Imposing Equilibrium Restrictions in the Estimation of Dynamic Discrete Games

Victor Aguirregabiria & Mathieu Marcoux

University of Toronto & University of Montreal

RICE UNIVERSITY - APPLIED MICROECONOMICS SEMINAR

December 10, 2019

Context & Motivation

- **Estimation of dynamic discrete games**
  - Choice probabilities and structural parameters
  - Two-step methods: simpler, but less appealing in finite samples

- **Gains from imposing equilibrium restrictions**
  - Reduce finite sample bias and improve efficiency
  - Allow for richer unobserved heterogeneity
  - Counterfactuals

- **Merits and limitations of existing algorithms**
  - Nested Pseudo Likelihood (NPL): relatively easy to compute, but convergence issues (in games)
  - Alternatives: local convergence, but high-dim Jacobians

Contributions

- **New asymptotic statistical properties of NPL algorithm**
  - Account for not convergence in some finite samples
  - Convergence depends on the sample NPL mapping
- **Characterize "convergence selection" problem**
  - Samples where NPL algorithm converges differ from others
  - Selection varies with the degree of instability
  - Attenuation bias in strategic interaction parameter
- **Propose spectral algorithm to compute NPL estimator**
  - Instability does not prevent convergence
  - Relatively small computational cost

Related Literature

- **Estimation of dynamic discrete games**
  - Most applications use two-step methods without imposing equilibrium restrictions
  - Some exceptions: ...

- **Papers on properties of NPL algorithm in games**
  - Pesendorfer & Schmidt-Dengler (ECMA, 2010): convergence to inconsistent estimator (e.g., unstable equilibria).
  - Kasahara and Shimotsu (ECMA, 2012): Higher order asymptotic properties.
  - Egesdal, Lai, and Su (2015): simulation evidence of convergence failure even with stable equilibria.

- **Alternative algorithms that impose equilibrium restrictions**
  - Kasahara and Shimotsu (2012): relaxation method
  - Egesdal, Lai, and Su (2015): Mathematical Program with

Outline

1. Model

2. Estimation methods: MLE; 2-steps; NPL

3. Algorithms to compute NPL estimator
   (i) NPL algorithm
   (ii) Spectral algorithm

4. Convergence properties of algorithms

5. Monte Carlo experiments

6. Conclusions

1. Model

Model: Basic framework

- Discrete time indexed by t. N players indexed by i.

- $\mathbf{y}_t = (y_{1t}, y_{2t}, ..., y_{Nt})$ vector with **discrete actions** of N players.
  $\mathbf{x}_t =$ vector of observable discrete **state variables**.

- $\mathbf{P}^0 = \left\{ P^0 \left(\mathbf{y}_t | \mathbf{x}_t \right) \right\} \in \mathcal{P}$ is the true probability distribution.

- **Model for $\mathbf{P}^0$:**
  - $\boldsymbol{\theta} \in \Theta$ vector of **structural parameters**
  - Equilibrium func: $\boldsymbol{\Psi} \left(\boldsymbol{\theta}, \mathbf{P} \right) = \left\{ \Psi \left(\mathbf{y}_t | \mathbf{x}_t ; \boldsymbol{\theta}, \mathbf{P} \right) \right\} : \Theta \times \mathcal{P} \rightarrow \mathcal{P}$
  - For the true value $\boldsymbol{\theta}^0$, we have that:

$$\mathbf{P}^0 = \boldsymbol{\Psi} \left(\boldsymbol{\theta}^0, \mathbf{P}^0 \right)$$

Example: Dynamic game of market entry/exit

- $y_{it} \in \{0, 1\}$ decision of firm i of being in the market at period t.

- Players maximize $\mathbb{E}_t \left(\sum_{s=0}^{\infty} \beta^j \ U_{it+s} \right)$
- State variables:
  - Observable ($\mathbf{x}_t$): Exogenous market characteristics, S_t; Incumbency statuses of all the firms at previous period: $\mathbf{y}_{t-1}$
  - Unobservable: Private information shocks: ε_{it}

- **[Additive Separability]** Profit function of firm i:

$$U_{it} = u_i \left(\mathbf{y}_t, \mathbf{x}_t \right) + \varepsilon_{it} \left(y_{it} \right)$$

- **[Conditional Independence]** Transition probability:

$$\Pr \left(\mathbf{x}_{t+1}, \varepsilon_{i,t+1} \middle| \mathbf{x}_t, \varepsilon_{it}, \mathbf{y}_t \right) = \Gamma \left(\mathbf{x}_{t+1} \middle| \mathbf{x}_t, \mathbf{y}_t \right) \ G \left(\varepsilon_{i,t+1} \right)$$

Example: Markov Perfect Equilibrium (MPE)

- A MPE is $\{\sigma_i(\mathbf{x}_t, \varepsilon_{it}) : i = 1, 2, ..., N\}$ such that $\sigma_i(\mathbf{x}_t, \varepsilon_{it}) =$

$$\arg\max_{y_{it}} \ \mathbb{E}_{\varepsilon_{-it}} \left[\begin{array}{c} u_i\left(y_{it}, \sigma_{-i}\left(\mathbf{x}_t, \varepsilon_{-it}\right), \mathbf{x}_t\right) + \varepsilon_{it}\left(y_{it}\right) + \\ \beta \sum_{\mathbf{x}_{t+1}} V_i^{\sigma}\left(\mathbf{x}_{t+1}\right) \ \Gamma\left(\mathbf{x}_{t+1} | \mathbf{x}_t, y_{it}, \sigma_{-i}\left(\mathbf{x}_t, \varepsilon_{-it}\right)\right) \end{array} \right]$$

- And – given other players' strategy functions – every player value function V_i^{σ} is the unique solution to the Bellman equation:

$$V_i^{\sigma}\left(\mathbf{x}_t\right) = \max_{y_{it}} \ \mathbb{E}_{\varepsilon_{-it}} \left[\begin{array}{c} u_i\left(y_{it}, \sigma_{-i}\left(\mathbf{x}_t, \varepsilon_{-it}\right), \mathbf{x}_t\right) + \varepsilon_{it}\left(y_{it}\right) + \\ \beta \sum_{\mathbf{x}_{t+1}} V_i^{\sigma}\left(\mathbf{x}_{t+1}\right) \ \Gamma\left(\mathbf{x}_{t+1} | \mathbf{x}_t, y_{it}, \sigma_{-i}\left(\mathbf{x}_t, \varepsilon_{-it}\right)\right) \end{array} \right]$$

Example: Conditional Choice Probabilities

- The strategy function of a player – $\sigma_i(\mathbf{x}_t, \varepsilon_{it})$– can be described using a **choice probability function** (CCP):

$$P_i(y_i|\mathbf{x}_t) = \int 1\{y_i = \sigma_i(\mathbf{x}_t, \varepsilon_{it})\}\ dG(\varepsilon_{it}) = \mathbb{E}_{\varepsilon_{it}}[1\{y_i = \sigma_i(\mathbf{x}_t, \varepsilon_{it})\}]$$

- Under Additive Separability and Conditional Independence there is one-to-one relationship between $P_i(y_i|\mathbf{x}_t)$ and $\sigma_i(\mathbf{x}_t, \varepsilon_{it})$.

- This implies that we can represent a **MPE as a fixed point of a mapping in the space of CCPs**.

- **MPE:** $\{P_i(y_{it}|\mathbf{x}_t) : i = 1, 2, ..., N\}$ such that:

$$P_i(y_{it}|\mathbf{x}_t) = \Psi_i(y_{it}, \mathbf{x}_t; \mathbf{P})$$

$\Psi_i(., .; \mathbf{P})$ is the **best response probability function** for player i.

Example: MPE mapping in CCPs

- The **best response probability functions**:

$$\Psi_i\left(y_{it}, \mathbf{x}_t; \mathbf{P}\right) = \mathbb{E}_{\varepsilon_{it}}\left[1\left\{\begin{array}{c} y_{it} = \arg\max\limits_{y_i \in \{0,1\}} \sum\limits_{y_{-i} \in \{0,1\}^{N-1}} \prod\limits_{j \neq i} P_j\left(y_j | \mathbf{x}_t\right) \\ \left[\begin{array}{c} u_i\left(y_i, \mathbf{y}_{-i}, \mathbf{x}_t\right) + \varepsilon_{it}\left(y_i\right) + \\ +\beta \sum\limits_{\mathbf{x}_{t+1}} V_i^{\mathbf{P}}\left(\mathbf{x}_{t+1}\right) \; \Gamma\left(\mathbf{x}_{t+1} | \mathbf{x}_t, y_i, \mathbf{y}_{-i}\right) \end{array}\right] \end{array}\right.\right.$$

- Given all players' CCP functions – $V_i^{\mathbf{P}}$ is the unique solution to:

$$V_i^{\mathbf{P}}\left(\mathbf{x}_t\right) = \sum_{\mathbf{y} \in \{0,1\}^{N-1}} \prod_{j=1}^{N} P_j\left(y_j | \mathbf{x}_t\right) \left[\begin{array}{c} u_i\left(\mathbf{y}, \mathbf{x}_t\right) + e_i\left(y_i; P_i(\mathbf{x_t})\right) + \\ +\beta \sum\limits_{\mathbf{x}_{t+1}} V_i^{\mathbf{P}}\left(\mathbf{x}_{t+1}\right) \; \Gamma\left(\mathbf{x}_{t+1} | \mathbf{x}_t, \mathbf{y}\right) \end{array}\right]$$

Some challenges in the computation of MPE

- A MPE is a vector **P** with dimension $N \, |\mathcal{Y}|^N \, |\mathcal{X}|$ such that:

$$\mathbf{P} = \Psi(\mathbf{P})$$

- By Brower's Theorem, there exists at least on MPE.

- **[1]** In general, $\Psi(.)$ is **NOT a contraction**. Fixed-point iterations do not guarantee convergence.

- **[2] Multiple equilibria.** In general, $\Psi(.)$ can have many MPE.

- **[3] Curse of dimensionality:** The dimension $|\mathcal{X}|$ increases exponentially with N.

  Example: Game of capital investment. A firm's capital stock can take 10 values. With $N = 6$ firms, we have that $|\mathcal{X}| > 1$ million.

2. Estimation methods

Data and Pseudo (extended) likelihood function

- Data: M local markets where we observe the game over T periods $\{\mathbf{y}_{mt}, \mathbf{x}_{mt} : m = 1, \ldots, M; t = 1, \ldots, T\}$ iid across markets, $M \rightarrow \infty$.

- **Pseudo log-likelihood function**. For any $\boldsymbol{\theta} \in \Theta$ and $\mathbf{P} \in \mathcal{P}$:

$$\hat{Q}\left(\boldsymbol{\theta}, \mathbf{P}\right) \equiv M^{-1} \sum_{m=1}^{M} \sum_{t=1}^{T} \ln\left[\Psi\left(\mathbf{y}_{mt} | \mathbf{x}_{mt}, \boldsymbol{\theta}, \mathbf{P}\right)\right]$$

- Population counterpart: Computing $\mathbb{E}_0\left[\cdot\right]$ with respect to $\mathbf{P}^0$:

$$Q^0\left(\boldsymbol{\theta}, \mathbf{P}\right) \equiv \mathbb{E}_0\left[\sum_{t=1}^{T} \ln\left[\Psi\left(\mathbf{y}_{mt} | \mathbf{x}_{mt}, \boldsymbol{\theta}, \mathbf{P}\right)\right]\right]$$

Maximum Likelihood Estimator (MLE)

$$\left(\widehat{\theta}_{MLE}, \widehat{\mathbf{P}}_{MLE}\right) = \arg\max_{(\theta,\mathbf{P})} \hat{Q}\left(\theta, \mathbf{P}\right) \qquad \text{s.t: } \mathbf{P} = \Psi(\theta, \mathbf{P})$$

- Or equivalently,

$$\left(\widehat{\theta}_{MLE}, \widehat{\mathbf{P}}_{MLE}, \widehat{\lambda}_{MLE}\right) = \arg \max_{(\theta,\mathbf{P},\lambda)} \hat{Q}\left(\theta, \mathbf{P}\right) + \lambda' \left[\mathbf{P} - \Psi(\theta, \mathbf{P})\right]$$

- Or the root $\left(\widehat{\theta}, \widehat{\mathbf{P}}, \widehat{\lambda}\right)$ of the Lagrangian equations:

$$\begin{array}{lll} 1. & \hat{\mathbf{P}} - \mathbf{\Psi}\left(\hat{\theta}, \hat{\mathbf{P}}\right) & = \mathbf{0} \\ 2. & \nabla_{\theta}\hat{Q}\left(\hat{\theta}, \hat{\mathbf{P}}\right) - \lambda' \nabla_{\theta}\mathbf{\Psi}\left(\hat{\theta}, \hat{\mathbf{P}}_{\text{ML}}\right) & = \mathbf{0} \\ 3. & \nabla_{\mathbf{P}}\hat{Q}\left(\hat{\theta}, \hat{\mathbf{P}}\right) - \lambda' \nabla_{\mathbf{P}}\mathbf{\Psi}\left(\hat{\theta}, \hat{\mathbf{P}}\right) & = \mathbf{0} \end{array}$$

that provides the maximum value of $\hat{Q}\left(\theta, \mathbf{P}\right)$.

Computational challenges to obtain MLE

- **Gradient search:**

  - $(\mathbf{P}, \boldsymbol{\lambda})$ have very high dimension.

  - $\hat{Q}(\boldsymbol{\theta}, \mathbf{P})$ is not globally concave in $\mathbf{P}$. Many local maxima / minima.

  - The Jacobian matrix $\nabla_{\mathbf{P}} \boldsymbol{\Psi}$ is very high dimensional. And **it is NOT a sparse matrix !!!**

  - Computing the Jacobian matrix $\nabla_{\mathbf{P}} \boldsymbol{\Psi}$ involves calculating an inverse matrix of the same dimension: complexity $|\mathcal{X}|^3$.

- **Global search** – over the roots of the Lagrangian equations.

Two-step Pseudo MLE

- The challenges of computing the MLE have motivated using much simpler two-step methods.

- **Two-step PML:** $\widehat{\mathbf{P}^0}$ is a nonparametric estimator of $\mathbf{P}^0$, and

$$\widehat{\theta}_{2S} = \hat{\theta}\left(\widehat{\mathbf{P}^0}\right) \equiv \arg\max_{\theta\in\Theta} \hat{Q}\left(\theta, \widehat{\mathbf{P}^0}\right)$$

- **Main advantages:**
  - $\hat{Q}(\theta, \mathbf{P})$ is globally concave in θ.
  - Evaluating $\hat{Q}(\theta, \mathbf{P})$ for multiple θ 's is computationally simple.
  - No computation of $\nabla_{\mathbf{P}}\Psi$; high dim. matrix inversion only once.

- **Main limitations:** Not imposing equilibrium (or even best response) restrictions implies:
  - Cost of efficiency lost and finite sample bias.
  - Restricted forms of unobserved heterogeneity.

Nested Pseudo MLE

- Motivated by challenges of implementing MLE and limitations of two-step, Aguirregabiria & Mira (2002, 2007) propose NPL estimator.
- And **NPL root (or fixed point)**: $\left(\widehat{\theta}, \widehat{\mathbf{P}}\right)$ such that:

$$
\begin{array}{llr}
1. & \hat{\mathbf{P}} - \mathbf{\Psi}\left(\hat{\theta}, \hat{\mathbf{P}}\right) & = \mathbf{0} \\
2. & \hat{\theta} = \arg\max_{\theta \in \Theta} \hat{Q}\left(\theta, \hat{\mathbf{P}}\right) \quad \text{[glob. concave]} \ \nabla_{\theta} \hat{Q}\left(\hat{\theta}, \hat{\mathbf{P}}\right) & = \mathbf{0}
\end{array}
$$

- The **NPL estimator** is the NPL root with the maximum value of $\hat{Q}(\theta, \mathbf{P})$.

Differences ML and NPL estimators

NPL root	ML root
1. $\hat{\mathbf{P}} - \mathbf{\Psi}\left(\hat{\theta}, \hat{\mathbf{P}}\right) = \mathbf{0}$	1. $\hat{\mathbf{P}} = \mathbf{\Psi}\left(\hat{\theta}, \hat{\mathbf{P}}\right)$
2. $\nabla_{\boldsymbol{\theta}} \hat{Q}\left(\hat{\theta}, \hat{\mathbf{P}}\right) = \mathbf{0}$	2. $\nabla_{\boldsymbol{\theta}} \hat{Q}\left(\hat{\theta}, \hat{\mathbf{P}}\right) - \lambda' \nabla_{\boldsymbol{\theta}} \mathbf{\Psi}\left(\hat{\theta}, \hat{\mathbf{P}}\right) = \mathbf{0}$
	3. $\nabla_{\mathbf{P}} \hat{Q}\left(\hat{\theta}, \hat{\mathbf{P}}\right) - \lambda' \nabla_{\mathbf{P}} \mathbf{\Psi}\left(\hat{\theta}, \hat{\mathbf{P}}\right) = \mathbf{0}$

- In **single-agent** DDC models, the two sets of conditions are equivalent [because the zero Jacobian property].

- In **dynamic games**, the two sets of conditions are not equivalent.

Properties of NPL estimator in games

- **Statistical properties.** Under standard regularity conditions, the NPL estimator is:
  - Consistent and asymptotically normal;
  - Not efficient, but numerical experiments show that the efficiency lost is typically small.
  - More efficient than 2-step estimator.

- **"Potential" computational advantages**
  - Because it does not require computing $\nabla_{\mathbf{P}} \mathbf{\Psi}\left(\hat{\theta}, \hat{\mathbf{P}}\right)$, it may have substantial computational gains relative to the implementation of MLE.

- ... but we need to be specific about the algorithms to implement NPL.

3. Algorithms to Implement the NPL Estimator

Fixed Point Algorithm (NPL algorithm)

- Fixed point iterations in sample NPL mapping
- At each iteration k, given $\hat{\mathbf{P}}_{k-1}$:

$$\hat{\boldsymbol{\theta}}_k = \widehat{\theta}(\hat{\mathbf{P}}_{k-1}) \quad \arg\max_{\boldsymbol{\theta}\in\Theta} \hat{Q}\left(\boldsymbol{\theta}, \hat{\mathbf{P}}_{k-1}\right)$$

$$\hat{\mathbf{P}}_k = \qquad \boldsymbol{\Psi}\left(\hat{\boldsymbol{\theta}}_k, \hat{\mathbf{P}}_{k-1}\right)$$

- Remarks:

  - Upon convergence, NPL algorithm delivers an NPL fixed point (not necessarily NPL estimator)

  - Unstable fixed points have singleton domain of attraction

Spectral Algorithm: Motivation

- NPL estimator is a solution to a stochastic system of nonlinear equations:

$$\hat{\boldsymbol{\phi}}(\mathbf{P}) = 0$$

where $\hat{\boldsymbol{\phi}}(\mathbf{P}) \equiv \mathbf{P} - \boldsymbol{\Psi}\left(\widehat{\theta}(\mathbf{P}), \mathbf{P}\right)$.

- Another alternative: quasi-Newton methods
  - Local convergence guaranteed despite fixed point instability
  - Approximation and inversion of high-dimensional Jacobians

- Derivative free non-monotone spectral residual methods
  - Initially proposed by Barzilai and Borwein (1988)
  - Applicable to high-dimensional settings
  - La Cruz, Martinez, and Raydan (2006) propose non-monotone line search as a globalization strategy

Spectral Algorithm

- Define:

$$\Delta\hat{\mathbf{P}}_k \equiv \hat{\mathbf{P}}_k - \hat{\mathbf{P}}_{k-1}$$
$$\Delta\hat{\boldsymbol{\phi}}\left(\hat{\mathbf{P}}_k\right) \equiv \hat{\boldsymbol{\phi}}\left(\hat{\mathbf{P}}_k\right) - \hat{\boldsymbol{\phi}}\left(\hat{\mathbf{P}}_{k-1}\right)$$

- Spectral algorithm iteration:

$$\hat{\mathbf{P}}_{k+1} = \hat{\mathbf{P}}_k - \alpha_k \hat{\boldsymbol{\phi}}\left(\hat{\mathbf{P}}_k\right)$$

with

$$\alpha_k = \frac{\Delta\hat{\mathbf{P}}_k' \; \Delta\hat{\mathbf{P}}_k}{\Delta\hat{\mathbf{P}}_k' \; \Delta\hat{\boldsymbol{\phi}}\left(\hat{\mathbf{P}}_k\right)}$$

- The spectral step-length α_k is a scalar!

4. Convergence of Algorithms and Asymptotic Properties of Estimators

Stable NPL fixed points

- Let $\rho\left(\mathbf{A}\right)$ be the spectral radius of matrix $\mathbf{A}$
  - Maximum absolute value of $\mathbf{A}$'s eigenvalues

- NPL algorithm iterated when $M \rightarrow \infty$, i.e. Kasahara and Shimotsu (2012):
  - $\rho\left(\nabla\boldsymbol{\varphi}^0_{(\mathbf{P}^0)}\right) < 1$ implies stable NPL fixed point in population
  - As $M \rightarrow \infty$, $\nabla\hat{\boldsymbol{\varphi}}_{(\mathbf{P})}$ is well approximated by $\nabla\boldsymbol{\varphi}^0_{(\mathbf{P})}$
  - Expect convergence to NPL estimator if $\rho^0 \equiv \rho\left(\nabla\boldsymbol{\varphi}^0_{(\mathbf{P}^0)}\right) < 1$

- NPL algorithm iterated with in finite M:
  - $\rho\left(\nabla\boldsymbol{\varphi}^0_{(\mathbf{P}^0)}\right) < 1$ is not sufficient for convergence to NPL estimator
  - Lemma 2: convergence requires $\hat{\rho}_{\mathrm{NPL}} \equiv \rho\left(\nabla\hat{\boldsymbol{\varphi}}_{(\hat{\mathbf{P}}_{\mathrm{NPL}})}\right) < 1$

Convergence is stochastic

Lemma 3 (Distribution of sample spectral radius)

As $M \to \infty$, $\sqrt{M}\left(\hat{\rho}_{\mathrm{NPL}} - \rho^0\right) \xrightarrow{d} \mathbf{G}^0\mathbf{Z}\left(\mathbf{P}^0\right) + \mathbf{J}^0\mathbf{L}\left(\mathbf{P}^0\right)$ where $\mathbf{G}^0$ and $\mathbf{J}^0$ are deterministic matrices; $\mathbf{Z}\left(\mathbf{P}^0\right)$ and $\mathbf{L}\left(\mathbf{P}^0\right)$ are the vectors of normal random variables. Therefore:

$$\Pr\left(\hat{\rho}_{\mathrm{NPL}} < 1\right) = \Phi\left(\frac{\sqrt{M}\left[1 - \rho^0\right]}{\sqrt{\mathrm{Var}\left[\hat{\rho}_{\mathrm{NPL}}\right]}}\right)$$

- Positive probability of no convergence for any ρ^0
- As $M \to \infty$:
  - $\Pr\left(\hat{\rho}_{\mathrm{NPL}} < 1\right) \to 1$ if $\rho^0 < 1$
  - $\Pr\left(\hat{\rho}_{\mathrm{NPL}} < 1\right) \to 0$ if $\rho^0 \geqslant 1$

Scenarios for NPL algorithm iterated in fixed samples

- NPL algorithm initiated at a finite set of starting values

(i) NPL algorithm converges and $\hat{\rho}_{\mathrm{NPL}} < 1$
  - For some starting value, $\lim_{K\to\infty} \hat{\boldsymbol{\theta}}_K = \hat{\boldsymbol{\theta}}_{\mathrm{NPL}}$
  - Convergence to NPL estimator

(ii) NPL algorithm converges and $\hat{\rho}_{\mathrm{NPL}} \geqslant 1$
  - $\lim_{K\to\infty} \hat{\boldsymbol{\theta}}_K = \hat{\boldsymbol{\theta}}_* \neq \hat{\boldsymbol{\theta}}_{\mathrm{NPL}}$
  - Convergence to inconsistent estimator

(iii) NPL algorithm fails to converge for all starting values
  - Estimator not well defined

Convergence to inconsistent estimator

Proposition 2 (Convergence to an inconsistent estimator)

Suppose NPL algorithm always converges to some $\hat{\boldsymbol{\theta}}_*$ whenever $\hat{\rho}_{\mathrm{NPL}} \geqslant 1$ such that $\sqrt{M}\left(\hat{\boldsymbol{\theta}}_* - \hat{\boldsymbol{\theta}}_{\mathrm{NPL}}\right) \xrightarrow{d} \mathbf{W}_*$. Then, $\forall K \geqslant \bar{K}$ for large enough $\bar{K}$,
$\hat{\boldsymbol{\theta}}_K = \hat{\boldsymbol{\theta}}_{\mathrm{NPL}} + \mathbb{1}\left\{\hat{\rho}_{\mathrm{NPL}} \geqslant 1\right\}\left[\hat{\boldsymbol{\theta}}_* - \hat{\boldsymbol{\theta}}_{\mathrm{NPL}}\right]$ and:

$$\sqrt{M}\left(\hat{\boldsymbol{\theta}}_K - \boldsymbol{\theta}^0\right) \xrightarrow{d} \sqrt{M}\left(\hat{\boldsymbol{\theta}}_{\mathrm{NPL}} - \boldsymbol{\theta}^0\right) + \mathbf{W}_*\left[1 - \Phi\left(\frac{\sqrt{M}\left[1-\rho^0\right]}{\sqrt{\mathrm{Var}\left[\hat{\rho}_{\mathrm{NPL}}\right]}}\right)\right]$$

- The resulting asymptotic distribution is a mixture of:
  - the NPL estimator; and
  - the difference between the inconsistent and NPL estimators

Convergence selection

Proposition 3 (Convergence selection)

Suppose NPL algorithm never converges whenever $\hat{\rho}_{\text{NPL}} \geqslant 1$. Then, $\forall K \geqslant \bar{K}$ for large enough $\bar{K}$, well-defined NPL algorithm estimators are such that:

$$\sqrt{M}\left(\hat{\boldsymbol{\theta}}_K - \boldsymbol{\theta}^0\right) \xrightarrow{d} \sqrt{M}\left(\hat{\boldsymbol{\theta}}_{\text{NPL}} - \boldsymbol{\theta}^0\right) \times \mathbb{1}\left\{\sqrt{M}\left(\hat{\rho}_{\text{NPL}} - \rho^0\right) < \sqrt{M}\left(1 - \rho^0\right)\right\}$$

- The resulting asymptotic distribution is:
  - a truncated version of the NPL estimator's distribution
  - with truncation determine by $\hat{\rho}_{\text{NPL}}$ and ρ^0

Findings and implications

1. $\rho^0 < 1$ is not sufficient for convergence when the iterative procedure is implemented if finite samples
2. Convergence to inconsistent estimator and convergence selection affects asymptotic distribution of NPL algorithm estimator
3. Convergence issues more relevant in cases close to stability cut-off
4. Randomness in sample NPL mapping less concerning when $M \to \infty$

5. Monte Carlo Experiments

Data generating process

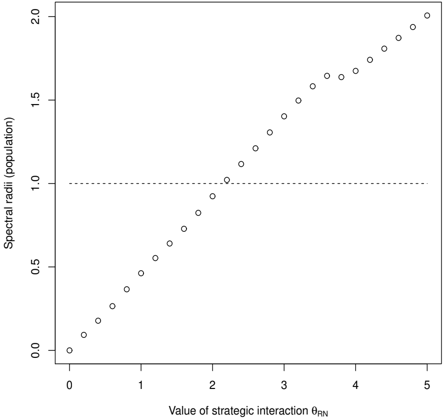
- Game of market entry and exit between $N = 5$ firms
  - Based on Experiment 3 in Aguirregabiria and Mira (2007)
  - Also Kasahara and Shimotsu (2012); Egesdal, Lai, and Su (2015)

- Current profits:

$$U_{imt} = \begin{cases} \theta_{\text{RS}} S_{mt} - \theta_{\text{RN}} \ln \left[1 + \sum_{j \neq i} y_{jmt} \right] & \\ \qquad -\theta_{\text{EC}} \left(1 - y_{im,t-1}\right) - \theta_{\text{FC},i} + \varepsilon_{imt}(1) & \text{if } y_{imt} = 1 \\ \varepsilon_{imt}(0) & \text{if } y_{imt} = 0 \end{cases}$$

- - 160 different states
  - $M \in \{400, 5000\}$
  - $\theta_{\text{RN}} \in \{1,\ 2,\ 2.4,\ 4\}$ for different levels of stability

Grid of strategic interactions



- Stability condition holds for smaller θ_{RN}

Competition statistics

	Very stable $\theta_{\mathrm{RN}} = 1$	**Mildly stable** $\theta_{\mathrm{RN}} = 2$	**Mildly unstable** $\theta_{\mathrm{RN}} = 2.4$	**Very unstable** $\theta_{\mathrm{RN}} = 4$
Number of active firms				
Average	2.7652	1.9939	1.8451	1.2225
Std. dev.	1.6622	1.4320	1.3635	1.0024
Average number of entries	0.6917	0.7473	0.7380	0.5492
Average number of exits	0.6933	0.7569	0.7448	0.5558
Probabilities of being active				
Firm 1	0.4993	0.3222	0.2866	0.1239
Firm 2	0.5222	0.3552	0.3219	0.1457
Firm 3	0.5536	0.3975	0.3662	0.1871
Firm 4	0.5797	0.4363	0.4098	0.2689
Firm 5	0.6103	0.4827	0.4606	0.4968

Notes: Statistics computed using $M = 50000$ markets drawn from the ergodic distribution of the state variables.

- As θ_{RN} increases:
  - The number of active firms decreases
  - The probabilities of being active decrease

Competition statistics

	Very stable $\theta_{\mathrm{RN}} = 1$	Mildly stable $\theta_{\mathrm{RN}} = 2$	Mildly unstable $\theta_{\mathrm{RN}} = 2.4$	Very unstable $\theta_{\mathrm{RN}} = 4$
Number of active firms				
Average	2.7652	1.9939	1.8451	1.2225
Std. dev.	1.6622	1.4320	1.3635	1.0024
Average number of entries	0.6917	0.7473	0.7380	0.5492
Average number of exits	0.6933	0.7569	0.7448	0.5558
Probabilities of being active				
Firm 1	0.4993	0.3222	0.2866	0.1239
Firm 2	0.5222	0.3552	0.3219	0.1457
Firm 3	0.5536	0.3975	0.3662	0.1871
Firm 4	0.5797	0.4363	0.4098	0.2689
Firm 5	0.6103	0.4827	0.4606	0.4968

Notes: Statistics computed using $M = 50000$ markets drawn from the ergodic distribution of the state variables.

- As θ_{RN} increases:
  - The number of active firms decreases
  - The probabilities of being active decrease

Competition statistics

	Very stable $\theta_{\mathrm{RN}} = 1$	Mildly stable $\theta_{\mathrm{RN}} = 2$	Mildly unstable $\theta_{\mathrm{RN}} = 2.4$	Very unstable $\theta_{\mathrm{RN}} = 4$
Number of active firms				
Average	2.7652	1.9939	1.8451	1.2225
Std. dev.	1.6622	1.4320	1.3635	1.0024
Average number of entries	0.6917	0.7473	0.7380	0.5492
Average number of exits	0.6933	0.7569	0.7448	0.5558
Probabilities of being active				
Firm 1	0.4993	0.3222	0.2866	0.1239
Firm 2	0.5222	0.3552	0.3219	0.1457
Firm 3	0.5536	0.3975	0.3662	0.1871
Firm 4	0.5797	0.4363	0.4098	0.2689
Firm 5	0.6103	0.4827	0.4606	0.4968

Notes: Statistics computed using $M = 50000$ markets drawn from the ergodic distribution of the state variables.

- As θ_{RN} increases:
  - The number of active firms decreases
  - The probabilities of being active decrease

Convergence to the NPL estimator

- NPL estimates:
  - Use NPL and spectral algorithms to find NPL fixed points
  - Treat fixed point that maximizes log-likelihood as NPL estimates
  - Tolerance: within 10^{-5} of the NPL estimates' log-likelihood

	Very stable		**Mildly stable**		**Mildly unstable**		**Very unstable**	
	400	5K	400	5K	400	5K	400	5K
% NPL algorithm = NPL estimator	92.4	99.6	57.6	74.0	43.6	29.2	3.2	0.0
% Spectral solver = NPL estimator	100.0	99.8	100.0	99.6	99.8	100.0	99.6	99.6
% Both = NPL estimator	92.4	99.6	57.6	74.0	43.6	29.2	3.2	0.0

Notes: Percentages computed over 500 Monte Carlo samples.

- NPL algorithm often converges to NPL estimator in stable cases
- Spectral algorithm is NPL estimator upon convergence

Convergence to the NPL estimator

- NPL estimates:
  - Use NPL and spectral algorithms to find NPL fixed points
  - Treat fixed point that maximizes log-likelihood as NPL estimates
  - Tolerance: within 10^{-5} of the NPL estimates' log-likelihood

	Very stable		**Mildly stable**		**Mildly unstable**		**Very unstable**	
	400	5K	400	5K	400	5K	400	5K
% NPL algorithm = NPL estimator	92.4	99.6	57.6	74.0	43.6	29.2	3.2	0.0
% Spectral solver = NPL estimator	100.0	99.8	100.0	99.6	99.8	100.0	99.6	99.6
% Both = NPL estimator	92.4	99.6	57.6	74.0	43.6	29.2	3.2	0.0

Notes: Percentages computed over 500 Monte Carlo samples.

- NPL algorithm often converges to NPL estimator in stable cases
- Spectral algorithm is NPL estimator upon convergence

Convergence to the NPL estimator

- NPL estimates:
  - Use NPL and spectral algorithms to find NPL fixed points
  - Treat fixed point that maximizes log-likelihood as NPL estimates
  - Tolerance: within 10^{-5} of the NPL estimates' log-likelihood

	Very stable		**Mildly stable**		**Mildly unstable**		**Very unstable**	
	400	5K	400	5K	400	5K	400	5K
% NPL algorithm = NPL estimator	92.4	99.6	57.6	74.0	43.6	29.2	3.2	0.0
% Spectral solver = NPL estimator	100.0	99.8	100.0	99.6	99.8	100.0	99.6	99.6
% Both = NPL estimator	92.4	99.6	57.6	74.0	43.6	29.2	3.2	0.0

Notes: Percentages computed over 500 Monte Carlo samples.

- NPL algorithm often converges to NPL estimator in stable cases
- Spectral algorithm is NPL estimator upon convergence

Sample stability and convergence

	Very stable		Mildly stable		Mildly unstable		Very unstable	
	400	5K	400	5K	400	5K	400	5K
Spectral radius < 1	95.8	99.6	63.8	88.2	50.0	47.0	4.6	0.0
% NPL algo converged	92.4	99.4	57.2	73.6	43.4	29.2	3.2	0.0
% Spectral algo converged	95.2	99.6	63.6	87.6	49.6	46.8	4.6	0.0
Spectral radius $\geqslant 1$	4.2	0.2	36.2	11.4	49.8	53.0	95.0	99.6
% NPL algo converged	0.0	0.2	0.4	0.4	0.2	0.0	0.0	0.0
% Spectral algo converged	4.2	0.2	35.2	10.8	49.8	52.8	94.8	99.2
Spectral solver did not converge	0.0	0.2	0.0	0.4	0.2	0.0	0.4	0.4

Notes: Percentages computed over 500 Monte Carlo samples. "Spectral radius" refers to $\hat{\rho}_{\mathrm{NPL}}$. NPL and spectral algorithms converged if $\max\left\{\left|\hat{\boldsymbol{\theta}}_{100} - \hat{\boldsymbol{\theta}}_{99}\right|\right\} < 10^{-5}$.

- As data generating process is less stable:
  - % of samples NPL algorithm converges decreases
  - % of samples with $\hat{\rho}_{\mathrm{NPL}} < 1$ decreases

- NPL algorithm almost never converges when $\hat{\rho}_{\mathrm{NPL}} \geqslant 1$
  - Convergence to inconsistent estimator not relevant here
  - Obvious convergence selection

Sample stability and convergence

	Very stable		Mildly stable		Mildly unstable		Very unstable	
	400	5K	400	5K	400	5K	400	5K
Spectral radius < 1	95.8	99.6	63.8	88.2	50.0	47.0	4.6	0.0
% NPL algo converged	92.4	99.4	57.2	73.6	43.4	29.2	3.2	0.0
% Spectral algo converged	95.2	99.6	63.6	87.6	49.6	46.8	4.6	0.0
Spectral radius $\geqslant 1$	4.2	0.2	36.2	11.4	49.8	53.0	95.0	99.6
% NPL algo converged	0.0	0.2	0.4	0.4	0.2	0.0	0.0	0.0
% Spectral algo converged	4.2	0.2	35.2	10.8	49.8	52.8	94.8	99.2
Spectral solver did not converge	0.0	0.2	0.0	0.4	0.2	0.0	0.4	0.4

Notes: Percentages computed over 500 Monte Carlo samples. “Spectral radius” refers to $\hat{\rho}_{\mathrm{NPL}}$. NPL and spectral algorithms converged if $\max\left\{\left|\hat{\boldsymbol{\theta}}_{100} - \hat{\boldsymbol{\theta}}_{99}\right|\right\} < 10^{-5}$.

- As data generating process is less stable:
  - % of samples NPL algorithm converges decreases
  - % of samples with $\hat{\rho}_{\mathrm{NPL}} < 1$ decreases

- NPL algorithm almost never converges when $\hat{\rho}_{\mathrm{NPL}} \geqslant 1$
  - Convergence to inconsistent estimator not relevant here
  - Obvious convergence selection

Sample stability and convergence

	Very stable		Mildly stable		Mildly unstable		Very unstable	
	400	5K	400	5K	400	5K	400	5K
Spectral radius < 1	95.8	99.6	63.8	88.2	50.0	47.0	4.6	0.0
% NPL algo converged	92.4	99.4	57.2	73.6	43.4	29.2	3.2	0.0
% Spectral algo converged	95.2	99.6	63.6	87.6	49.6	46.8	4.6	0.0
Spectral radius $\geqslant 1$	4.2	0.2	36.2	11.4	49.8	53.0	95.0	99.6
% NPL algo converged	0.0	0.2	0.4	0.4	0.2	0.0	0.0	0.0
% Spectral algo converged	4.2	0.2	35.2	10.8	49.8	52.8	94.8	99.2
Spectral solver did not converge	0.0	0.2	0.0	0.4	0.2	0.0	0.4	0.4

Notes: Percentages computed over 500 Monte Carlo samples. "Spectral radius" refers to $\hat{\rho}_{\mathrm{NPL}}$. NPL and spectral algorithms converged if $\max\left\{\left|\hat{\boldsymbol{\theta}}_{100} - \hat{\boldsymbol{\theta}}_{99}\right|\right\} < 10^{-5}$.

- As data generating process is less stable:
  - % of samples NPL algorithm converges decreases
  - % of samples with $\hat{\rho}_{\mathrm{NPL}} < 1$ decreases

- NPL algorithm almost never converges when $\hat{\rho}_{\mathrm{NPL}} \geqslant 1$
  - Convergence to inconsistent estimator not relevant here
  - Obvious convergence selection

Sample stability and convergence

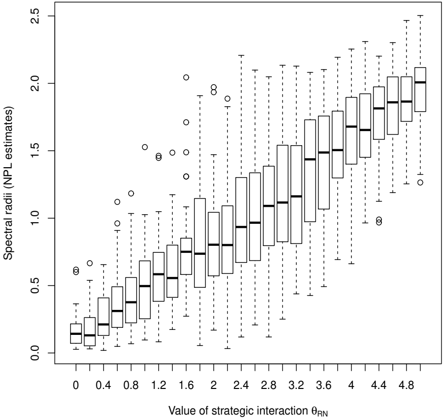
	Very stable		Mildly stable		Mildly unstable		Very unstable	
	400	5K	400	5K	400	5K	400	5K
Spectral radius < 1	95.8	99.6	63.8	88.2	50.0	47.0	4.6	0.0
% NPL algo converged	92.4	99.4	57.2	73.6	43.4	29.2	3.2	0.0
% Spectral algo converged	95.2	99.6	63.6	87.6	49.6	46.8	4.6	0.0
Spectral radius $\geqslant 1$	4.2	0.2	36.2	11.4	49.8	53.0	95.0	99.6
% NPL algo converged	0.0	0.2	0.4	0.4	0.2	0.0	0.0	0.0
% Spectral algo converged	4.2	0.2	35.2	10.8	49.8	52.8	94.8	99.2
Spectral solver did not converge	0.0	0.2	0.0	0.4	0.2	0.0	0.4	0.4

Notes: Percentages computed over 500 Monte Carlo samples. "Spectral radius" refers to $\hat{\rho}_{\mathrm{NPL}}$. NPL and spectral algorithms converged if $\max\left\{\left|\hat{\boldsymbol{\theta}}_{100} - \hat{\boldsymbol{\theta}}_{99}\right|\right\} < 10^{-5}$.

- As data generating process is less stable:
  - % of samples NPL algorithm converges decreases
  - % of samples with $\hat{\rho}_{\mathrm{NPL}} < 1$ decreases

- NPL algorithm almost never converges when $\hat{\rho}_{\mathrm{NPL}} \geqslant 1$
  - Convergence to inconsistent estimator not relevant here
  - Obvious convergence selection

Boxplots of $\hat{\rho}_{\mathrm{NPL}}$



- Large fraction of $\hat{\rho}_{\mathrm{NPL}}$ on both sides of 1 when close to the stability cut-off

Averages and standard errors – $M = 400$

	$\theta_{\mathrm{RS}} = 1$	θ_{RN}	$\theta_{\mathrm{EC}} = 1$	$\theta_{\mathrm{FC},1} = 1.9$
	Converged $K = 100$ NPL algorithm estimates			
Very stable ($\theta_{\mathrm{RN}} = 1$)	1.0038	0.9891	1.0047	1.9186
	(0.1949)	(0.5923)	(0.1156)	(0.2250)
Mildly stable ($\theta_{\mathrm{RN}} = 2$)	0.8339	1.3737	1.0547	1.9126
	(0.1503)	(0.5272)	(0.1094)	(0.2081)
Mildly unstable ($\theta_{\mathrm{RN}} = 2.4$)	0.7677	1.4587	1.0974	1.9062
	(0.1246)	(0.4646)	(0.1132)	(0.2113)
Very unstable ($\theta_{\mathrm{RN}} = 4$)	0.6444	1.8959	1.3416	2.0073
	(0.0930)	(0.2967)	(0.1131)	(0.2946)
	Spectral solver estimates			
Very stable ($\theta_{\mathrm{RN}} = 1$)	1.0416	1.1071	0.9973	1.9043
	(0.2377)	(0.7264)	(0.1204)	(0.2328)
Mildly Stable ($\theta_{\mathrm{RN}} = 2$)	1.0308	2.0941	0.9869	1.9209
	(0.2922)	(1.0461)	(0.1383)	(0.2300)
Mildly Unstable ($\theta_{\mathrm{RN}} = 2.4$)	1.0135	2.4331	0.9953	1.9228
	(0.2725)	(1.0539)	(0.1510)	(0.2418)
Very unstable ($\theta_{\mathrm{RN}} = 4$)	0.9813	3.8541	1.0320	1.9315
	(0.1592)	(0.7964)	(0.1796)	(0.3031)

Notes: Averages and standard errors (in brackets) computed over 500 Monte Carlo samples.

- Consequences of convergence selection on θ_{RN}:
  - NPL algorithm estimates more biased (attenuation) in unstable cases
  - Spectral estimates have larger variances

Averages and standard errors – $M = 400$

	$\theta_{\mathrm{RS}} = 1$	θ_{RN}	$\theta_{\mathrm{EC}} = 1$	$\theta_{\mathrm{FC},1} = 1.9$
	Converged $K = 100$ NPL algorithm estimates			
Very stable ($\theta_{\mathrm{RN}} = 1$)	1.0038	0.9891	1.0047	1.9186
	(0.1949)	(0.5923)	(0.1156)	(0.2250)
Mildly stable ($\theta_{\mathrm{RN}} = 2$)	0.8339	1.3737	1.0547	1.9126
	(0.1503)	(0.5272)	(0.1094)	(0.2081)
Mildly unstable ($\theta_{\mathrm{RN}} = 2.4$)	0.7677	1.4587	1.0974	1.9062
	(0.1246)	(0.4646)	(0.1132)	(0.2113)
Very unstable ($\theta_{\mathrm{RN}} = 4$)	0.6444	1.8959	1.3416	2.0073
	(0.0930)	(0.2967)	(0.1131)	(0.2946)
	Spectral solver estimates			
Very stable ($\theta_{\mathrm{RN}} = 1$)	1.0416	1.1071	0.9973	1.9043
	(0.2377)	(0.7264)	(0.1204)	(0.2328)
Mildly Stable ($\theta_{\mathrm{RN}} = 2$)	1.0308	2.0941	0.9869	1.9209
	(0.2922)	(1.0461)	(0.1383)	(0.2300)
Mildly Unstable ($\theta_{\mathrm{RN}} = 2.4$)	1.0135	2.4331	0.9953	1.9228
	(0.2725)	(1.0539)	(0.1510)	(0.2418)
Very unstable ($\theta_{\mathrm{RN}} = 4$)	0.9813	3.8541	1.0320	1.9315
	(0.1592)	(0.7964)	(0.1796)	(0.3031)

Notes: Averages and standard errors (in brackets) computed over 500 Monte Carlo samples.

- Consequences of convergence selection on θ_{RN}:
  - NPL algorithm estimates more biased (attenuation) in unstable cases
  - Spectral estimates have larger variances

Averages and standard errors – $M = 400$

	$\theta_{\mathrm{RS}} = 1$	θ_{RN}	$\theta_{\mathrm{EC}} = 1$	$\theta_{\mathrm{FC},1} = 1.9$
	Converged $K = 100$ NPL algorithm estimates			
Very stable ($\theta_{\mathrm{RN}} = 1$)	1.0038	0.9891	1.0047	1.9186
	(0.1949)	(0.5923)	(0.1156)	(0.2250)
Mildly stable ($\theta_{\mathrm{RN}} = 2$)	0.8339	1.3737	1.0547	1.9126
	(0.1503)	(0.5272)	(0.1094)	(0.2081)
Mildly unstable ($\theta_{\mathrm{RN}} = 2.4$)	0.7677	1.4587	1.0974	1.9062
	(0.1246)	(0.4646)	(0.1132)	(0.2113)
Very unstable ($\theta_{\mathrm{RN}} = 4$)	0.6444	1.8959	1.3416	2.0073
	(0.0930)	(0.2967)	(0.1131)	(0.2946)
	Spectral solver estimates			
Very stable ($\theta_{\mathrm{RN}} = 1$)	1.0416	1.1071	0.9973	1.9043
	(0.2377)	(0.7264)	(0.1204)	(0.2328)
Mildly Stable ($\theta_{\mathrm{RN}} = 2$)	1.0308	2.0941	0.9869	1.9209
	(0.2922)	(1.0461)	(0.1383)	(0.2300)
Mildly Unstable ($\theta_{\mathrm{RN}} = 2.4$)	1.0135	2.4331	0.9953	1.9228
	(0.2725)	(1.0539)	(0.1510)	(0.2418)
Very unstable ($\theta_{\mathrm{RN}} = 4$)	0.9813	3.8541	1.0320	1.9315
	(0.1592)	(0.7964)	(0.1796)	(0.3031)

Notes: Averages and standard errors (in brackets) computed over 500 Monte Carlo samples.

- Consequences of convergence selection on θ_{RN}:
  - NPL algorithm estimates more biased (attenuation) in unstable cases
  - Spectral estimates have larger variances

Averages and standard errors – $M = 5000$

	$\theta_{\mathrm{RS}} = 1$	θ_{RN}	$\theta_{\mathrm{EC}} = 1$	$\theta_{\mathrm{FC},1} = 1.9$
	Converged $K = 100$ NPL algorithm estimates			
Very stable ($\theta_{\mathrm{RN}} = 1$)	1.0030	1.0079	1.0008	1.9011
	(0.0659)	(0.2053)	(0.0355)	(0.0662)
Mildly stable ($\theta_{\mathrm{RN}} = 2$)	0.9665	1.8753	1.0143	1.9037
	(0.0658)	(0.2374)	(0.0350)	(0.0634)
Mildly unstable ($\theta_{\mathrm{RN}} = 2.4$)	0.9069	2.0172	1.0402	1.9066
	(0.0408)	(0.1508)	(0.0333)	(0.0650)
Very unstable ($\theta_{\mathrm{RN}} = 4$)	–	–	–	–
	(–)	(–)	(–)	(–)
	Spectral solver estimates			
Very stable ($\theta_{\mathrm{RN}} = 1$)	1.0031	1.0082	1.0008	1.9011
	(0.0659)	(0.2052)	(0.0355)	(0.0661)
Mildly stable ($\theta_{\mathrm{RN}} = 2$)	0.9990	1.9972	1.0029	1.9011
	(0.0854)	(0.3122)	(0.0400)	(0.0661)
Mildly unstable ($\theta_{\mathrm{RN}} = 2.4$)	1.0035	2.4121	0.9991	1.9049
	(0.0886)	(0.3486)	(0.0465)	(0.0708)
Very unstable ($\theta_{\mathrm{RN}} = 4$)	0.9993	3.9918	1.0025	1.9046
	(0.0431)	(0.2131)	(0.0498)	(0.0913)

Notes: Averages and standard errors (in brackets) computed over 500 Monte Carlo samples.

- Spectral solver estimates unbiased regardless level of stability

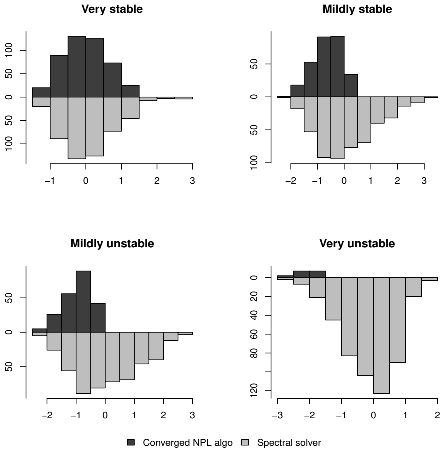
Averages and standard errors – $M = 5000$

	$\theta_{\mathrm{RS}} = 1$	θ_{RN}	$\theta_{\mathrm{EC}} = 1$	$\theta_{\mathrm{FC},1} = 1.9$
	Converged $K = 100$ NPL algorithm estimates			
Very stable ($\theta_{\mathrm{RN}} = 1$)	1.0030	1.0079	1.0008	1.9011
	(0.0659)	(0.2053)	(0.0355)	(0.0662)
Mildly stable ($\theta_{\mathrm{RN}} = 2$)	0.9665	1.8753	1.0143	1.9037
	(0.0658)	(0.2374)	(0.0350)	(0.0634)
Mildly unstable ($\theta_{\mathrm{RN}} = 2.4$)	0.9069	2.0172	1.0402	1.9066
	(0.0408)	(0.1508)	(0.0333)	(0.0650)
Very unstable ($\theta_{\mathrm{RN}} = 4$)	–	–	–	–
	(–)	(–)	(–)	(–)
	Spectral solver estimates			
Very stable ($\theta_{\mathrm{RN}} = 1$)	1.0031	1.0082	1.0008	1.9011
	(0.0659)	(0.2052)	(0.0355)	(0.0661)
Mildly stable ($\theta_{\mathrm{RN}} = 2$)	0.9990	1.9972	1.0029	1.9011
	(0.0854)	(0.3122)	(0.0400)	(0.0661)
Mildly unstable ($\theta_{\mathrm{RN}} = 2.4$)	1.0035	2.4121	0.9991	1.9049
	(0.0886)	(0.3486)	(0.0465)	(0.0708)
Very unstable ($\theta_{\mathrm{RN}} = 4$)	0.9993	3.9918	1.0025	1.9046
	(0.0431)	(0.2131)	(0.0498)	(0.0913)

Notes: Averages and standard errors (in brackets) computed over 500 Monte Carlo samples.

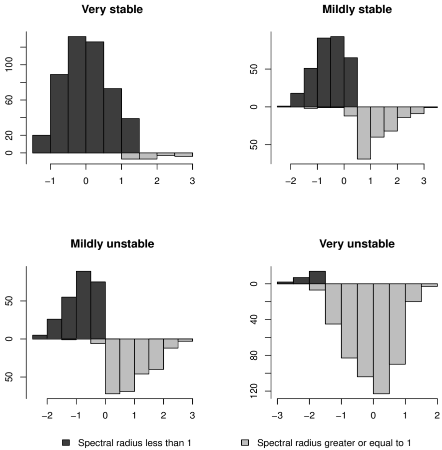
- Spectral solver estimates unbiased regardless level of stability

Converged NPL algorithm vs spectral solver – $M = 400$



- Distribution of NPL algorithm estimates is truncated

Spectral solver by values of $\hat{\rho}_{\mathrm{NPL}}$ – $M = 400$



- $\hat{\rho}_{\mathrm{NPL}} \geqslant 1$ associated with larger estimates of θ_{RN}

6. Conclusions

Sample stability and convergence [4]

- Randomness in the NPL mapping affects properties of NPL algorithm
  - Explains failure to converge in stable populations
  - Introduces convergence selection issues
  - Attenuation of the strategic interactions estimates
- Spectral algorithm to find NPL estimates
  - Impose equilibrium restrictions
  - Local convergence despite instability
  - No Jacobians to approximate and/or invert
  - Dimensionality of problem smaller than MPEC