Comparing Computational Methods for Heterogeneous Agent Models with Aggregate Uncertainty

John Ryan

October 22, 2025

Outline

Two Broad Approaches

Challenge: distribution of agents $\mu_t(k, \epsilon; \mathbf{Z})$ is an infinite-dimensional state variable.

1. Linearization

- Solves a linearized version of the model around the deterministic steady state.
- This linearization can be either around the discretized state of the economy (Reiter), or w.r.t. aggregates in the sequence space (BKM, Auclert et al.)
- Doesn't require expansion in HH's state
- Relies on "certainty equivalence" assumption

2. Approximate aggregation

- Solves the full non-linear model, agents internalize the approximate state of the economy and use it to forecast (Krusell-Smith)
- Handles large shocks and non-linearities (e.g., precautionary motives, regime switching).
- Computationally intensive.

Model

- Let the economy be as in Krusell-Smith, with aggregate shocks:

$$\log(Z_{t+t}) = \rho \log(Z_t) + \sigma \eta_{t+1}$$

- We assume the unemployment transition matrix Π is independent of Z
- Agents with uninsurable unemployment shocks $\epsilon_t \sim \Pi(\epsilon_t | \epsilon_{t-1})$
- HHs save in physical capital k_{t+1}
- Representative firm with $F(Z, K, L) = ZK^{\alpha} L^{1-\alpha}$

What does “Linear in aggregates” mean?

- The model is approximately linear in aggregates if the following holds:

$$g(x; X) \approx g(x; X_{SS}) + (X - X_{SS})' \cdot \nabla_X g(x; X_{SS})$$

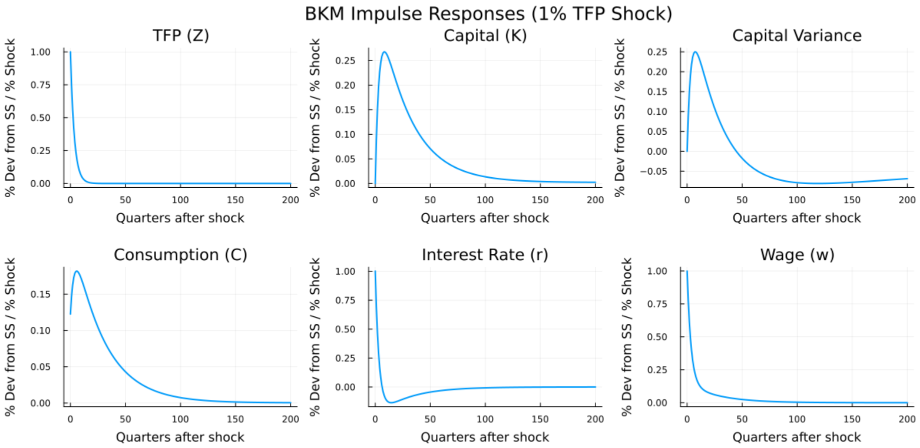
- Where $g(x; X)$ is the decision rule for HH with idiosyncratic state x and aggregate state X
- Note this is fully nonlinear in the idiosyncratic state, but linear w.r.t. aggregate state
- This holds if shocks are small
    - agents don’t want to insure against aggregate risk
- Theoretically $X = (Z, \mu)$ in KS, but need to have some approximation...

Boppart, Krusell, Mitman (2018): IRF as a Numerical Derivative

- **Core Idea:** Use a non-linear transition path of length T from an MIT shock to steady state to approximate the economy's linear response.
- This computed Impulse Response Function (IRF) is treated as a **numerical derivative** in sequence space.
- It represents the marginal effect of a shock at time 0 on all future values of an endogenous variable Y.
- The IRF is found from the resulting sequence $\{Y_0, Y_1, Y_2, \dots Y_T\}$ from the one-time shock.

$$\text{MIT Shock } \eta_0 \quad \Longrightarrow \quad \text{IRF} \quad \mathcal{I}^Y = \left\{ \frac{Y_0 - Y_{SS}}{\varepsilon_0^Z}, \frac{Y_1 - Y_{SS}}{\varepsilon_0^Z}, \dots \right\}$$

Impulse responses



Simulation from IRFs

- Because the system is linear in aggregates, responses to shocks are scalable and additive[1].
- This allows dynamics of the economy with recurring shocks to be simulated by combining the IRF with the history of innovations.
- For a single shock to Z with IRFs $\{\mathcal{I}^Y\}$, the value of Y at time s is the convolution[2]:

$$Y_s = Y_{SS} + \sum_{t=0}^{T} \mathcal{I}_t^Y \cdot \eta_{s-t} = Y_{SS} + \mathcal{I}_0^Y \cdot \eta_s + \mathcal{I}_1^Y \cdot \eta_{s-1} + \dots$$

- This is like an MA(T) representation for Y

[1]This is because shocks are small, aggregates are close to their steady state value. Linear convolution of responses with their shocks is known as the "principle of superposition"

[2]Recall that these IRFs are the same as those from the plots on the previous slide. They are directly obtained from the solution to the transition path

Pseudocode: Boppart, Krusell, Mitman (2018)

1. Initialize

- Compute SS equilibrium $(g_{SS}, \mu_{SS}, K_{SS})$
- Define deterministic MIT shock path: $\{\log(Z_t) = \rho^t \varepsilon_0\}_{t=0}^{T-1}$
- Choose horizon T for return to SS

2. Solve Deterministic Transition Path

- Guess path for aggregates, e.g., $\{K_t\}_{t=0}^{T-1}$.
- Iterate until convergence:
    - Solve household problem backwards: $k_{t+1} = g_t(k_t, \epsilon_t)$.
    - Simulate distribution $\{\mu_t\}$ forward from $\mu_0 = \mu_{SS}$.
    - Check market clearing & update K_t path.

3. Calculate IRFs (Numerical Derivatives)

- IRF is the normalized deviation from SS:

$$IRF_K(t) \leftarrow \mathcal{I}_t^K = \frac{K_t - K_{SS}}{\eta_0}$$

- The values for K_t are from transition path

4. Simulate with Shocks via Convolution

- Generate random shock innovations $\{\varepsilon_t\}_{t=1}^{S}$.
- Simulated path is the convolution of IRF and innovations:

$$K_s = K_{SS} + \sum_{t=0}^{T-1} \mathcal{I}_t^K \eta_{s-t}$$

- This is an MA(T) representation of aggregates in the model.

Reiter (2009)

- Reiter is the first and most literal linearization method
- Discretizes the economy (policy functions, distribution of agents on histogram, TFP)
- Performs a Taylor approximation of the discretized economy around steady state
- The solution with aggregate uncertainty just extrapolates this around the steady state
- This uses tools like Dynare for DSGE solvers
- Simulate the whole economy as a Taylor approximation

Reiter's System

- **Step 1: Discretize the Economy**
  - Approximate policy functions parameterized by vector g_t.
  - Approximate the distribution of agents μ_t with a histogram.
- **Step 2: Define the System Function**
  - Collect all equilibrium conditions into a single vector function f.
  - The state variables are $x_t = \{Z_t, \mu_t\}$, controls are the policy function $y_t = \{g_t\}$.
  - Note: K_t is the first moment of μ_t. Prices $r(x_t)$ and $w(x_t)$ are functions of the state x_t.
  - The system takes the general form $\mathbb{E}_t[f(x_t, y_t, x_{t+1}, y_{t+1}, \eta_{t+1})] = 0$.

$$f(x, y, x', y', \eta') = \begin{pmatrix} u'(c(k_j, \epsilon; y)) - \beta \mathbb{E}\left[(1 + r(x') - \delta) \sum_{\epsilon'} \Pi_{\epsilon'|\epsilon} u'(c'(k'(k_j, \epsilon), \epsilon'; y'))\right] \\ \\ \mu'(k_{j'}, \epsilon') - \sum_{\epsilon} \Pi_{\epsilon'|\epsilon} \sum_{k_j \in \mathcal{A}_d} \omega_{j,\epsilon,j'}(y) \mu(k_j, \epsilon), \quad \forall k_{j'} \in \mathcal{A}_d, \epsilon' \\ \\ \log(Z') - \rho \log(Z) - \sigma \eta' \end{pmatrix}$$

Reiter: Linearization and Solution

- **Step 3: Taylor Expansion**
  - Linearly approximate the system f around the steady state point $f(x^*, y^*, x^*, y^*, 0) = 0$.

$$D_x(x_t - x^*) + D_y(y_t - y^*) + D_{x'}(x_{t+1} - x^*) + D_{y'}(y_{t+1} - y^*) + D_\varepsilon \varepsilon_{t+1} = 0$$

  - The Jacobians $(D_x, D_y, \dots)$ are constant matrices evaluated at the steady state - solved via finite differences or automatic differentiation
- **Step 4: Solve the Linear System**
  - We have a linear system of equations in (x_{t+1}, y_{t+1}).
  - This is solved with algorithms for DSGE models (e.g., 'gensys', Dynare) to get the solution matrices G and H.

$$y_t = Gx_t \quad , \quad \mathbb{E}_t[x_{t+1}] = Hx_t$$

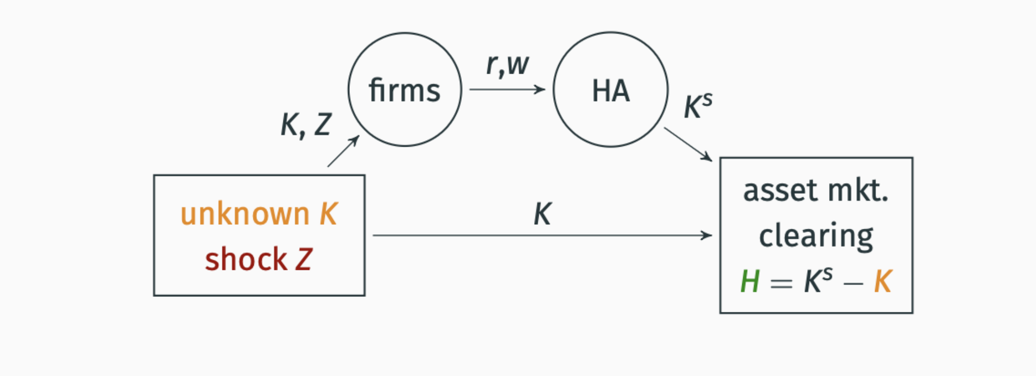
  - From this, we can compute IRFs and simulate the economy's dynamics.

Sequence Space Jacobians

Sequence Space Jacobians (Auclert, Bardoczy, Rognlie, Straub, 2021)

- **Core Idea:** Decompose the model into interconnected blocks. The model's linearized dynamics are solved by combining the derivatives (Jacobians) of each block.
- Solution is a high dimensional application of the implicit function thm. and chain rule
- **Three Main Steps**:
  1. Model the economy as a Directed Acyclic Graph (DAG) of blocks.
  2. Compute each block's Jacobian (the derivative of its outputs with respect to its inputs).
  3. Use the chain rule to combine these Jacobians and solve for the economy's impulse responses.
- This is the fastest method - only need a single length T backward and forward iteration

Directed Acyclic Graph



The DAG and Block Jacobians

- **Inputs**: Exogenous shocks (e.g., Z_t) and endogenous unknowns (e.g., K_t).
- **Outputs**: Intermediate variables and a final market-clearing target (e.g., $\mathbf{H} = \mathbf{K}^s - \mathbf{K}$).
- Equilibrium requires finding the path of unknowns that sets $\mathbf{H} = 0$.
- A **block Jacobian** is the derivative of an output sequence w.r.t. an input sequence, e.g., $\mathcal{J}_{ts}^{K^s,w} = \partial K_t^s / \partial w_s$.

Chaining Jacobians for IRFs

- The GE condition $H(\{K_t\}, \{Z_t\}) = 0$ is linearized around the steady state:

$$\frac{\partial H}{\partial K} dK + \frac{\partial H}{\partial Z} dZ = 0$$

- Total Jacobians, $\frac{\partial H}{\partial K}$ and $\frac{\partial H}{\partial Z}$, are found by applying the chain rule along the DAG.
- The IRF for the unknown sequence dK is solved directly via matrix inversion:

$$dK = -\left(\frac{\partial H}{\partial K}\right)^{-1} \frac{\partial H}{\partial Z} dZ$$

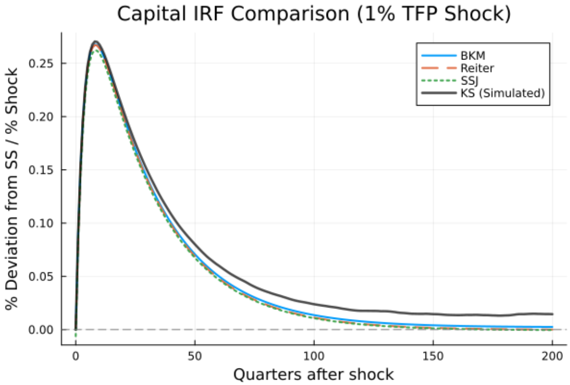
- Their "Fake News" algorithm rapidly computes the Jacobians for the heterogeneous-agent block, making the entire method extremely fast.

Pseudocode: SSJ

1. **Initialize**
   - Model as a DAG with GE condition: $H(\{Z_t\}, \{K_t\}) = \{K_t^S\} - \{K_t\} = 0$.
   - Compute the steady state $(K_{ss}, r_{ss}, w_{ss}, \mu_{ss})$.
   - Choose a truncation horizon T.
2. **Solve for Sequence-Space Jacobians**
   - Compute sequence Jacobians for each block: $\mathcal{J}^{r,K}$, $\mathcal{J}^{w,K}$, $\mathcal{J}^{K^S,r}$, $\mathcal{J}^{K^S,w}$
   - Direct Method, column s of $\mathcal{J}^{\mathcal{K},r}$:
     - Perturb input $r_s \to r_s + \delta$.
     - Solve HH problem backward for $\{k_t^*\}$.
     - Iterate forward from $\mu_0 = \mu_{ss}$
     - Aggregate to get capital path $\{K_t^S\}$.
     - Column $s = (\{K_t^S\} - \{K_{ss}\})/\delta$.
   - Can use faster *Fake News Algorithm*
3. **Linearize and Chain Jacobians**
   - Linearize GE condition: $H_K dK + H_Z dZ = 0$.
   - Use chain rule along DAG to build total Jacobians H_K, H_Z from block Jacobians.
   - Example element of H_K: $[H_K]_{t,s} = \frac{\partial K_t^S}{\partial r_s}\frac{\partial r_s}{\partial K_s} + \frac{\partial K_t^S}{\partial w_s}\frac{\partial w_s}{\partial K_s} - \mathbf{1}_{s=t}$
4. **Invert and Simulate**
   - Solve for GE response by inversion: $dK = -H_K^{-1} H_Z dZ$.
   - GE IRFs: $G \equiv -H_K^{-1} H_Z$.
   - These IRFs are approximately equal to the deviations found in BKM
   - Simulate panel of aggregates from IRFs like BKM

IRFs to a Small TFP Shock (1%)

Here we compare the dynamic response to a 1 percent shock to aggregate productivity Z_t.



Observation: For small shocks, the impulse responses from linearization are very close.

Stochastic Simulation: Moments

We simulate both models for 10,000 periods using the same sequence of shocks to Z_t.

Table: Moment Comparison for 1% TFP Shock

Moment	**KS (Full)**	**BKM (Seq)**	**Reiter (State)**
Mean Capital (K)	11.6283	11.6273	11.6258
Variance of Capital (K)	0.0097	0.0098	0.0097
Mean X-Sect Var(K)	32.1128	32.2479	32.2808
Mean Consumption (C)	0.8386	0.8387	0.8386
Mean Return (R)	1.0100	1.0100	1.0100
Mean Wage (w)	2.3758	2.3759	2.3756

Observation: Both models produce very similar statistics.

Scaling Up the Shocks

Table: Moment Comparison With Scaled Shocks

TFP Shock Scale	KS Capital (K)	BKM Capital (K)	Reiter Capital (K)
1%	11.6283	11.6273	11.6258
5%	11.6730	11.6276	11.6271
10%	11.7851	11.6280	11.6267
25%	12.5175	11.6293	11.6267

- We can see the linearization methods stay near steady state, while the Krusell-Smith solution incorporates a precautionary savings motive against aggregate risk.
- We confirm the approximate equivalence between sequence and state space methods

Computation

Table: Computation Times (Baseline 1% Shock)

Method	**Time (seconds)**
Steady State	0.3358
BKM	3.7741
SSJ	0.3566
Reiter	2.4864
KS	74.2161

The Case Against Linearization (Cons)

Key Limitations

- **Inaccurate for Large Shocks:** Fails when the economy moves far from the steady state (e.g., financial crises, pandemics).
- **Misses Non-Linearities:** Cannot capture effects from:
  - Precautionary behavior changing with the aggregate state.
  - Can't do regime switching (orginal Krusell-Smith)
  - Models where there is no concept of steady state

Pros of linearization

- Computational advantage - can be massive for models with rich state space
- Can include richer aggregate uncertainty - Krusell-Smith uses only 2 points for the TFP process, only one type of aggregate shock (Kaplan, Mitman and Violante example with 3 aggregate shocks)
- Can scale up or down shocks without re-solving (subject to linearization still holding)