to recover the posterior distribution

Speed up:

- Run multiple chains
- Go sequential Monte Carlo (NY Fed has implemented this for our perturbation approach)

Gaussian State Space

- Solution to linearized model takes state space form, which can be written as

$$x_{t+1} = Gx_t + w_{t+1},\ w_{t+1} \overset{iid}{\sim} \mathcal{N}(0,Q) \tag{17}$$

$$y_t = Hx_t + v_t,\ v_t \overset{iid}{\sim} \mathcal{N}(0,R) \tag{18}$$

- G, H, Q, R are functions of the model parameters θ
- x_t is an $n_x \times 1$ vector of states
- y_t is an $n_y \times 1$ vector of observables
- w_t is a $p \times 1$ vector of structural errors
- v_t a vector of measurement errors
- Assumption: w_t and v_t are orthogonal

$$E_t(w_{t+1}v_s) = 0 \quad \forall\ t+1 \text{ and } s \geq 0$$

Fundamental Problem: Unobserved States

- This implies that
$$y_t = H\left(Gx_{t-1} + w_t\right) + v_t \tag{19}$$
- Thus, y_t is normally distributed:
$$y_t \sim \mathcal{N}\left(HGx_{t-1}, HQH + R\right) \tag{20}$$
- If all states were observed, we could directly construct the likelihood $f(y_T, \ldots, y_1|\theta)$
- We could then run optimizer over our estimated parameter set $\tilde{\theta} \subseteq \theta$ to get ML estimate of $\tilde{\theta}$
- Problem: we have unobserved states and cannot use equation (20)
- Solution: turn to Kalman filter to back out states from the observed data $\rightarrow$ Filtering problem

Kalman Filter: Summary

At time t, given $\hat{x}_{t|t-1}, \Sigma_{t|t-1}$ and observing y_t

1. Compute the forecast error in the observations using

$$a_t = y_t - H\hat{x}_{t|t-1} \tag{21}$$

2. Compute the Kalman Gain K_t using

$$K_t = G\Sigma_{t|t-1}H'\left(H\Sigma_{t|t-1}H' + R\right)^{-1} \tag{22}$$

3. Compute the state forecast for next period given today's information

$$\hat{x}_{t+1|t} = G\hat{x}_{t|t-1} + K_t\left(y_t - H\hat{x}_{t|t-1}\right) = G\hat{x}_{t|t-1} + K_t a_t \tag{23}$$

4. Update the covariance matrix

$$\Sigma_{t+1|t} = (G - K_t H)\,\Sigma_{t|t-1}\,(G - K_t H)' + Q + K_t R K_t' \tag{24}$$

Kalman Filter: Initialization

- How to initialize filter at $t = 0$ where no observations are available?

  $\rightarrow$ start with unconditional mean $E(x)$ and Variance Σ

- Given covariance stationarity, the unconditional mean is

$$E(x) = Ex_{t+1} = E(Gx_t + w_{t+1}) = GE(x) \Rightarrow (I - G)E(x) = 0$$

  hence, $E(x) = 0$

- For the covariance matrix, we have

$$\begin{aligned}\Sigma &= E\left[(Gx_t + w_t)\,(Gx_t + w_t)'\right]\\ &= E\left[Gx_t{x_t}'G' + w_t{w_t}'\right]\\ &= G\Sigma G' + Q \end{aligned} \tag{25}$$

  $\rightarrow$ so-called Lyapunov-equation

Metropolis Hastings-Algorithm

- Start with a vector θ_0
- Repeat for $j = 1, \ldots, N$
  - Generate $\tilde{\theta}$ from $q(\theta_{j-1}, \cdot)$ and u from $\mathcal{U}(0,1)$
  - If $\tilde{\theta}$ is valid parameter draw (steady state exists, Blanchard-Kahn conditions satisfied etc.) and $u < \alpha(\theta^{j-1}, \theta^{j})$ set $\theta_j = \tilde{\theta}$
  - Otherwise, set $\theta_j = \theta_{j-1}$ (implies setting $\pi(\tilde{\theta}) = 0$ if draw invalid)
- Return the values $\{\theta_0, \ldots, \theta_N\}$
- After the chain has passed the transient stage and the effect of the starting values has subsided, the subsequent draws can be considered draws from the posterior

$\Rightarrow$ burnin required that assures remaining chain has converged

The Random-Walk Metropolis Hastings Algorithm

- As long as the regularity conditions are satisfied, any proposal density will ultimately lead to convergence to the invariant distribution
- However: speed of convergence may differ significantly
- In practice, people often use the Random-Walk Metropolis Hastings algorithm where

$$q\left(\theta,\tilde{\theta}\right) = q_{RW}\left(\tilde{\theta}-\theta\right) \tag{26}$$

  and q_{RW} is a multivariate density
- The candidate $\tilde{\theta}$ is thus given by the old value θ plus a random variable increment

$$\tilde{\theta} = \theta + z, z \sim q_{RW} \tag{27}$$

Estimation: Two-step procedure

- First, we calibrate or fix all parameters that affect the steady state of the model.
- Second, we estimate by full-information methods all parameters that only matter for the dynamics of the model, i.e., the aggregate shocks, frictions, and policy rules.
- We set the priors for shocks, frictions, and policy rules to standard values from the representative agent literature

Observables

Quarterly US data from 1954Q1 – 2019Q4

In first-differences

- GDP, Consumption, Investment
- the real wage

In log-levels

- GDP deflator based inflation rates
- Hours worked per capita
- the (shadow) federal funds rate

All demeaned and without measurement error.

Observables

Further data non-quarterly availability

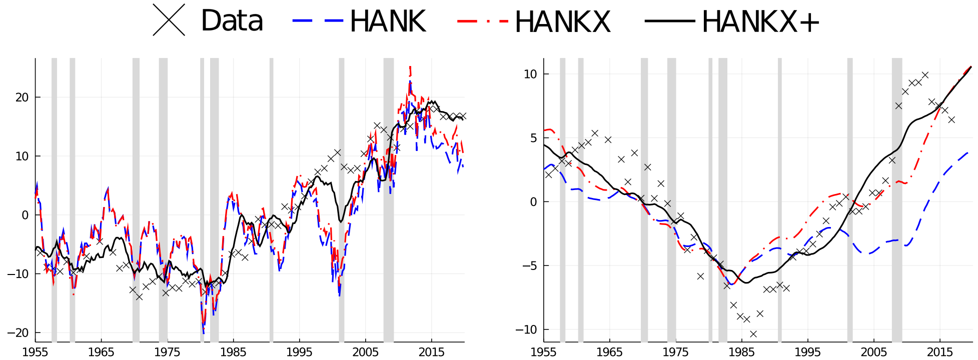
- Measures of inequality:
  - Wealth share of the top 10% (Piketty-Saez WID) (1954 – 2019)
  - Income share of the top 10% (Piketty-Saez WID) (1954 – 2019)

All in log-levels, demeaned and with measurement error.

Estimated model variants

Data / Shocks	Aggregate Data	+ Cross-sectional Data
Aggregate Shocks	HANK (vs RANK)	HANKX
+ Cross-sectional Shocks		HANKX+

Wealth Inequality in the US



(a) Top 10% income share (b) Top 10% wealth share

Shocks and Frictions in US Business Cycles

Variance decomposition: GDP and components

US business cycles: Summary

HANK and RANK models give only a somewhat different view

Estimation results

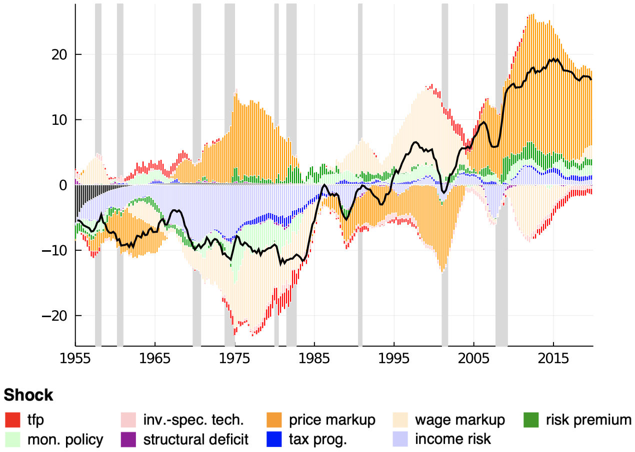
- Key is the estimation that makes the dynamics of both models more similar
- Estimated HANK model features less nominal and real frictions than RANK

Decomposition results

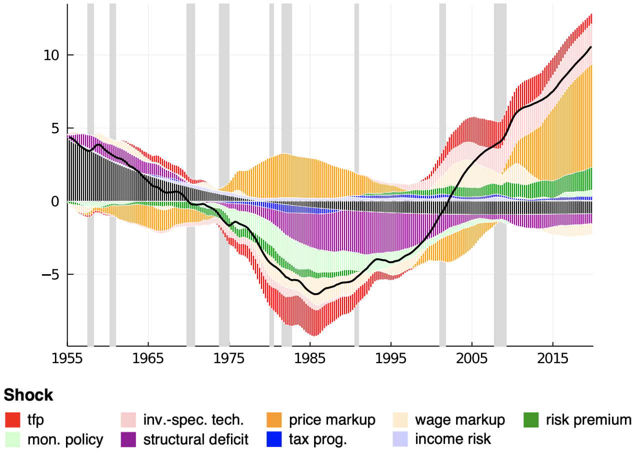
- Investment-specific technology becomes less important because it induces wealth effects on consumption via asset prices
- Risk premium, monetary, and wage markup shocks become more important
- Income risk shocks can partly replace risk premium shocks

Shocks and Frictions in US Inequality

Shock decomposition: Income share of top 10%



Shock decomposition: Wealth share of top 10%



Contribution of shocks to US inequality 1985-2019

Shock	Top 10% Income	Top 10% Wealth
TFP, ϵ^{Z}	-0.38	2.63
Inv.-spec. tech., ϵ^{Ψ}	-0.17	3.26
Price markup, $\epsilon^{\mu Y}$	11.69	4.3
Wage markup, $\epsilon^{\mu W}$	5.82	0.87
Risk premium, ϵ^{A}	-0.62	2.07
Income risk, ϵ^{σ}	2.57	-0.14
Monetary policy, ϵ^{R}	1.30	1.98
Structural deficit, ϵ^{G}	-0.05	1.60
Tax progressivity, τ^{P}	1.54	0.67
Sum of shocks	21.55	16.79

US inequality: Summary

Business cycle shocks are important drivers of inequality dynamics

Income inequality

- Price and wage markups explain two-third of the increase since 1985
- Rising income risk and falling tax progressivity explain the remaining one-third

Wealth inequality

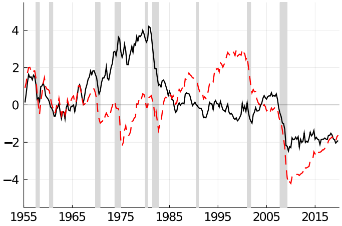
- Technology shocks via their effect on asset prices explain most of the increase since 1985
- The two markup shocks explain only one-third of this increase
- Monetary policy and fiscal deficit shocks are important as well

Policy Counterfactuals

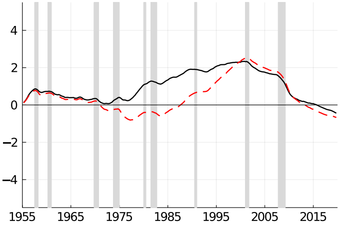
Policy counterfactual: Inequality

- How important are the estimated policy coefficients for the evolution of inequality?
- Run estimated shock sequence with counterfactually set policy parameters
  - Hawkish monetary policy (double inflation response, θ_π)
  - Dovish monetary policy (double output response, θ_Y)

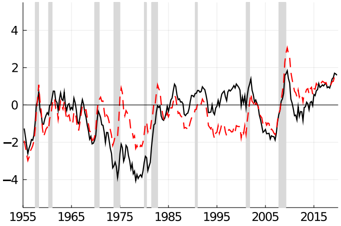
Counterfactual evolution of inequality: Monetary policy



Top 10% income share



Top 10% wealth share



Output

Log point deviations from baseline. Black: Hawkish; Red: Dovish

Policy counterfactual: Summary

Effect of monetary policy depends on supply vs demand shocks

Hawkish monetary policy (triple θ_π)

- Higher inequality in the 70s as markup (cost-push) shocks are important

Dovish monetary policy (triple θ_Y)

- Lower inequality in the 70s and aftermath of the Great Recession

Very persistent effect on wealth inequality.

Summary: Bayer et al. (2020)

Our HANK model can jointly explain the US business cycle and inequality

US business cycle

- Not a radically different view on the US business cycle
- HANK models stress the importance of portfolio choice for the transmission of aggregate shocks

US inequality

- Business cycles are important to understand the evolution of US inequality.
- Business cycle shocks and policy responses can account for most of the increase in US inequality since the 1980s.

Conclusion

No excuse!

- Even when heterogeneity is high dimensional,
- our algorithm is an easy approach to these models
- It is a fast and simple to code

Conclusion

No excuse!

- It requires knowledge of only two standard tools of macro:
  1. Solving a recursive het. agent model for a StE
  2. Linearizing a rep. agent model
  3. (and a little twist in between)
- The fixed design for dimensionality reduction allows to employ the method to estimate models with standard techniques

Bibliography I

Ahmed, N., T. Natarajan, and K. R. Rao (1974). "Discrete cosine transform". *IEEE Transactions on Computers C-23* (1), 90–93.

Ahn, SeHyoun, Greg Kaplan, Benjamin Moll, Thomas Winberry, and Christian Wolf (2018). "When inequality matters for macro and macro matters for inequality". *NBER Macroeconomics Annual 32* (1), 1–75.

An, Sungbae and Frank Schorfheide (2007). "Bayesian analysis of DSGE models". *Econometric Reviews 26* (2-4), 113–172.

Anderson, Gary and George Moore (1985). "A linear algebraic procedure for solving linear perfect foresight models". *Economics Letters 17* (3), 247–252.

Auclert, Adrien, Bence Bardóczy, Matthew Rognlie, and Ludwig Straub (2019). "Using the sequence-space Jacobian to solve and estimate heterogeneous-agent models". NBER Working Paper 26123.

Bayer, Christian, Benjamin Born, and Ralph Luetticke (2020). "Shocks, Frictions, and Inequality in US Business Cycles". CEPR Discussion Paper 14364.

Bibliography II

Bayer, Christian and Ralph Luetticke (2020). "Solving heterogeneous agent models in discrete time with many idiosyncratic states by perturbation methods". *Quantitative Economics forthcoming*.

Bayer, Christian, Ralph Luetticke, Lien Pham-Dao, and Volker Tjaden (2019). "Precautionary savings, illiquid assets, and the aggregate consequences of shocks to household income risk". *Econometrica 87* (1), 255–290.

Boppart, Timo, Per Krusell, and Kurt Mitman (2018). "Exploiting mit shocks in heterogeneous-agent economies: the impulse response as a numerical derivative". *Journal of Economic Dynamics and Control 89*, 68–92.

Carroll, C.D. (2006). "The method of endogenous gridpoints for solving dynamic stochastic optimization problems". *Economics Letters 91* (3), 312–320.

Den Haan, Wouter J (2010). "Assessing the accuracy of the aggregate law of motion in models with heterogeneous agents". *Journal of Economic Dynamics and Control 34* (1), 79–99.

Bibliography III

Den Haan, Wouter J, Kenneth L Judd, and Michel Juillard (2010). "Computational suite of models with heterogeneous agents: incomplete markets and aggregate uncertainty". *Journal of Economic Dynamics and Control 34* (1), 1–3.

Fernández-Villaverde, Jesús (2010). "The econometrics of DSGE models". *SERIEs 1* (1), 3–49.

Herbst, Edward (2014). "Using the "Chandrasekhar Recursions" for likelihood evaluation of DSGE models". *Computational Economics 45* (4), 693–705.

Hintermaier, T. and W. Koeniger (2010). "The method of endogenous gridpoints with occasionally binding constraints among endogenous variables". *Journal of Economic Dynamics and Control 34* (10), 2074–2088.

Kaplan, Greg, Giovanni L Violante, and Justin Weidner (2014). "The wealthy hand-to-mouth". *NBER Working Paper No. 20073.*

Krusell, Per and Anthony A. Smith (1998). "Income and wealth heterogeneity in the macroeconomy". *Journal of Political Economy 106* (5), 867–896.

Bibliography IV

Krusell, Per and Anthony A. Smith (1997). "Income and wealth heterogeneity, portfolio choice, and equilibrium asset returns". *Macroeconomic Dynamics 1* (02), 387–422.

Luetticke, Ralph (2020). "Transmission of monetary policy with heterogeneity in household portfolios". *American Economic Journal: Macroeconomics forthcoming*.

Reiter, Michael (2002). "Recursive computation of heterogeneous agent models". *mimeo, Universitat Pompeu Fabra*.

——— (2009). "Solving heterogeneous-agent models by projection and perturbation". *Journal of Economic Dynamics and Control 33* (3), 649–665.

Schmitt-Grohé, Stephanie and Martın Uribe (2004). "Solving dynamic general equilibrium models using a second-order approximation to the policy function". *Journal of Economic Dynamics and Control 28* (4), 755–775.

Tobin, James (1969). "A general equilibrium approach to monetary theory". *Journal of Money, Credit and Banking 1* (1), 15–29.

Bibliography V

Winberry, Thomas (2016). “A toolbox for solving and estimating heterogeneous agent macro models”. *mimeo, Chicago Booth.*