Lecture 8: Directional dynamic games and Multiplicity of equilibria

Fedor Iskhakov, Australian National University
John Rust, Georgetown University
Bertel Schjerning, University of Copenhagen

2022 Econometric Society Summer School in Dynamic Structural Econometrics

MIT, 15-20 August, 2022

ROAD MAP for the lecture

1. Multiplicity of equlibria and estimation of static games of imcomplete infromation
2. Bertrand pricing and investment game (Leapfrogging game)
   - Collusion of Australian corrugated fibre packaging (CFP) producers
3. Directional dynamic games (DDGs)
   - State recursion algorithm
   - Recursive lexicographical search (RLS) algorithm
   - Full solution for the leapfrogging game
4. Structural estimation of directional dynamic games with Nested RLS algorithm

Estimating discrete-choice games of incomplete information: Simple static examples

Su (Quant Mark Econ, 2014)

Che-Lin Su, 1974 - 2017

Estimating Games of Incomplete Information

Pakes, Ostrovsky and Berry, 2007
Various 2-Step CCP estimators: pseudo-ML, MoM, min χ^2

Bajari, Benkard and Levin, 2007
2-Step minimum distance estimator using equilibrium inequalities

Aguirregabiria and Mira, 2007
Recursive 2-Step CCP estimator

Pesendorfer and Schmidt-Dengler, 2008
2-Step least squares

Kasahara and Shimotsu, 2012
Modified (dampened) NPL

Su, 2013 and Egesdal, Lai and Su, 2015
Constrained optimization (MPEC)

Example: Static Entry Game

- Two firms: a and b
- Actions: each firm has two possible actions:

$$d_a = \begin{cases} 1, & \text{if firm } \underline{\text{a}} \text{ choose to enter the market} \\ 0, & \text{if firm } \underline{\text{a}} \text{ choose not to enter the market} \end{cases} \tag{1}$$

$$d_b = \begin{cases} 1, & \text{if firm } \underline{\text{b}} \text{ choose to enter the market} \\ 0, & \text{if firm } \underline{\text{b}} \text{ choose not to enter the market} \end{cases} \tag{2}$$

Example: Static Entry Game of Incomplete Information

Utility: Ex-post payoff to firms

$$u_a(d_a, d_b, x_a, \epsilon_a) = \begin{cases} [\alpha + d_b * (\beta - \alpha)]x_a + \epsilon_{a1}, & \text{if } d_a = 1 \\ 0 + \epsilon_{a0}, & \text{if } d_a = 0 \end{cases}$$

$$u_b(d_a, d_b, x_a, \epsilon_b) = \begin{cases} [\alpha + d_a * (\beta - \alpha)]x_b + \epsilon_{b1}, & \text{if } d_b = 1 \\ 0 + \epsilon_{b0}, & \text{if } d_b = 0 \end{cases}$$

- (α, β): structural parameters to be estimated
- (x_a, x_b): firms' observed types; **common knowledge**
- $\epsilon_a = (\epsilon_{a0}, \epsilon_{a1}), \epsilon_b = (\epsilon_{b0}, \epsilon_{b1})$: firms' unobserved types, **private information**
- (ϵ_a, ϵ_b) are observed only by each firm, but not by their opponent firm nor by the econometrician

Example: Static Entry Game of Incomplete Information

- Assume the error terms (ϵ_a, ϵ_b) have EV Type I distribution
- A Bayesian Nash equilibrium (p_a, p_b) satisfies

$$
\begin{aligned}
p_a &= \frac{\exp[p_b \beta x_a + (1 - p_b)\alpha x_a]}{1 + \exp[p_b \beta x_a + (1 - p_b)\alpha x_a]} \\
&= \frac{1}{1 + \exp[-\alpha x_a + p_b x_a(\alpha - \beta)]} \\
&\equiv \Psi_a(p_b, x_a; \alpha, \beta)
\end{aligned}
$$

$$
\begin{aligned}
p_b &= \frac{1}{1 + \exp[-\alpha x_b + p_a x_b(\alpha - \beta)]} \\
&\equiv \Psi_b(p_a, x_b; \alpha, \beta)
\end{aligned}
$$

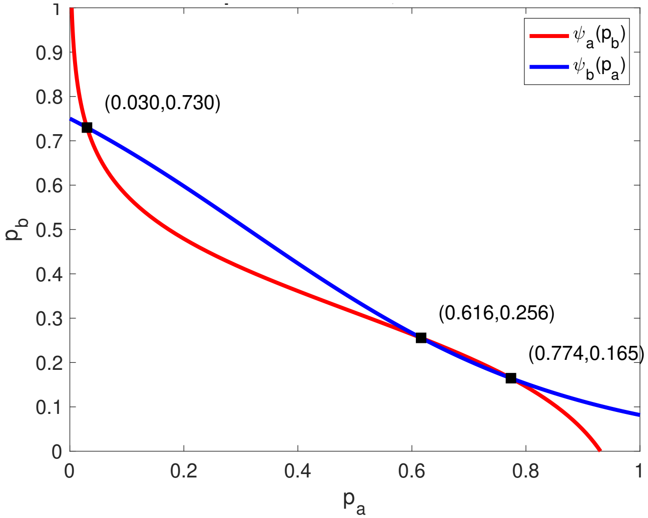
Static Game Example: Parameters

We consider a very contestable game throughout

- Monopoly profits: $\alpha * x_j = 5 * x_j$
- Duopoly profits: $\beta * x_j = -11 * x_j$
- Firm types: $(x_a, x_b) = (0.52, 0.22)$

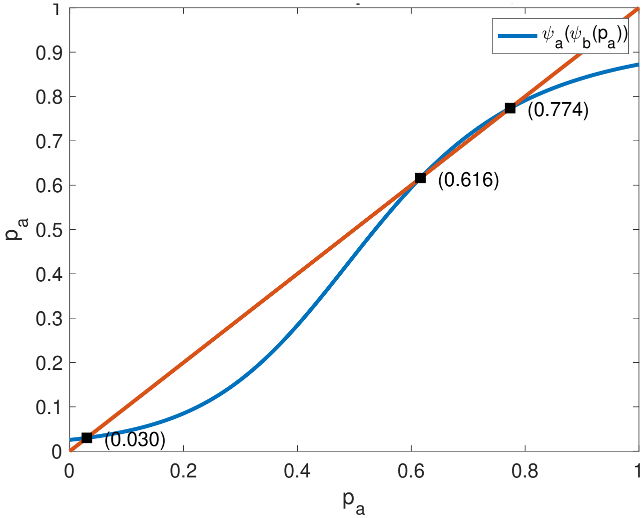
Static Game Example: Three Bayesian Nash Equilibria

Figure: Equilibria at intersections of best response functions



Static Game Example: Solving for Equilibria

Figure: Fixed points on second order best response function



Static Game Example: Solving for Equilibria

Solution method: Combination of succesive approximations and bisection algorithm

Succesive approximations (SA)

- Converge to nearest stable equilibrium.
- Start SA at $p_a = 0$ and $p_a = 1$.
- Unique equilibrium (K=1): SA will converge to it from anywhere.
- Three equilibria (K=3): Two will be stable, and one will be unstable.
- More equilibria (K>3): Not in this model.

Bisection method

- Use this to find the unstable equilibrium (if K=3).
- The bisection method that repeatedly bisects an interval and then selects a subinterval in which the fixed point (or root) must lie.
- The two stable equilibria, defines the initial interval to search over.
- The bisection method is a very simple and robust method, but it is also relatively slow.

Static Game Example: Data Generation and Identification

- Data Generating Process (DGP): the data are generated by a single equilibrium
- The two players use the same equilibrium to play 1000 times
- Data: $X = \{(d_a^i, d_b^i)_{i=1}^{1000}, (x_a, x_b) = (0.52, 0.22)\}$
- Given data X, we want to recover structural parameters α and β

Static Game Example: Maximum Likelihood Estimation

- Maximize the likelihood function

$$
\begin{aligned}
\max_{\alpha,\beta} \quad \log \quad & \mathcal{L}(p_a(\alpha,\beta);X) \\
= \quad & \sum_{i=1}^{N} \left(d_a^i * \log(p_a(\alpha,\beta)) + (1-d_a^i*)\log(1-p_a(\alpha,\beta)) \right) \\
+ \quad & \sum_{i=1}^{N} \left(d_b^i * \log(p_b(\alpha,\beta)) + (1-d_b^i*)\log(1-p_b(\alpha,\beta)) \right)
\end{aligned}
$$

- $p_a(\alpha,\beta)$ and $p_a(\alpha,\beta)$ are the solutions of the Bayesian-Nash Equilibrium equations

$$
\begin{aligned}
p_a &= \frac{1}{1+\exp[-\alpha x_a + p_b x_a(\alpha-\beta)]} \equiv \Psi_a(p_b, x_a; \alpha, \beta) \\
p_b &= \frac{1}{1+\exp[-\alpha x_b + p_a x_b(\alpha-\beta)]} \equiv \Psi_b(p_a, x_b; \alpha, \beta)
\end{aligned}
$$

Static Game Example: MLE via NFXP

- Outer Loop
  - Choose (α, β) to maximize the likelihood function

$$\log \mathcal{L}(p_a(\alpha, \beta), p_b(\alpha, \beta); X)$$

- Inner loop:
  - For a given (α, β), solve the BNE equations for **ALL** equilibria: $(p_a^k(\alpha, \beta), p_b^k(\alpha, \beta)), k = 1, ..., K$
  - Choose the equilibrium that gives the highest likelihood value:

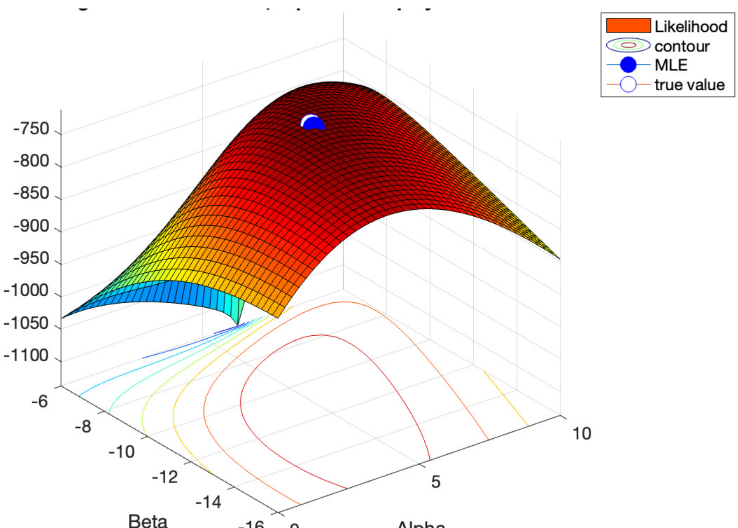
$$k^* = \arg \max_{k=1,\dots,K} \log \mathcal{L}(p_a^k(\alpha, \beta), p_b^k(\alpha, \beta); X)$$

such that

$$(p_a(\alpha, \beta), p_b(\alpha, \beta)) = (p_a^{k*}(\alpha, \beta), p_b^{k*}(\alpha, \beta))$$

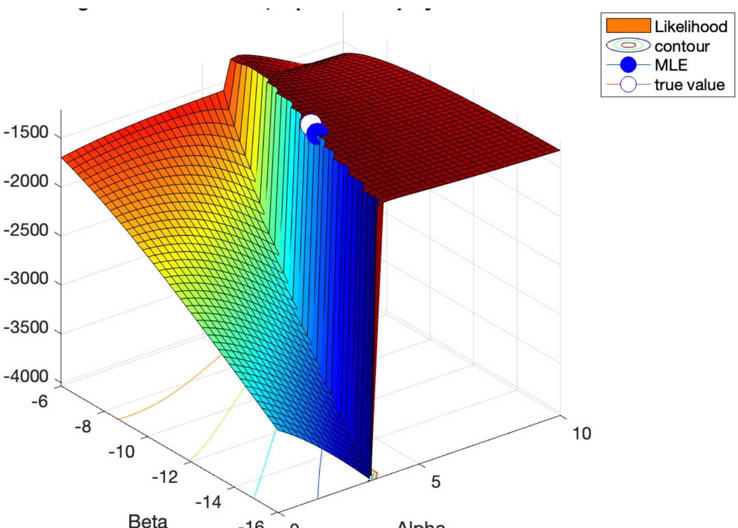
NFXP's Likelihood as a Function of (α, β) - Eq 1

Figure: Data generated from equilibrium 1



NFXP's Likelihood as a Function of (α, β) - Eq 2

Figure: Data generated from equilibrium 2



NFXP's Likelihood as a Function of (α, β) - Eq 3

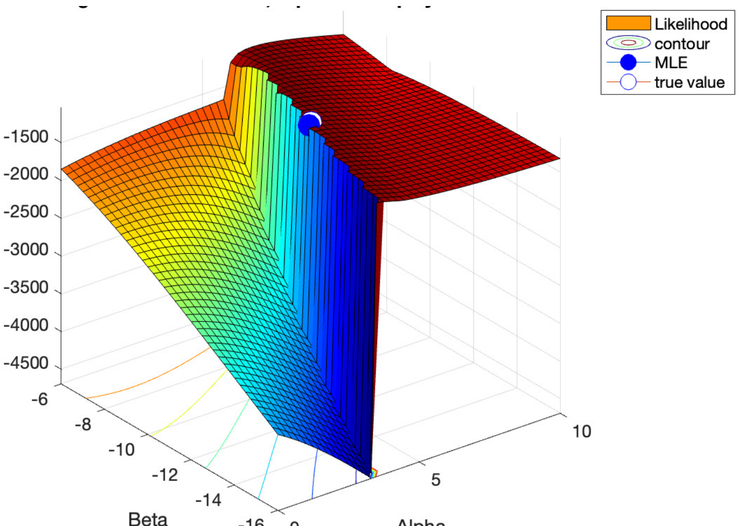


Figure: Data generated from equilibrium 3

Static Game Example: MLE via MPEC

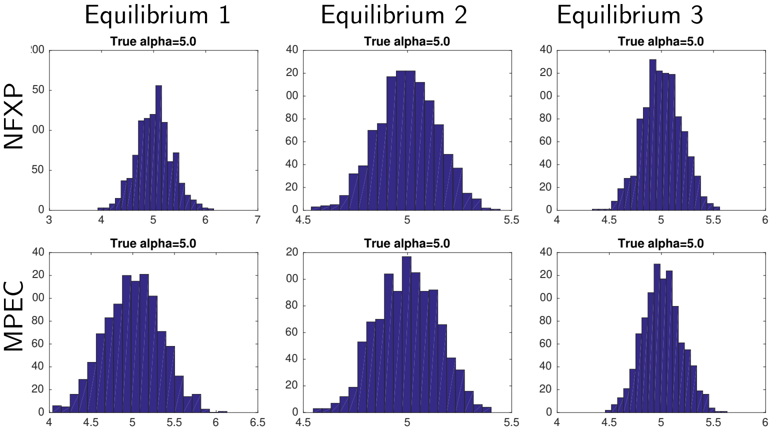
- Maximize the likelihood function

$$
\begin{aligned}
\max_{\alpha,\beta,p_a,p_b} \quad \log \quad & \mathcal{L}(p_a; X) \\
= \quad & \sum_{i=1}^{N} \left(d_a^i * \log(p_a) + (1 - d_a^i *) \log(1 - p_a) \right) \\
+ \quad & \sum_{i=1}^{N} \left(d_b^i * \log(p_b) + (1 - d_b^i *) \log(1 - p_b) \right)
\end{aligned}
$$

- Subject to p_a and p_a are the solutions of the Bayesian-Nash Equilibrium equations

$$
\begin{aligned}
p_a &= \frac{1}{1 + \exp[-\alpha x_a + p_b x_a(\alpha - \beta)]} \\
p_b &= \frac{1}{1 + \exp[-\alpha x_b + p_a x_b(\alpha - \beta)]} \\
0 &\leq p_a, p_b \leq 1
\end{aligned}
$$

Monte Carlo Results



Estimates of parameter α
Data generated form each of the three equilibria

Some remarks

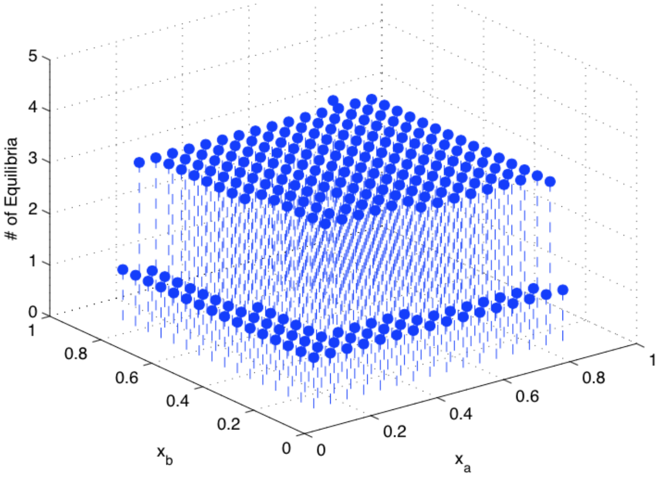
- The likelihood function is discontinuous in α and β for NFXP
- The objective function and constraints for MPEC is smooth in its variables $\theta = (\alpha, \beta, p_a, p_b)$
- Objective function is not differentiable, and not even continuous $\rightarrow$ Standard theorems for inference does not apply.
- This problem is extremely simple. p_a and p_b are scalars. Much harder to solve for p_b and p_b when they are high dimensional solutions to players Bellman equations?
- We cannot find all equilibria by iterating on player's Bellman equations? We may be able to find an equilibrium, but not necessarily the one played in the data.

Estimation with Multiple Markets

- There 25 different markets, i.e., 25 pairs of observed types $(x_a^m, x_b^m), m = 1, ..., 25$
- The grid on x_a has 5 points equally distributed between the interval $[0.12, 0.87]$, and similarly for x_b
- Use the same true parameter values: (α_0, β_0)
- For each market with (x_a^m, x_b^m), solve BNE conditions for (p_a^m, p_b^m).
- There are multiple equilibria in most of 25 markets
- For each market, we (randomly) choose an equilibrium to generate 1000 data points for that market
- The equilibrium used to generate data can be different in different markets - we flip a coin at each market.

of Equilibria with Different (x_a^m, x_b^m)

Figure: Number of equilibria



NFXP - Estimation with Multiple Markets

Inner loop:

$$\max_{\alpha,\beta} \quad \log \mathcal{L}(p_a^m(\alpha,\beta), p_b^m(\alpha,\beta); X)$$

Outer loop: For a given values of (α,β) solve BNE equations for ALL equilibria, $k = 1,..,K$ at each market, $m = 1,..,M$: That is, $p_a^{m,k}(\alpha,\beta)$ and $p_a^{m,k}(\alpha,\beta)$ are the solutions to

$$\begin{aligned} p_a^m &= \Psi_a(p_b^m, x_a^m; \alpha, \beta) \\ p_b^m &= \Psi_b(p_a^m, x_b^m; \alpha, \beta) \\ m &= 1, ..., M \end{aligned}$$

where we again choose the equilibrium, that gives the highest likelihood value at each market m

$$k^* = \arg \max_{k=1,\dots,K} \log \mathcal{L}(p_a^{m,k}(\alpha,\beta), p_b^{m,k}(\alpha,\beta); X)$$

such that

$$(p_a^m(\alpha,\beta), p_b^m(\alpha,\beta)) = (p_a^{m,k*}(\alpha,\beta), p_b^{m,k*}(\alpha,\beta))$$

Estimation with Multiple Markets - MPEC

Constrained optimization formulation

$$\max_{\alpha,\beta,p_a^m,p_b^m} \quad \log \mathcal{L}(p_a^m, p_b^m; X)$$

subject to

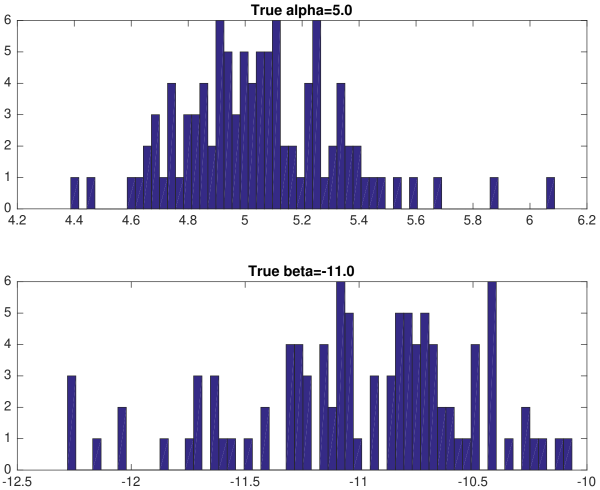
$$\begin{array}{rcl} p_a^m & = & \Psi_a(p_b^m, x_a^m; \alpha, \beta) \\ p_b^m & = & \Psi_b(p_a^m, x_b^m; \alpha, \beta) \\ 0 & \leq & p_a^m, p_b^m \leq 1, \\ m & = & 1, ..., M \end{array}$$

- MPEC does not explicitly solve the BNE equations to find ALL equilibria at each market - for every trial value of parameters.
- But the number of parameters is much larger.
- Both MPEC and NFXP are based on Full Information Maximum Likelihood (FIML) estimators.

NFXP: Monte Carlo - Multiple Markets (M=25, T=50)

Starting values $\alpha_0 = \alpha$, $\beta_0 = \beta$

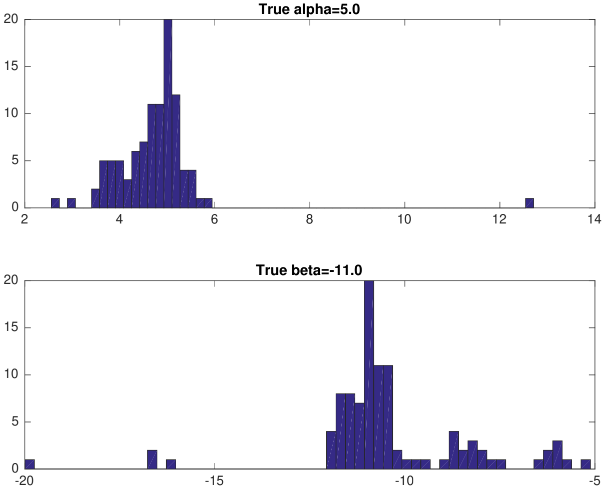
Random equilibrium selection in different markets



MPEC: Monte Carlo - Multiple Markets (M=25, T=50)

Starting values $\alpha_0 = \alpha$, $\beta_0 = \beta$

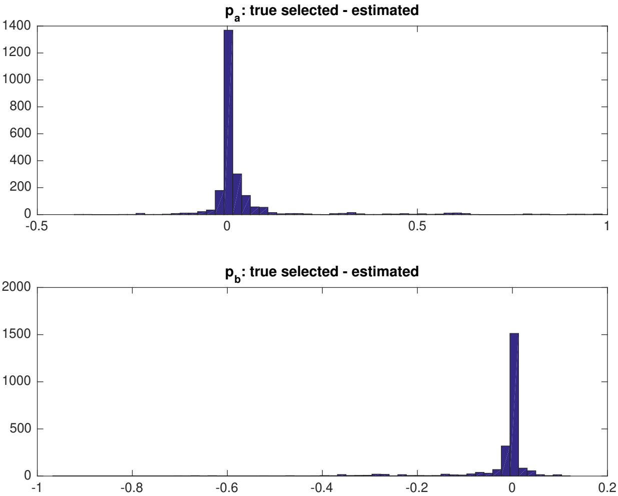
Random equilibrium selection in different markets



MPEC: Monte Carlo - Multiple Markets (M=25, T=50)

Starting values $\alpha_0 = \alpha$, $\beta_0 = \beta$

Random equilibrium selection in different markets



MPEC and NFXP: multiple markets

NFXP:

- 2 parameters in optimization problem
- we can estimate the equilibrium played in the data, $p_a^{m,k*}$ and $p_b^{m,k*}$ (but in models with observationally equivalent equilibria it may not be possible to obtain joint identification of structural parameters and equilibrium probabilities)
- Needs to find ALL equilibria at each market (very hard in more complex problems)
- Good full solution methods required

MPEC:

- $2 + 2M$ parameters in optimization problem
- Does not always converge towards the equilibrium played in the data, although NFXP indicates that $p_a^{m,k*}$ and $p_b^{m,k*}$ are actually identifiable
- Local minima with many markets.
- Disclaimer: Quick and dirty implementation of MPEC. Use AMPL/Knitro

Least Square Estimators

Pesendofer and Schmidt-Dengler (2008)

- Step 1: Estimate $\hat{p} = (\hat{p}_a, \hat{p}_b)$ from the data
- Step 2: Solve

$$\min_{\alpha,\beta} \left\{ (\hat{p}_a - \Psi_a(\hat{p}_b, x_a; \alpha, \beta))^2 + (\hat{p}_b - \Psi_b(\hat{p}_a, x_b; \alpha, \beta); X))^2 \right\}$$

For dynamic games, Markov perfect equilibrium conditions are characterized by

$$p = \Psi(p, \theta)$$

- Step 1: Estimate $\hat{p}$ from the data
- Step 2: Solve

$$\min_{\alpha,\beta} [\hat{p} - \Psi(\hat{p}; \theta)]' W [\hat{p} - \Psi(\hat{p}; \theta)]'$$

2-Step Methods: Pseudo Maximum Likelihood

In 2-step methods

- Step 1: Estimate $\hat{p} = (\hat{p}_a, \hat{p}_b)$ from the data
- Step 2: Solve

$$\max_{\alpha,\beta,p_a,p_b} \quad \log \mathcal{L}(p_a, p_b; X)$$

subject to

$$\begin{aligned} p_a &= \Psi_a(\hat{p}_a, x_a; \alpha, \beta) \\ p_b &= \Psi_b(\hat{p}_b, x_b; \alpha, \beta) \\ 0 &\leq p_a^m, p_b^m \leq 1, m = 1, ..., M \end{aligned}$$

Or equivalently

- Step 1: Estimate $\hat{p} = (\hat{p}_a, \hat{p}_b)$ from the data
- Step 2: Solve

$$\max_{\alpha,\beta} \quad \log \mathcal{L}(\Psi_a(\hat{p}_b, x_a; \alpha, \beta), \Psi_b(\hat{p}_a, x_b; \alpha, \beta); X)$$

Nested Pseudo Likelihood (NPL)

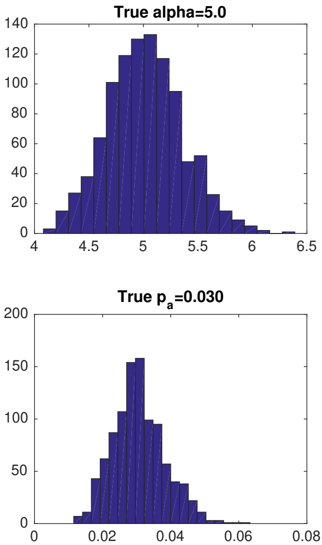
Aguirregabiria and Mira (2007)

NPL iterates on the 2-step methods

1. Step 1: Estimate $\hat{p}^0 = (\hat{p}_a^0, \hat{p}_b^0)$ from the data, set $k = 0$
2. Step 2:
   REPEAT
   2.1 Solve
   $$\alpha^{k+1}, \beta^{k+1} = \arg\max_{\alpha,\beta} \quad \log \mathcal{L}(\Psi_a(\hat{p}_b^k, x_a; \alpha, \beta), \Psi_b(\hat{p}_a^k, x_b; \alpha, \beta); X)$$
   2.2 One best-reply iteration on $\hat{p}^k$
   $$\begin{aligned} \hat{p}_a^{k+1} &= \Psi_a(\hat{p}_a^k, x_a; \alpha^{k+1}, \beta^{k+1}) \\ \hat{p}_a^{k+1} &= \Psi_b(\hat{p}_b^k, x_b; \alpha^{k+1}, \beta^{k+1}) \end{aligned}$$
   2.3 Let k:=k+1;
   UNTIL convergence in (α^k, β^k) and $(\hat{p}_a^k, \hat{p}_b^k)$

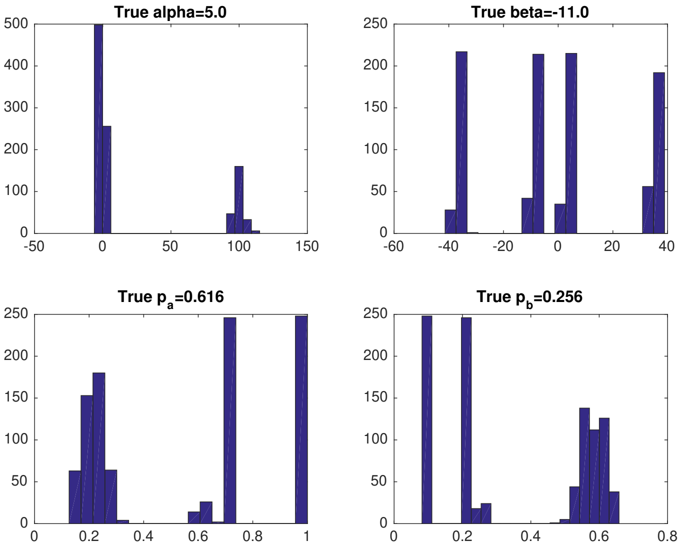
Monte Carlo Results: NPL with Eq 1

Figure: Equilibrium 1 - $\hat{p}_j = 1/N \sum_i I(d_j = 1)$



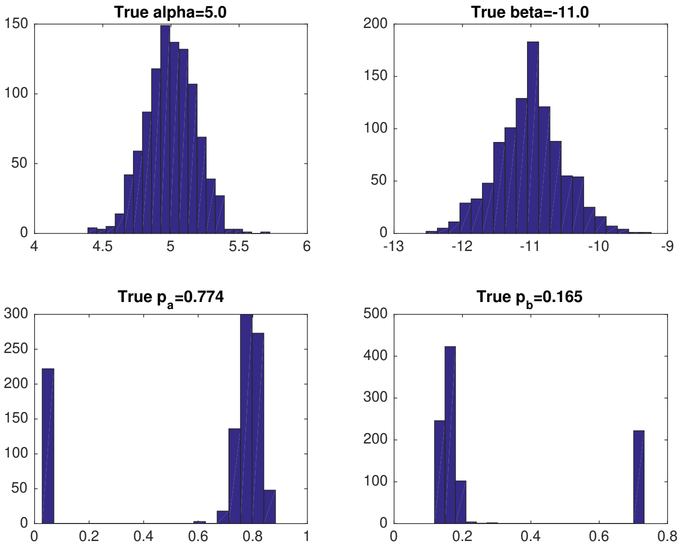
Monte Carlo Results: NPL with Eq 2

Figure: Equilibrium 2 - $\hat{p}_j = 1/N \sum_i I(d_j = 1)$



Monte Carlo Results: NPL with Eq 3

Figure: Equilibrium 3 - $\hat{p}_j = 1/N \sum_i I(d_j = 1)$

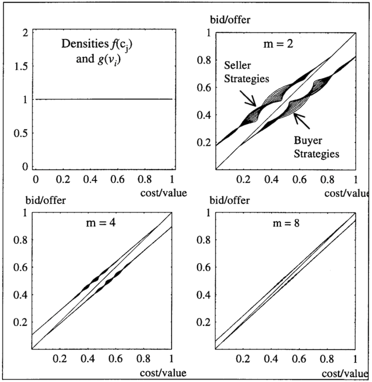


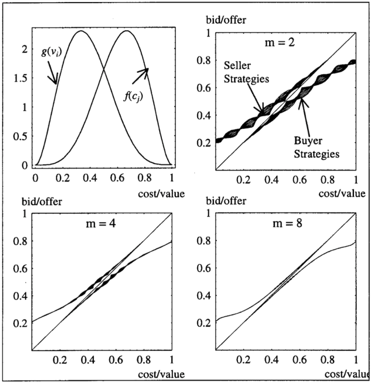
Conclusions

- NFXP/MPEC implementations of MLE is statistically efficient
  - NFXP is computational daunting as we need to compute ALL equilibria for each θ and find maxima of discontinuous likelihood
  - MPEC is computationally faster, but may get stuck in a local minimum at equilibria not played in the data.
- Two step estimators - computationally fast, but inefficient and biased in small samples.
- NPL (Aguirregabiria and Mira 2007) should bridge this gab, but can be unstable when estimating estimating games with multiple equilibria.
- Estimation of dynamic games is an interesting but challenging computational optimization problem
  - Multiple equilibria leads makes likelihood function discontinuous $\rightarrow$ non-standard inference and computational complexity
  - Multiple equilibria leads to indeterminacy problem and identification issues.
- All these problems are amplified by orders of magnitude when we move to Dynamic models

Solving and estimating directional dynamic games: Bertrand pricing and investment (leapfrogging) game

Multiplicity of equilibria in k-Double Auction mechanism





Satterthwaite, Williams, 2002 *Econometrica*
"The Optimality of a Simple Market Mechanism"
(numerical examples and plots in the preceding working paper)

Amcor-Visy collusion case

- Australian market for cardboard is essentially a duopoly
- Between 2000 and 2005 *Visy* and *Amcor* colluded to divide the market of cardboard and to fix prices
- 2007: Visy admits to have been manipulating the market, issued with $36 million fine
- July 2009: Cadbury vs. Amcor, damages estimated at $235.8 million, settles out of court
- March 2011: Class action suit against both Amcor and Visy settles out of court for $95 million

Cardboard industry in Australia

Bertrand price competition in the short run, with leapfrogging investments by both firms in the longer run

- Cardboard is a highly standardized product
- Strong incentives for Bertrand-like price cutting
- Amcor and Visy do minimal amounts of R&D themselves,
- Spend considerable amounts on cost reducing investments
  - Amcor plans to build state-of-the-art paper mill in Botany Bay before the collusion took place
  - "B9" plant finally opened on February 1, 2013
- Amcor and Visy purchase new technology from other companies that specialize in doing the R&D and reduce cost of production of cardboard

Leapfrogging

Leapfrogging equilibrium

- Firms invest in alternating fashion and take turns in cost leadership
- Market price makes permanent downward shifts

"The Bertrand Investment Paradox"

- Should Bertrand competitors undertake cost-reducing investments?
- If both firms acquire state-of-art technology simultaneously, the following Bertrand price competition leads to zero profits for each firm
- Since both firms have access to cost reducing technology, does either of them have any incentive to invest ex ante?

Dynamic Bertrand price competition

Stochastic dynamic game

- Two Bertrand competitors, $n = 2$, no entry or exit
- Discrete time, infinite horizon ($t = 1, 2, \ldots, \infty$)
- Each firm maximizes expected discounted profits, common discount factor $\beta \in (0, 1)$
- Each firm has two choices in each period:
  1. Price for the product
  2. Whether or not to buy the state of the art technology

Static Bertrand price competition in each period

- Continuum of consumers make static purchase decision
- No switching costs: buy from the lower price supplier

Cost-reducing investments

State-of-the-art production cost c process

- Initial value c_0, lowest value 0: $0 \leq c \leq c_0$
- Discretized with n points
- Follows exogenous Markov process and only improves
- Markov transition probability $\pi(c_{t+1}|c_t)$
  $\pi(c_{t+1}|c_t) = 0$ if $c_{t+1} > c_t$

Investment choice: binary

- Investment cost of $K(c)$ to obtain marginal cost c
- One period construction time: production with technology obtained at t starts at $t + 1$

State space and information structure

Common knowledge

- State of the game: cost structure (c_1, c_2, c)
- State space is $S = (c_1, c_2, c) \subset R^3$: $c_1 \geq c$, $c_2 \geq c$
- Actions are observable

Private information

- In each period each firm incurs additive costs (benefits) from not investing and investing $\eta\epsilon_{i,I}$ and $\eta\epsilon_{i,N}$
- $\epsilon_{i,I}$ and $\epsilon_{i,N}$ are extreme value distributed, independent across choice, time and firms
- $\eta \geq 0$ is a scaling parameter
- Investment choice probabilities have logit form for $\eta > 0$

Timing of moves

Pricing decisions are made simultaneously

Expected one period profit of firm i from Bertrand game ($j \neq i$)

$$r_i(c_1, c_2) = \begin{cases} 0 & \text{if } c_i \geq c_j \\ c_j - c_i & \text{if } c_i < c_j \end{cases}$$

Two versions regarding investment decisions

1. Simultaneous moves:
   - Investment decisions are made simultaneously
2. Alternating moves:
   - The "right to move" state variable $m \in \{1, 2\}$,
   - When $m = i$, only firm i can make a cost reducing investment
   - m follows an own Markov process (deterministic alternation as a special case).

Actions and behavior strategies

Two choices in each period

- $p_i(c_1, c_2, c) = \max(c_1, c_2)$ – Bertrand pricing decision
- $P_i^I(c_1, c_2, c)$ – probability of firm i to invest in state-of-the-art production technology

  $P_i^N(c_1, c_2, c) = 1 - P_i^I(c_1, c_2, c)$ – probability not to invest

Strategy profile

- $\sigma = (\sigma_1, \sigma_2)$ – pair of Markovian *behavior* strategies

  $\sigma_i = \Big(p_i(c_1, c_2, c), P_i^I(c_1, c_2, c)\Big) \in \mathbb{R}_+ \times [0, 1]$
- Pure strategies included as special case

Definition of Markov Perfect Equilibium

Definition (Markov perfect equilibrium (MPE))

MPE of Bertrand investment stochastic game is a pair of

- strategy profile $\sigma^* = (\sigma_1^*, \sigma_2^*)$, and
- pair of *value functions* $V(s) = \Big(V_1(s), V_2(s)\Big)$, $V_i : S \to R$,

such that

1. Bellman equations (below) are satisfied for each firm, and
2. strategies σ_1^* and σ_2^* constitute mutual best responses, and assign positive probabilities only to the actions in the set of maximizers of the Bellman equations.

Bellman equations, firm $i = 1$, simultaneous moves

$$V_i(c_1, c_2, c) = \max\left[v_i^I(c_1, c_2, c) + \eta\epsilon_{i,I}, v_i^N(c_1, c_2, c) + \eta\epsilon_{i,N}\right]$$

$$\begin{aligned} v_i^N(c_1, c_2, c) &= r^i(c_1, c_2) + \beta EV_i(c_1, c_2, c|N) \\ v_i^I(c_1, c_2, c) &= r^i(c_1, c_2) - K(c) + \beta EV_i(c_1, c_2, c|I) \end{aligned}$$

With extreme value shocks, the investment probability is

$$P_i^I(c_1, c_2, c) = \frac{\exp\{v_I^I(c_1, c_2, c)/\eta\}}{\exp\{v_i^I(c_1, c_2, c)/\eta\} + \exp\{v_i^N(c_1, c_2, c)/\eta\}}$$

Bellman equations, firm $i = 1$, simultaneous moves

The expected values are given by

$$
\begin{aligned}
EV_i(c_1, c_2, c|N) &= \int_0^c \left[P_j^I(c_1, c_2, c) H_i(c_1, c, c') + \right. \\
&\qquad \left. P_j^N(c_1, c_2, c) H_i(c_1, c_2, c') \right] \pi(dc'|c) \\
EV_i(c_1, c_2, c|I) &= \int_0^c \left[P_j^I(c_1, c_2, c) H_i(c, c, c') + \right. \\
&\qquad \left. P_j^N(c_1, c_2, c) H_i(c, c_2, c') \right] \pi(dc'|c)
\end{aligned}
$$

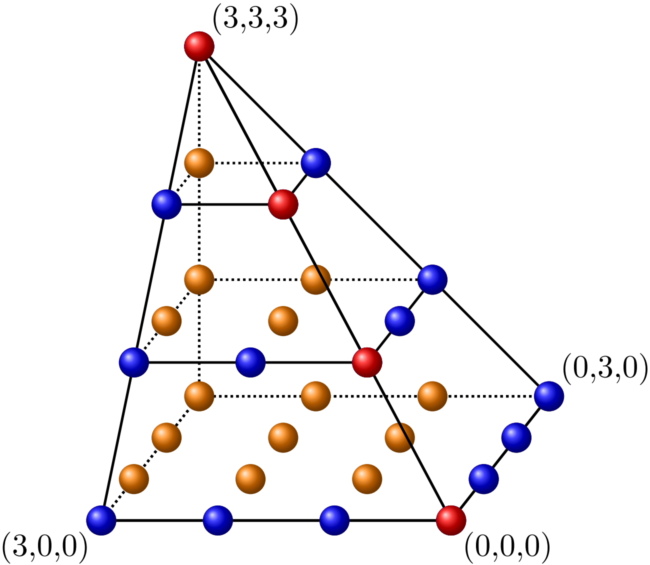
where

$$H_i(c_1, c_2, c) = \eta \log \left[\exp\left(v_i^N(c_1, c_2, c)/\eta\right) + \exp\left(v_i^I(c_1, c_2, c)/\eta\right) \right]$$

is the "smoothed max" or logsum function

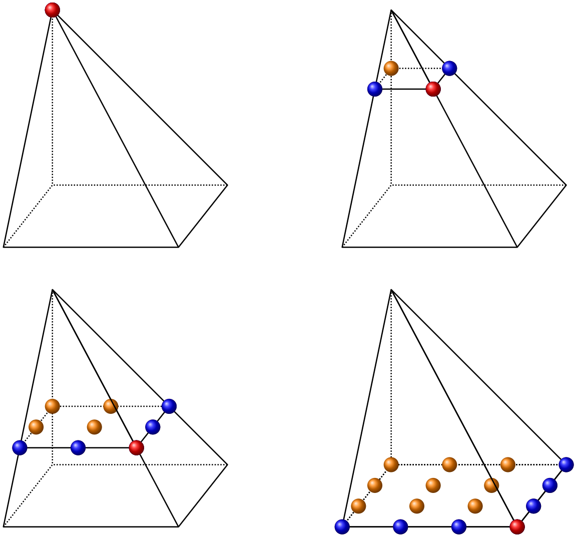
Discretized state space = a "quarter pyramid"

$S = \{(c_1, c_2, c) | c_1 \geq c,\ c_2 \geq c, c \in [0, 3]\},\ n = 4$



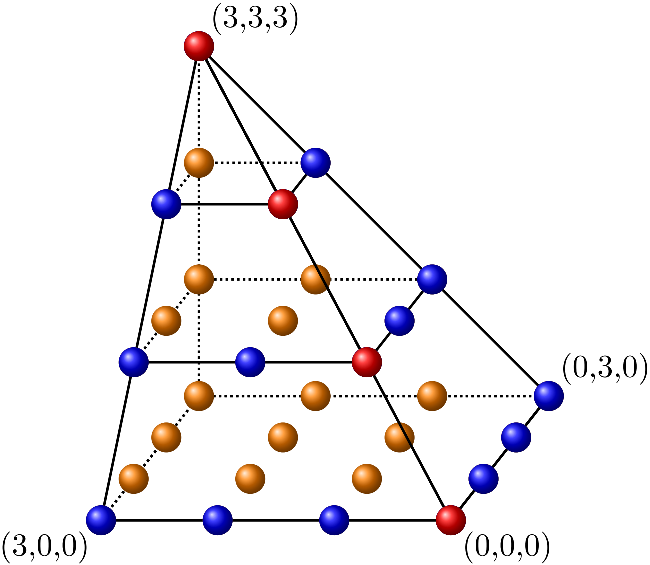
Transitions due to technological progress

As c decreases, the game falls through the layers of the pyramid



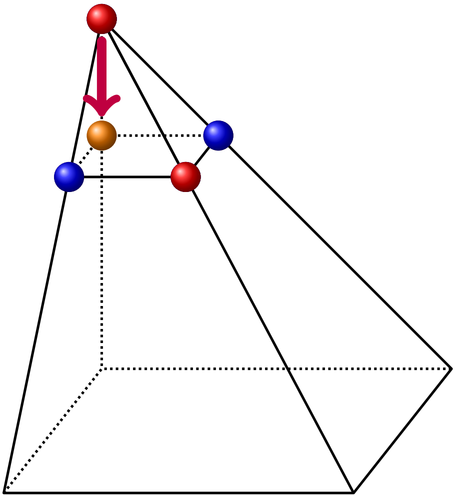
Discretized state space = a "quarter pyramid"

$S = \{(c_1, c_2, c) | c_1 \geq c,\ c_2 \geq c, c \in [0, \bar{c}]\},\ N = 4$



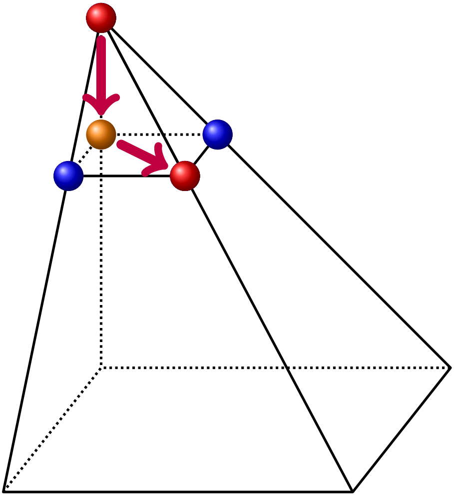
Game dynamics: example

The game starts at the apex, as some point technology improves



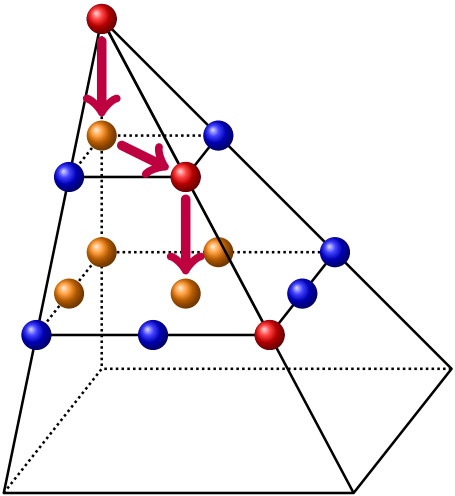
Game dynamics: example

Both firms buy new technology $c = 2 \rightsquigarrow (c_1, c_2, c) = (2, 2, 2)$



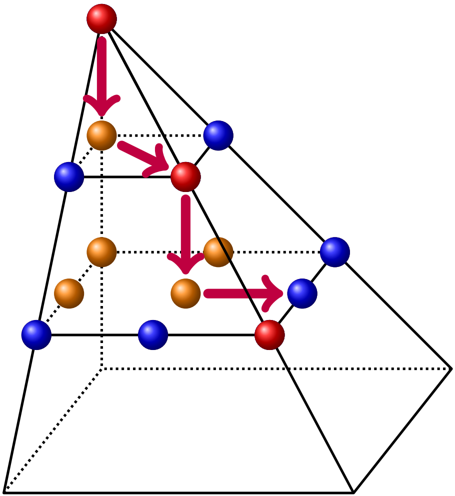
Game dynamics: example

State-of-the-art technology becomes $c = 1 \rightsquigarrow (c_1, c_2, c) = (2, 2, 1)$



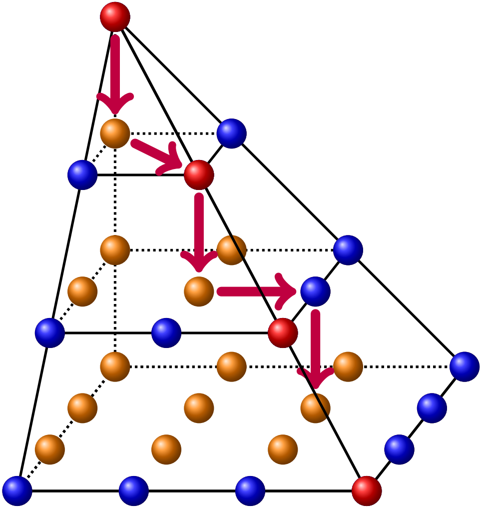
Game dynamics: example

Firm 1 invests and becomes cost leader $\rightsquigarrow (c_1, c_2, c) = (1, 2, 1)$



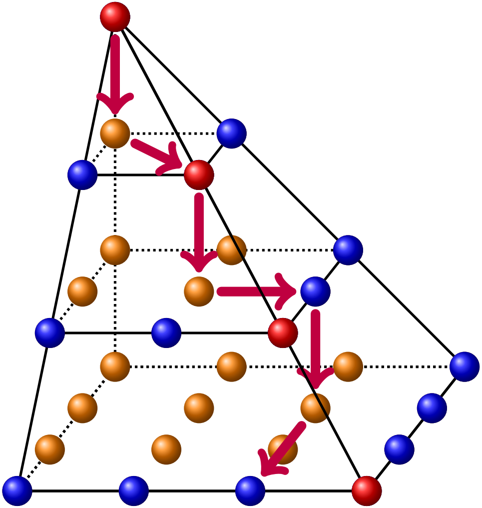
Game dynamics: example

State-of-the-art technology becomes $c = 0 \rightsquigarrow (c_1, c_2, c) = (1, 2, 0)$



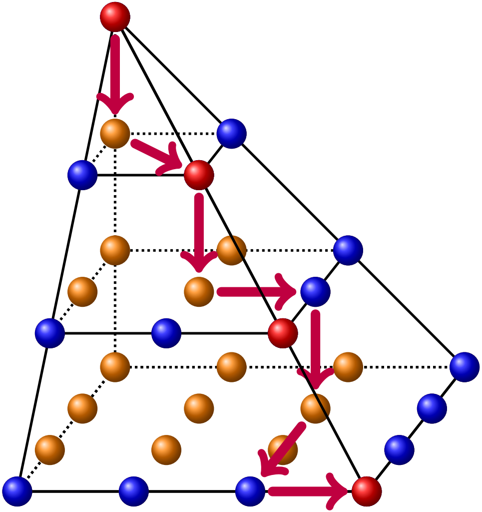
Game dynamics: example

Firm 2 leapfrogs firm 1 to become new cost leader $\rightsquigarrow (c_1, c_2, c) = (1, 0, 0)$



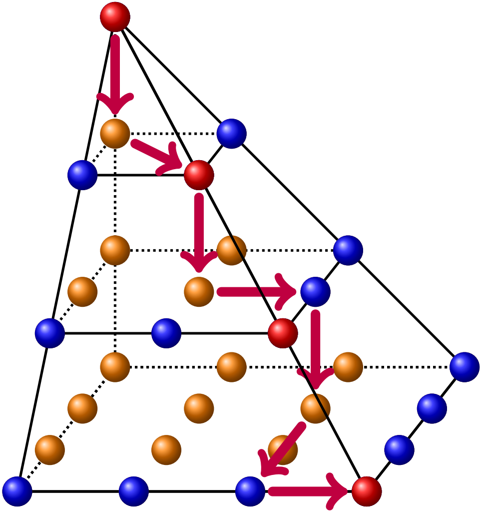
Game dynamics: example

Firm 1 invests, and the game reaches terminal state $\rightsquigarrow (c_1, c_2, c) = (0, 0, 0)$



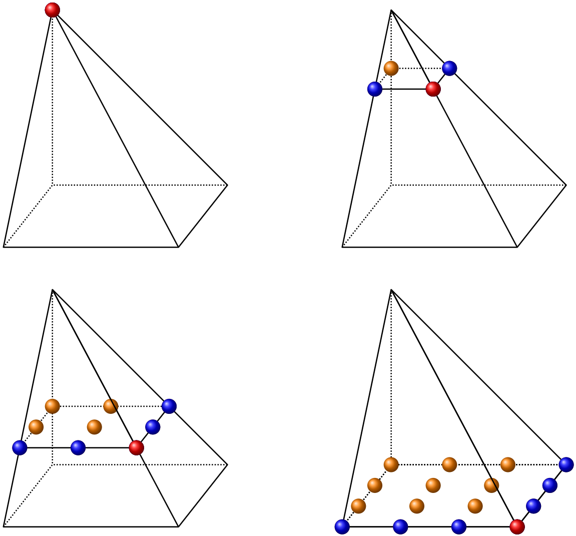
Game dynamics: example

Firm 1 invests, and the game reaches terminal state $\rightsquigarrow (c_1, c_2, c) = (0, 0, 0)$



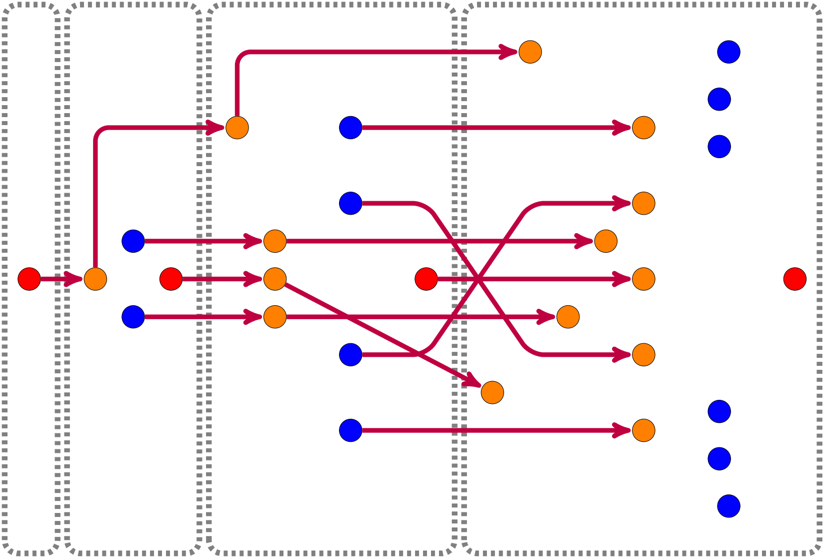
Transitions due to technological progress

As c decreases, the game falls through the layers of the pyramid



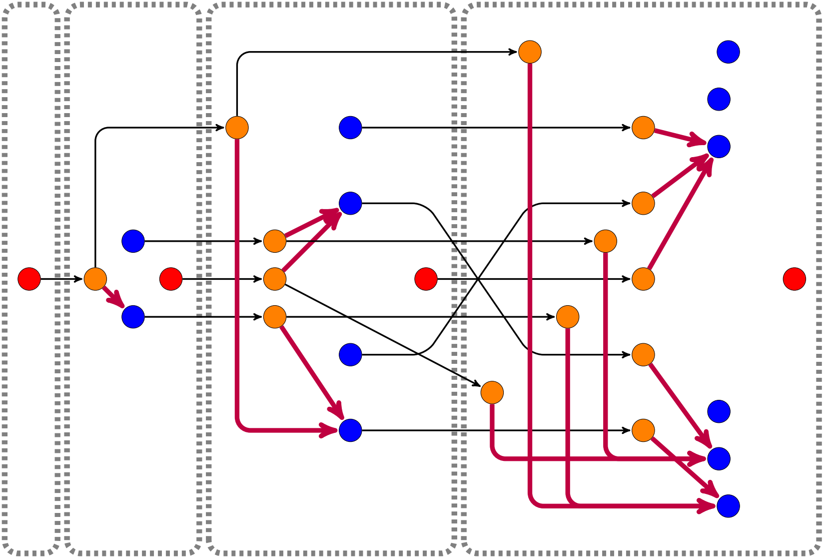
Transitions due to technological progress

As c decreases, the game falls through the layers of the pyramid



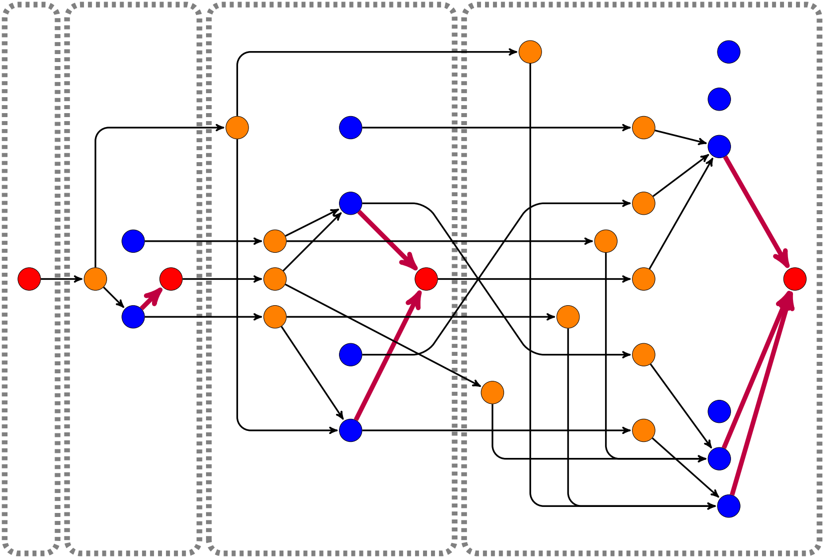
Strategy-specific partial order on S

Strategy σ_1 of firm 1: invest at all interior points



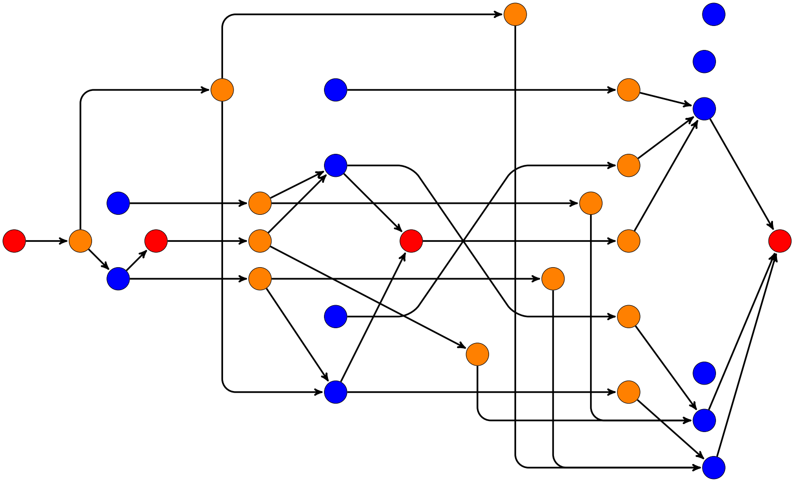
Strategy-specific partial order on S

Strategy σ_2 of firm 2: invest at all edge points



Strategy-specific partial order on S

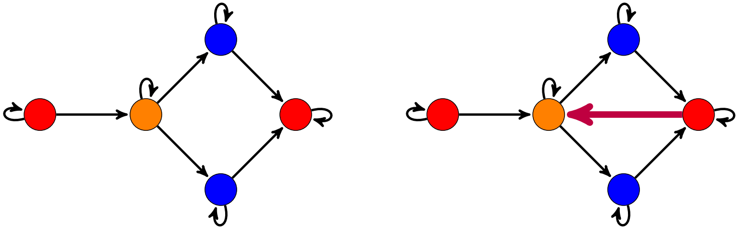
Strategy $\sigma = (\sigma_1, \sigma_2)$ of both firms



No loop (anti-cycling) condition

Hypothetical strategy profile inducing cycles

Self-loops appear when the game remains in the same state for two or more consecutive periods of time

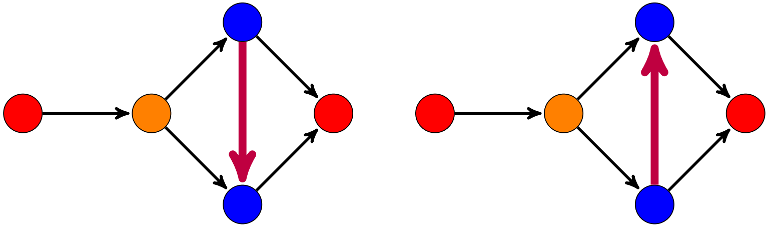


But loops between different states are not allowed

Consistency of strategy specific partial orders

Two hypothetical inconsistent strategies

Two strategies that induce opposite transitions are inconsistent



Note that in both cases the no-loop condition is satisfied

Definition of the Dynamic Directional Games

Definition (Dynamic Directional Games, DDG)

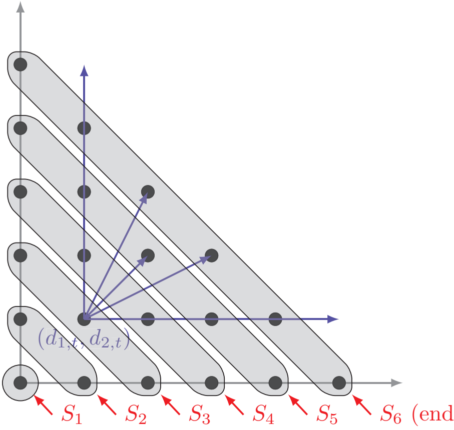
Finite state Markovian stochastic game is a DDG if it holds:

1. Every feasible Markovian strategy σ satisfies the no loop condition.
2. Every pair of feasible Markovian strategies σ and σ' induce consistent partial orders on the state space.

Example of DDG: Martket tipping game

Dubè, Hitsch and Chintagunta, 2010 *Marketing Science*
Tipping and Concentration in Markets with Indirect Network Effects

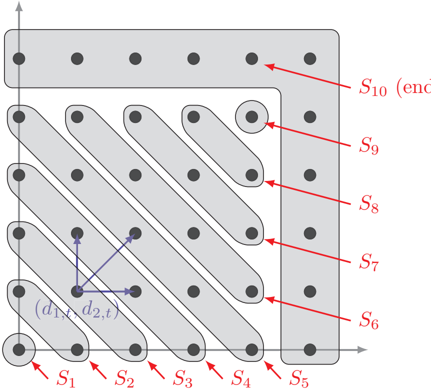
- Two competing gaming **platforms** (Xbox vs. Playstation)
- Number of games for each console depends on market share
- Consumers choose product they believe will win the war of the standards, or delay purchase
- Brand choices are absorbing: adoption base can only increase



Example of DDG: Patent race game

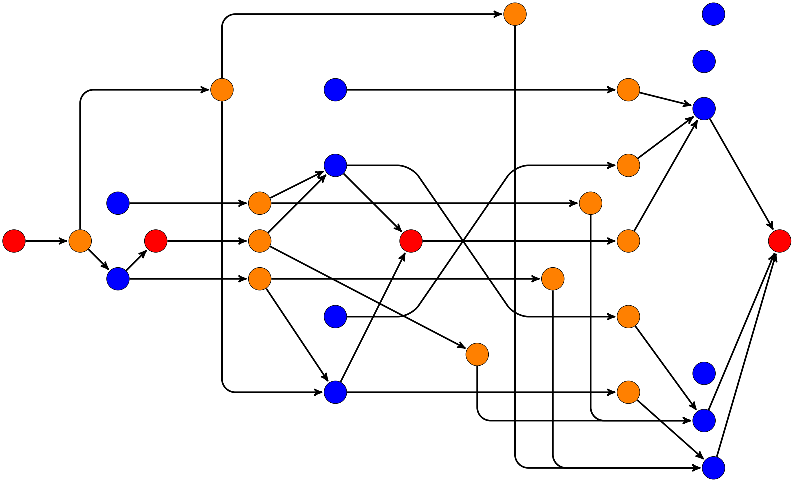
Judd, Schmedders and Yeltekin, 2012 *IER*
Optimal Rules for Patent Races

- Two firms are engaged in the N-steps race to acquire a patent
- Probability a step depends on the chosen R&D investment
- Only forward steps are possible, no "forgetting"
- Used state dependence structure to speed up computation of MPE by solving non-linear systems



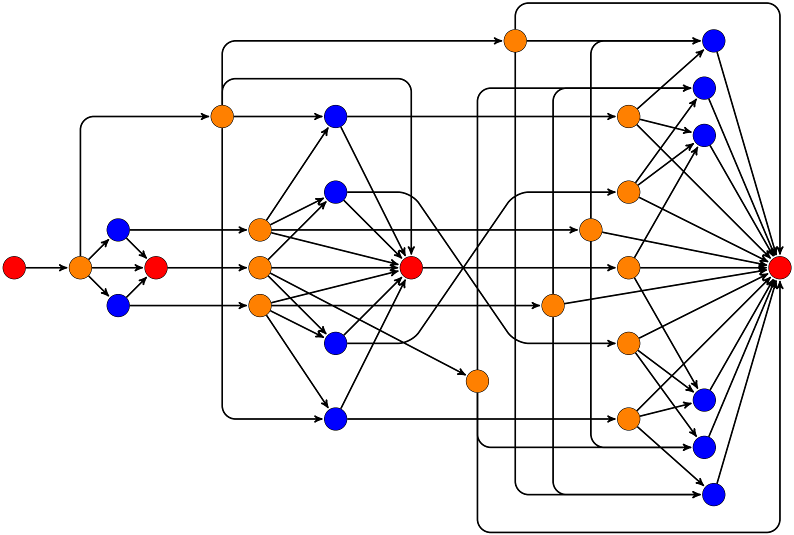
Strategy-specific partial order on S

Strategy $\sigma = (\sigma_1, \sigma_2)$ of both firms

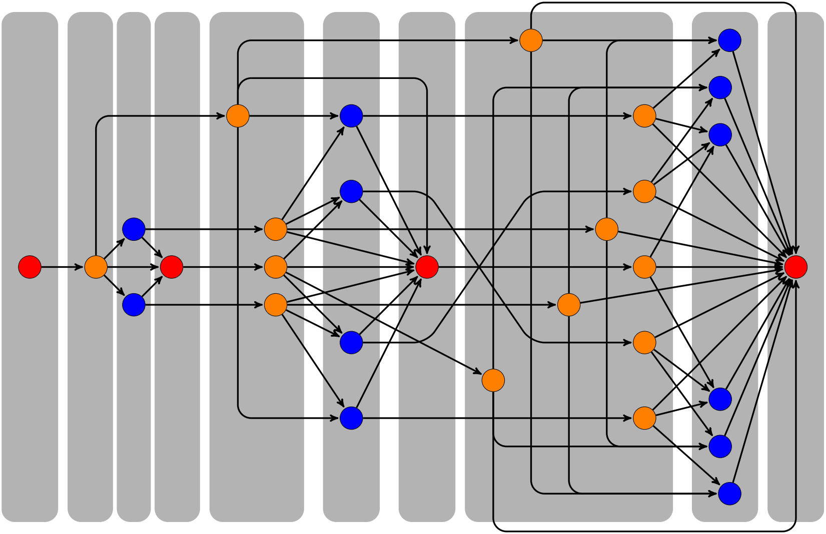


Strategy independent partial order on S

Coarsest common refinement of partial orders induced by all strategies

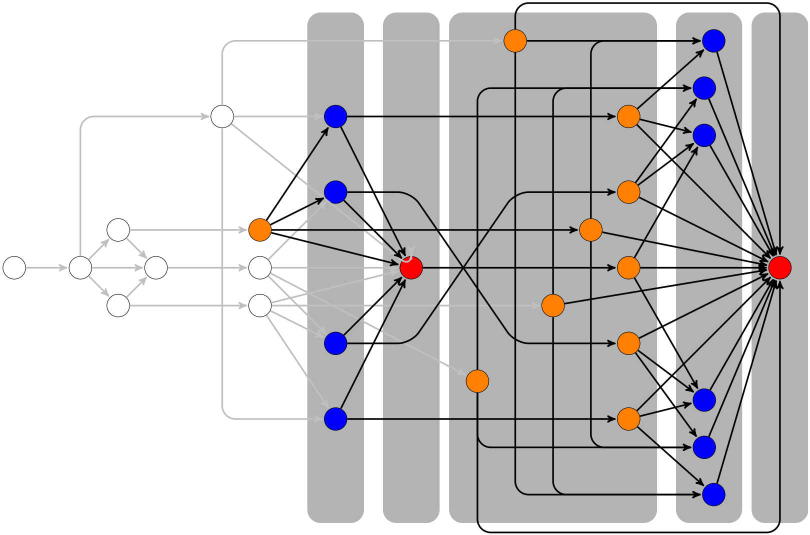


Total order on the set of stages



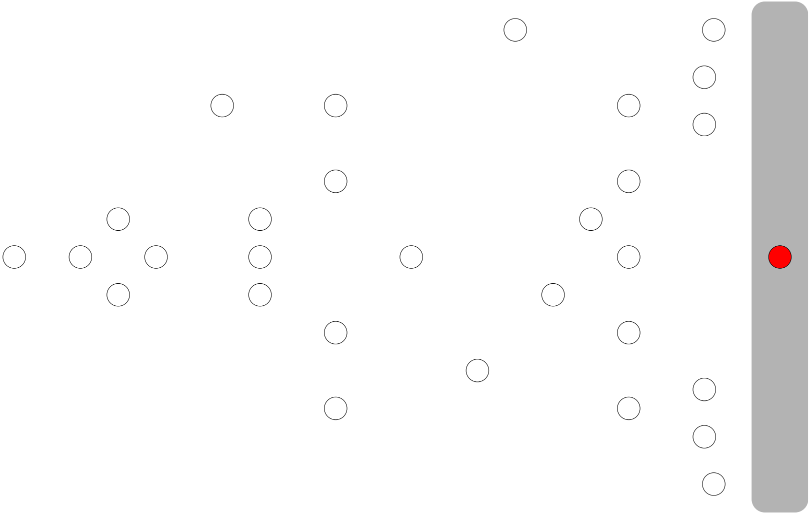
Subgames of DDG and continuation strategies

Subgames and continuation strategies



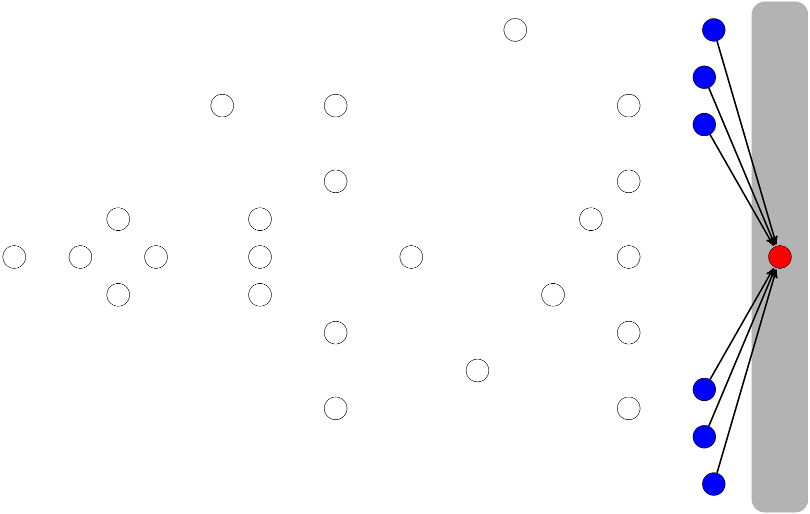
State recursion algorithm

Backward induction on stages of DDG



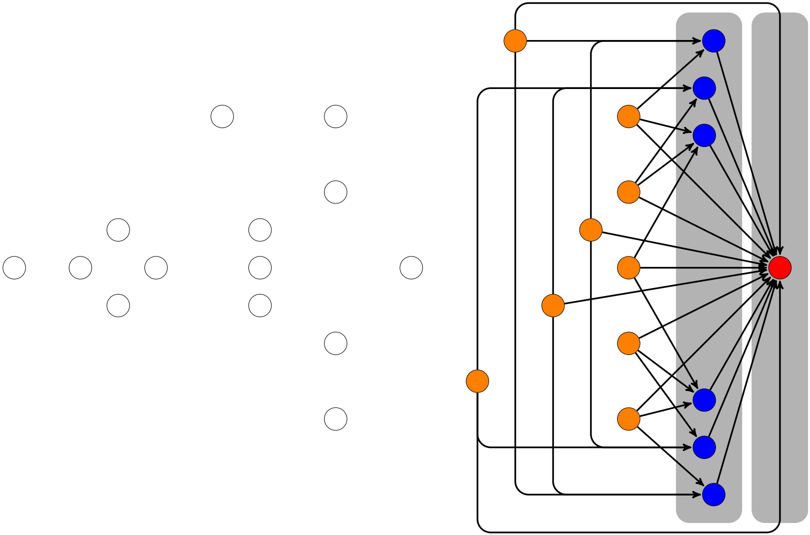
State recursion algorithm

Backward induction on stages of DDG



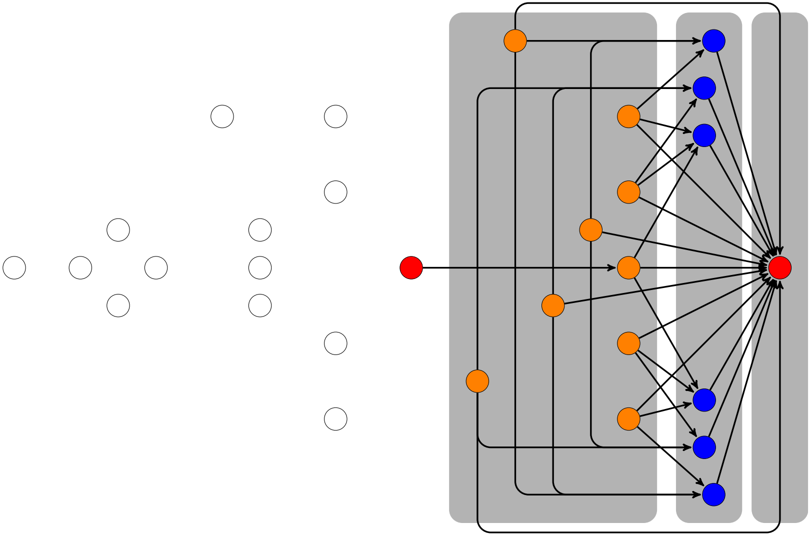
State recursion algorithm

Backward induction on stages of DDG



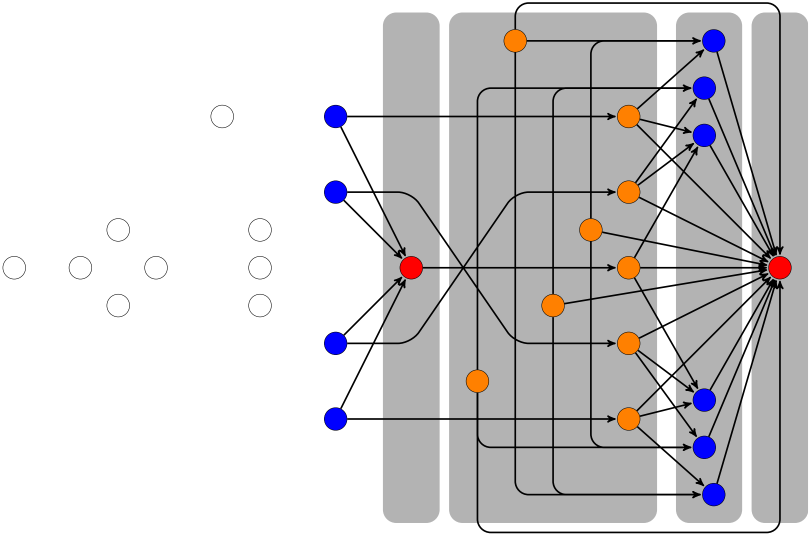
State recursion algorithm

Backward induction on stages of DDG



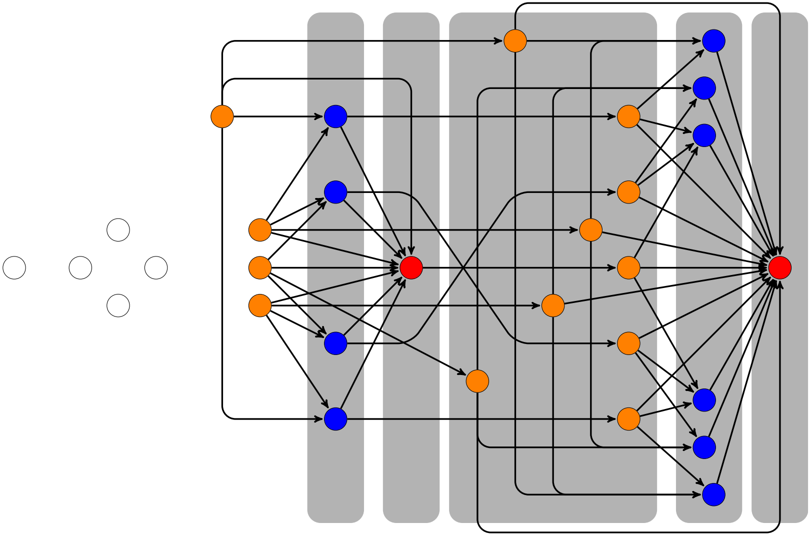
State recursion algorithm

Backward induction on stages of DDG



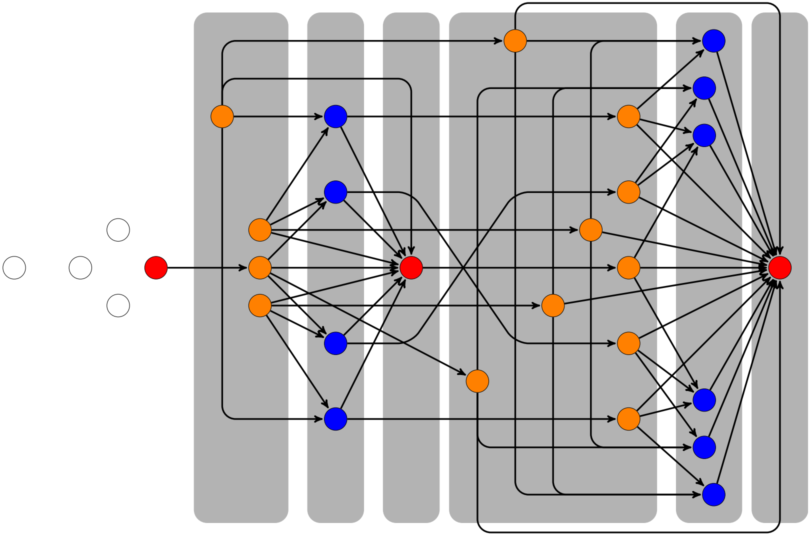
State recursion algorithm

Backward induction on stages of DDG



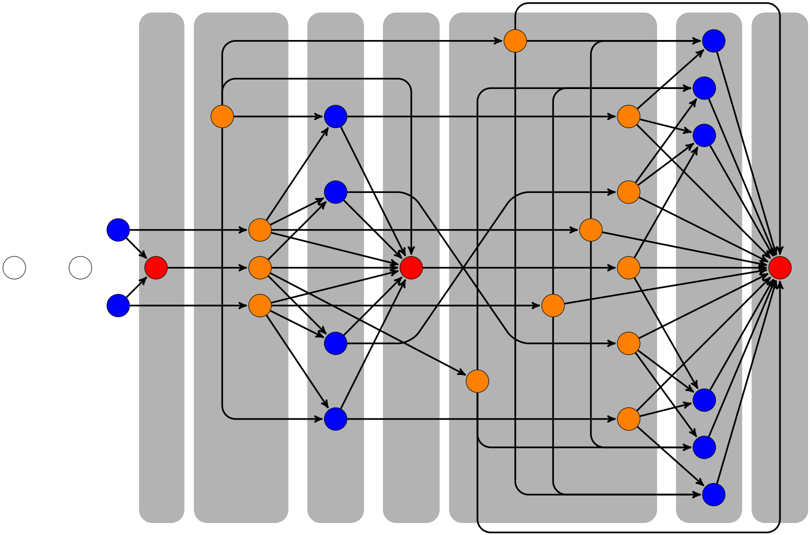
State recursion algorithm

Backward induction on stages of DDG



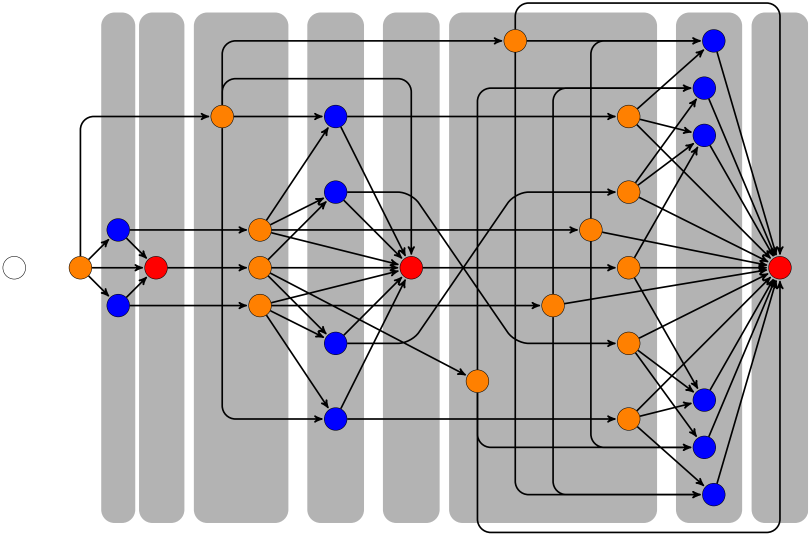
State recursion algorithm

Backward induction on stages of DDG



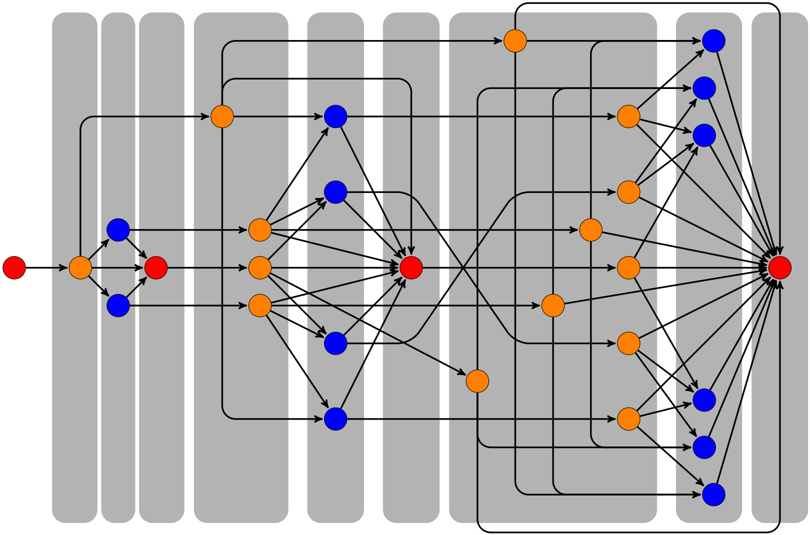
State recursion algorithm

Backward induction on stages of DDG



State recursion algorithm

Backward induction on stages of DDG

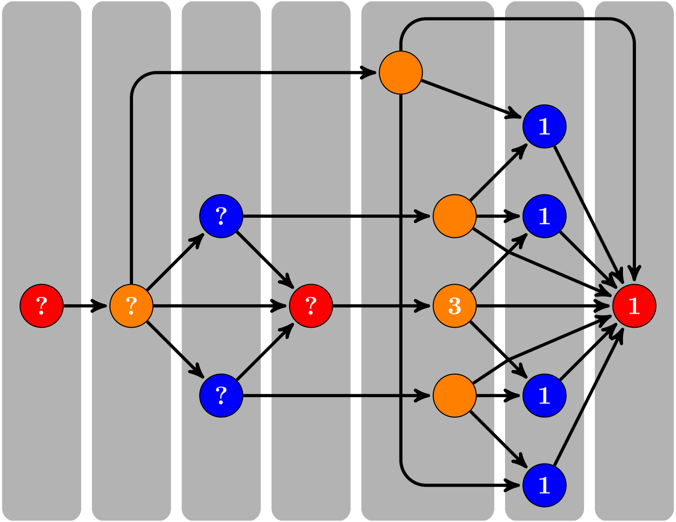


State Recursion versus Backward Induction

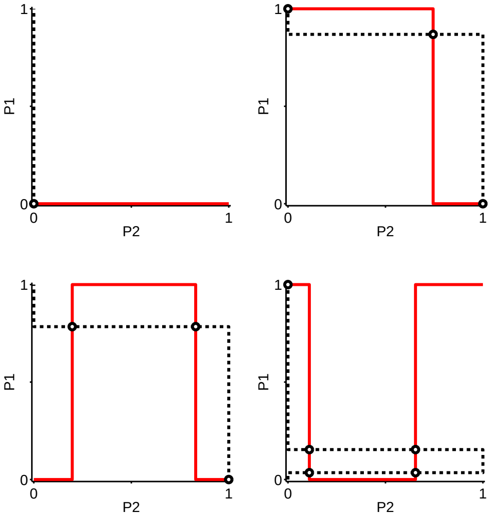
- State recursion – generalization of backward induction
- Runs on state space instead of time periods
- Time (t) evolves as $t \rightarrow t + 1$ with probability 1
- For stages of state space (τ) transitions are stochastic and not necessarily sequential
- Yet, probability of going $\tau \rightarrow \tau'$ is zero when $\tau' < \tau$
- With multiplicity, state recursion is performed conditional of a particular equilibrium selection rule (ESR)

Multiplicity of stage equibiria

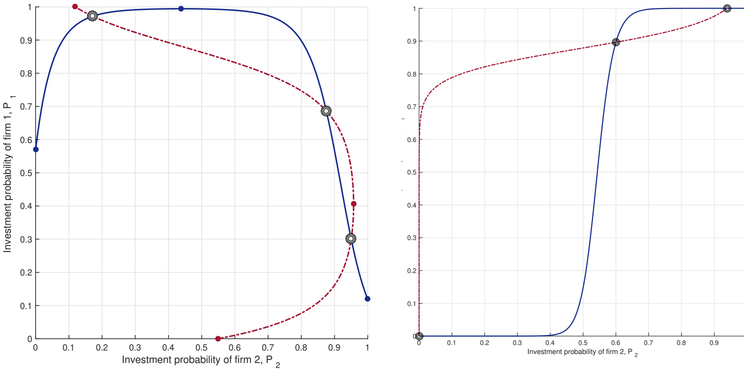
Number of equilibria in the higher stages depends on the selected equilibria



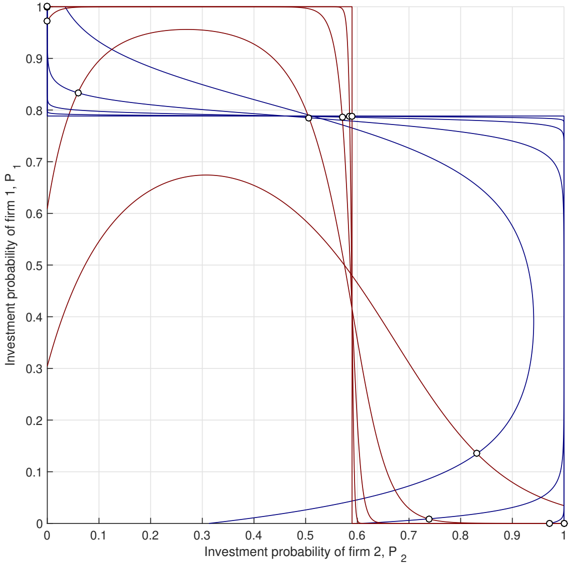
Best response correspondences for $\eta = 0$



Best response functions for $\eta > 0$



Best response functions as $\eta \to 0$



Recursive Lexicographic Search Algorithm

Building blocks of RLS algorithm:

1. State recursion algorithm solves the game conditional on equilibrium selection rule (ESR)
2. RLS algorithm efficiently cycles through all feasible ESRs

Challenge:

- Choice of a particular MPE for any stage game at any stage
- may alter the set and even the number of stage equilibria at earlier stages

Need to find feasible ESRs

- ESR = string of digits that index the selected stage equilibrium in each point

All possible ESR strings in lexicographic order

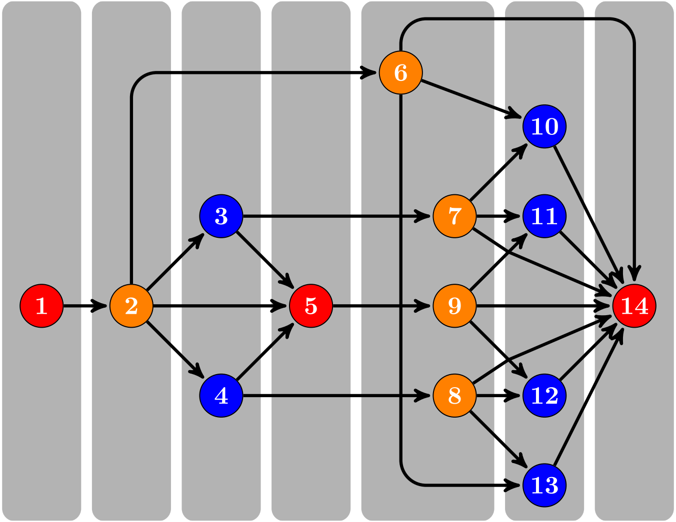
- Bound the maximum number of equilibria in each stage by $K = 3$

	c	e	e	e	e	i	i	i	i	c	e	e	i	c
ESR string	14	13	12	11	10	9	8	7	6	5	4	3	2	1
Lexicograph	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	0	0	0	0	0	0	0	0	0	0	0	0	0	1
	0	0	0	0	0	0	0	0	0	0	0	0	0	2
	0	0	0	0	0	0	0	0	0	0	0	0	1	0
	0	0	0	0	0	0	0	0	0	0	0	0	1	1
	0	0	0	0	0	0	0	0	0	0	0	0	1	2
	0	0	0	0	0	0	0	0	0	0	0	0	2	0
	0	0	0	0	0	0	0	0	0	0	0	0	2	1
	0	0	0	0	0	0	0	0	0	0	0	0	2	2
	0	0	0	0	0	0	0	0	0	0	0	1	0	0
	0	0	0	0	0	0	0	0	0	0	0	1	0	1
														…
	2	2	2	2	2	2	2	2	2	2	2	2	2	0
	2	2	2	2	2	2	2	2	2	2	2	2	2	1
	2	2	2	2	2	2	2	2	2	2	2	2	2	2

4,782,969

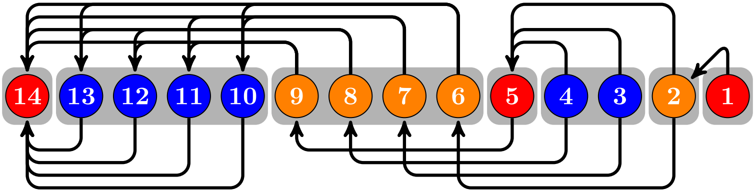
Indexing of points in the state space

Lower index for dependent points, highest for terminal stage

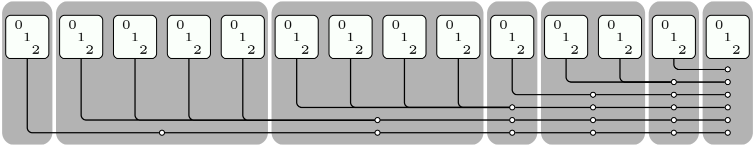


Preserving stage order in ESR strings

Formalization of the ESR as stings of digits



- Digits arranged to preserve the dependence structure



Recalculation of feasibility condition for new ESR

Avoid recalculation of subgames

	c	e	e	e	e	i	i	i	i	c	e	e	i	c	
ESR string	14	13	12	11	10	9	8	7	6	5	4	3	2	1	
	0	0	0	0	0	0	0	0	0	0	0	0	0	0	always admissible
Nr of eqb	1	1	1	1	1	3	3	3	3	1	1	1	**3**	1	
	0	0	0	0	0	0	0	0	0	0	0	0	0	**1**	
	0	0	0	0	0	0	0	0	0	0	0	0	0	**2**	
	0	0	0	0	0	0	0	0	0	0	0	0	**1**	0	admissible, solve
	1	1	1	1	1	3	3	3	3	1	1	1	3	*	

No changes in the solution of the game including the number of stage equilibria

Might have change

Jumping over blocks of infeasibles ESRs

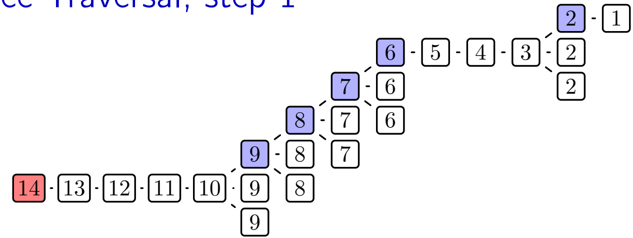
	c	e	e	e	e	i	i	i	i	c	e	e	i	c	
ESR string	14	13	12	11	10	9	8	7	6	5	4	3	2	1	Iteration:
	0	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	1
	1	1	1	1	1	3	3	3	3	1	1	1	**3**	**1**	1a
	0	0	0	0	0	0	0	0	0	0	0	0	0	**1**	
	0	0	0	0	0	0	0	0	0	0	0	0	0	2	
	0	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**1**	**0**	2
	1	1	1	1	1	3	3	3	3	1	1	1	**3**	**1**	2a
	0	0	0	0	0	0	0	0	0	0	0	0	1	**1**	
	0	0	0	0	0	0	0	0	0	0	0	0	1	2	
	0	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**2**	**0**	3
	1	1	1	1	1	3	3	3	**3**	**1**	**1**	**1**	**3**	**1**	
	0	0	0	0	0	0	0	0	0	0	0	0	2	**1**	3a
	0	0	0	0	0	0	0	0	0	0	0	0	2	2	
	0	0	0	0	0	0	0	0	0	0	0	**1**	0	0	3b
														…	
	0	0	0	0	0	0	0	0	0	0	**1**	0	0	0	3c
														…	
	0	0	0	0	0	0	0	0	0	**1**	0	0	0	0	3d
														…	
	0	**0**	**0**	**0**	**0**	**0**	**0**	**0**	**1**	**0**	**0**	**0**	**0**	**0**	4
														…	

Recursive Lexicographic Search (RLS) Algorithm

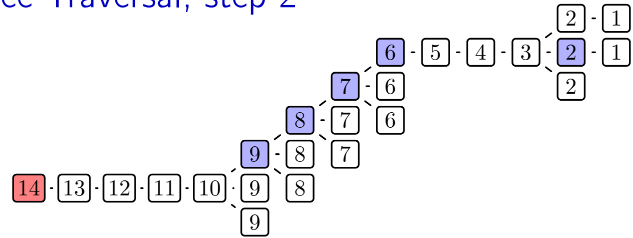
1. Set ESR $= (0, \ldots, 0)$
2. Run State Recursion using the current ESR
3. Save the number of equilibria in every stage game as *ne*(ESR)
4. Add 1 to the ESR in bases *ne*(ESR) to obtain new feasible ESR
5. Stopping rule: successor function exceeds the maximum number with given number of digits
6. Return to step 2

RLS = tree traversal!

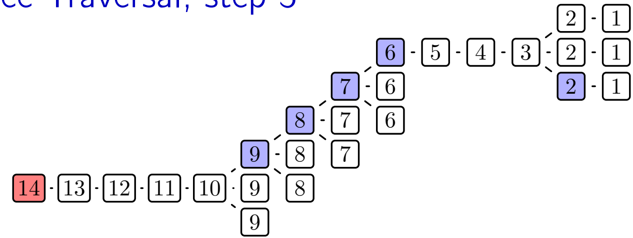
RLS Tree Traversal, step 1



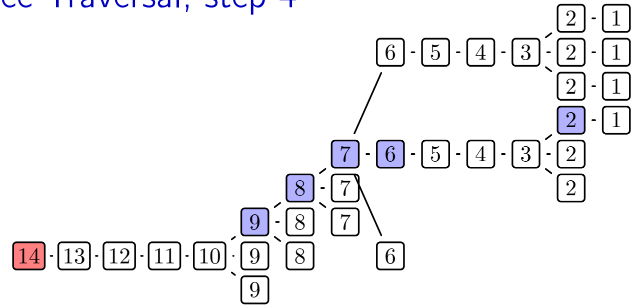
RLS Tree Traversal, step 2



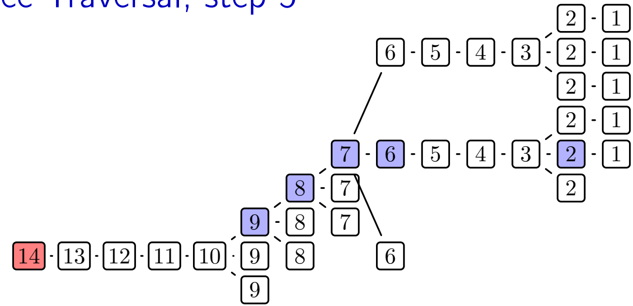
RLS Tree Traversal, step 3



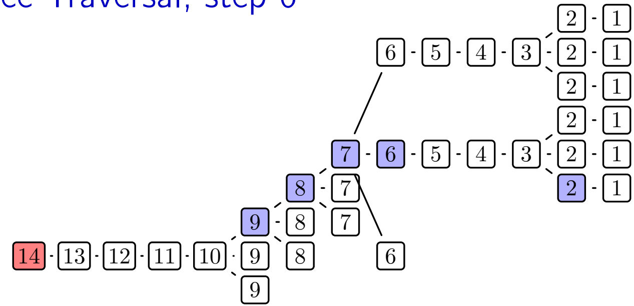
RLS Tree Traversal, step 4



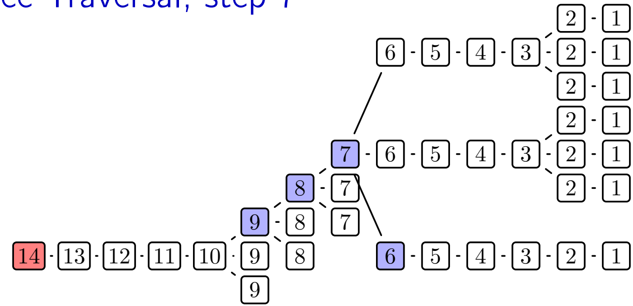
RLS Tree Traversal, step 5



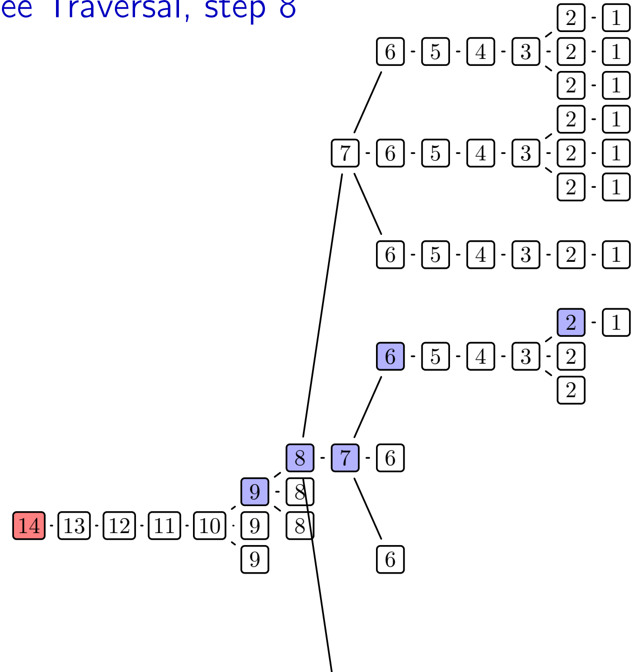
RLS Tree Traversal, step 6



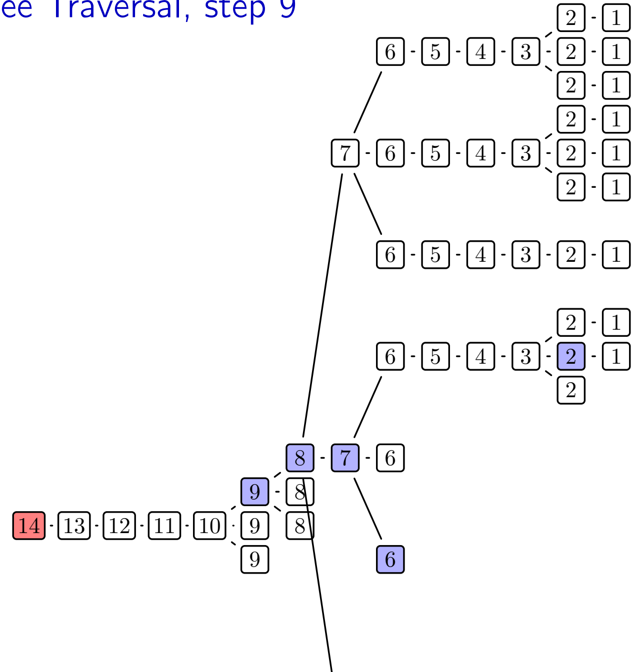
RLS Tree Traversal, step 7



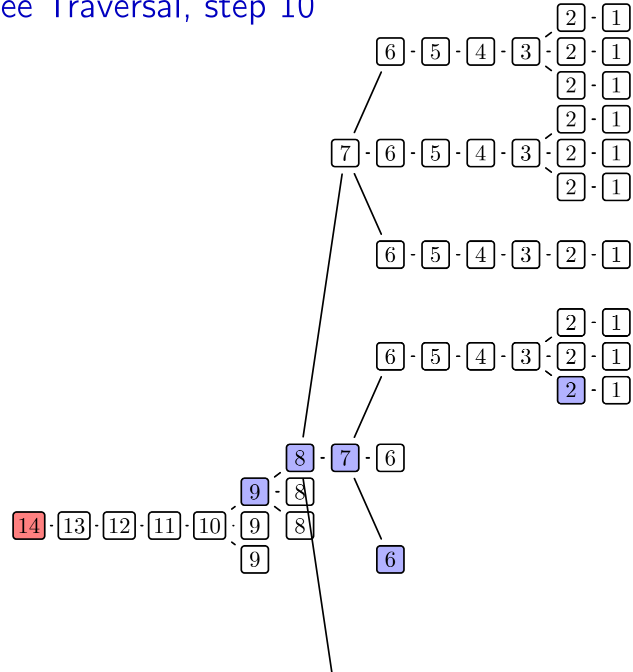
RLS Tree Traversal, step 8



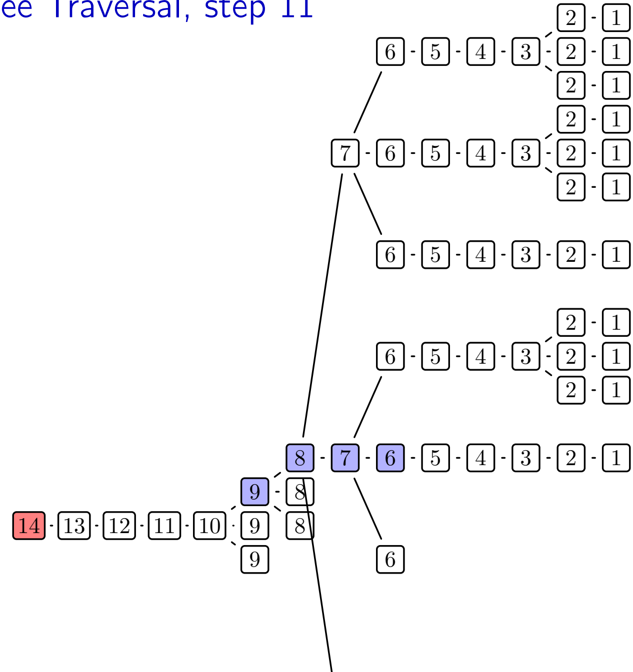
RLS Tree Traversal, step 9



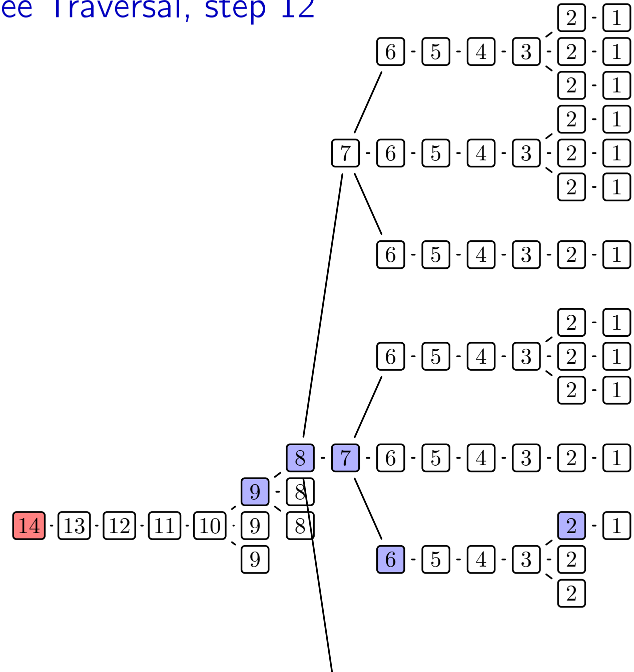
RLS Tree Traversal, step 10



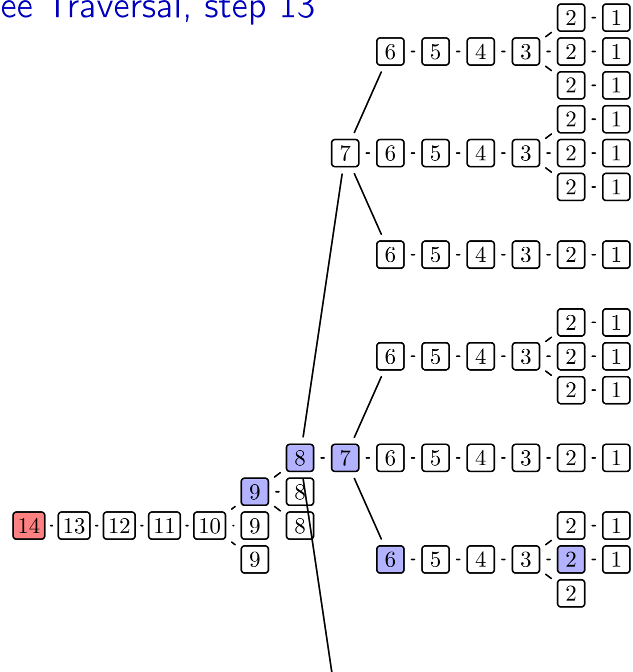
RLS Tree Traversal, step 11



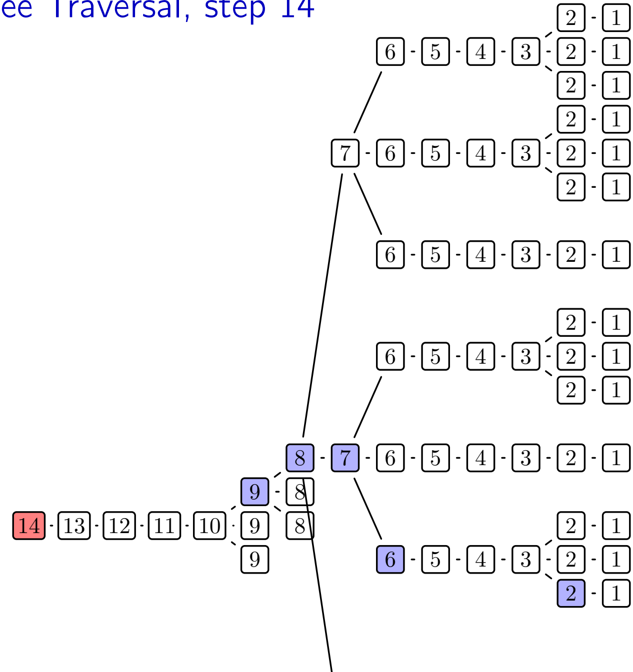
RLS Tree Traversal, step 12



RLS Tree Traversal, step 13



RLS Tree Traversal, step 14



Main result of the RLS Algorithm

Theorem (Decomposition theorem, strong)

Assume there exists an algorithm that can find all MPE of every stage game of the DDG, and that the number of these equilibria is finite in every stage game.
Then the RLS algorithm finds all MPE of the DDG in a finite number of steps, which equals the total number of MPE.

Iskhakov, Rust and Schjerning, 2016, ReStud
"Recursive lexicographical search: Finding all markov perfect equilibria of finite state directional dynamic games."

Main result of the RLS Algorithm

Theorem (Decomposition theorem, weak)

Assume there exists an algorithm that can at least one MPE of every stage game of the DDG, and that the number of these equilibria is finite in every stage game.
Then the RLS algorithm finds some (at least one) MPE of the DDG in a finite number of steps, which does not exceed the total number of MPE.

RLS algorithm: running times
$K = 3$

Simultaneous moves	$n = 3$	$n = 4$
Upper bound on number of MPE	4,782,969	3,948,865,611
Actual number of equilibria	127	46,707
Time used	0.008 sec.	0.334 sec.

Simultaneous moves	$n = 5$
Upper bound on number of MPE	174,449,211,009,120,166,087,753,728
Actual number of equilibria	192,736,405
Time used	45 min.

Alternating moves	$n = 5$
Upper bound on number of MPE	174,449,211,009,120,166,087,753,728
Actual number of equilibria	1
Time used	0.006 sec.

Resolution to the Bertrand investment paradox

Theorem (Solution to Bertrand investment paradox)

If investment is socially optimal at a state point $(c_1, c_2, c) \in S$, then

- *no investment by both firms cannot be an MPE outcome in the subgame starting from (c_1, c_2, c) in either the simultaneous or alternating move versions of the dynamic game.*

We show:

1. Many types of endog. coordination is possible in equilibrium
   - Leapfrogging (alternating investments)
   - Preemption (investment by cost leader)
   - Duplicative (simultaneous investments)
2. The equilibria are generally inefficient due to over-investment
   - Duplicative or excessively frequent investments

Multiplicity of equilibria

Theorem (Sufficient conditions for uniqueness)

In the dynamic Bertrand investment and pricing game a sufficient condition for the MPE to be unique is that

1. *firms move in alternating fashion (i.e. $m \neq 0$), and,*
2. *for each $c > 0$ in the support of π we have $\pi(c|c) = 0$.*

1. If firms move simultaneously,
   equilibrium is generally not unique.
2. If technological change is stochastic,
   equilibrium is generally not unique.

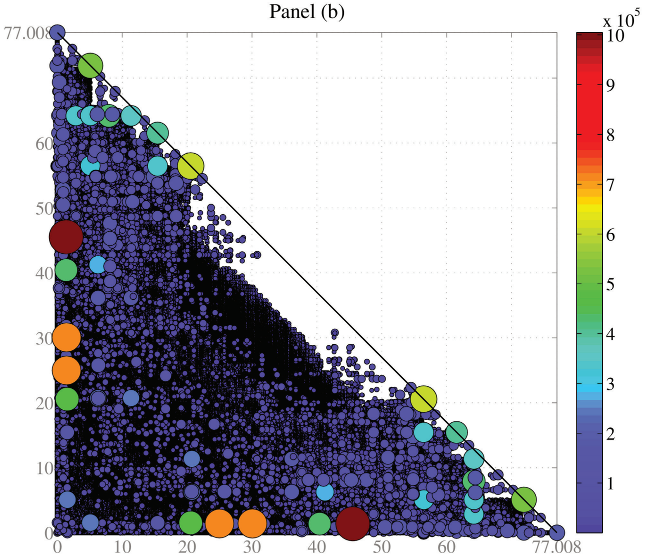
Pay-offs in the simultaneous move game

Theorem (Triangular payoffs in the simultaneous move game)

Suppose that the $\{c_t\}$ process has finite support, that there are no idiosyncratic shocks to investment (i.e. $\eta = 0$) and that firms move simultaneously

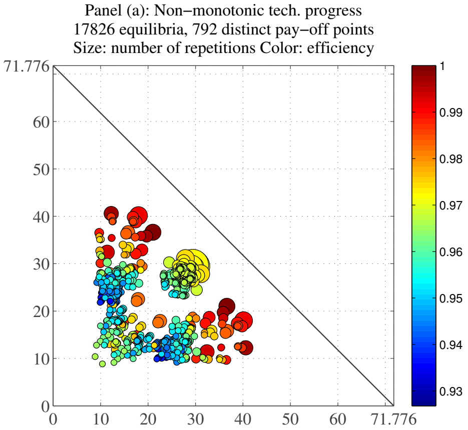
- *The (convex hull of the) set of the expected discounted equilibrium payoffs at the apex state $(c_0, c_0, c_0) \in S$ is a triangle*
- *The vertices of this triangle are at the points $(0, 0)$, $(0, V_M)$ and $(V_M, 0)$ where $V_M = v_{N,i}(c_0, c_0, c_0)$ is the expected discounted payoff to firm i in the monopoly equilibrium where firm i is the monopolist investor.*

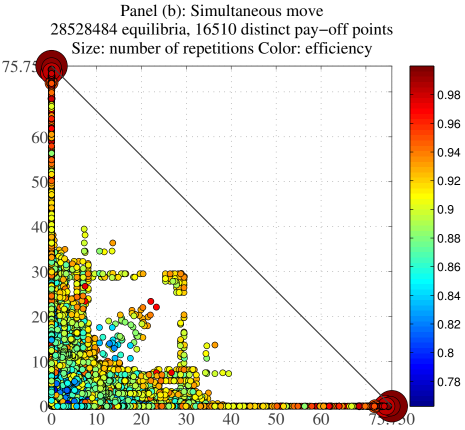
Pay-off map



Panel (b)
77.008
60
50
40
30
20
10
0
0
10
20
30
40
50
60
77.008
x 10^5
10
9
8
7
6
5
4
3
2
1

Pay-offs: alternating vs simultaneous move games

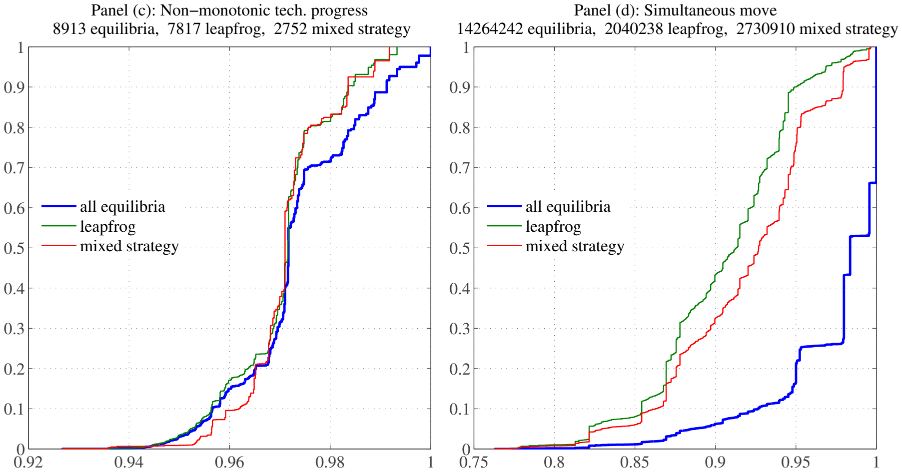




Theorem (Inefficiency of mixed strategy equilibria)

A necessary condition for efficiency in the dynamic Bertrand investment and pricing game is that along MPE path only pure strategy stage equilibria are played.

Efficiency: alternating vs simultaneous move games



Riordan and Salant: Full Preemption

Theorem (**Riordan and Salant, 1994**)

The continuous time investment game where

1. *right to move alternates deterministically.*
2. $K(c) = K$ *and is not prohibitively high.*
3. *technological progress is deterministic:* $c(t)$ *is a continuous, decreasing function*

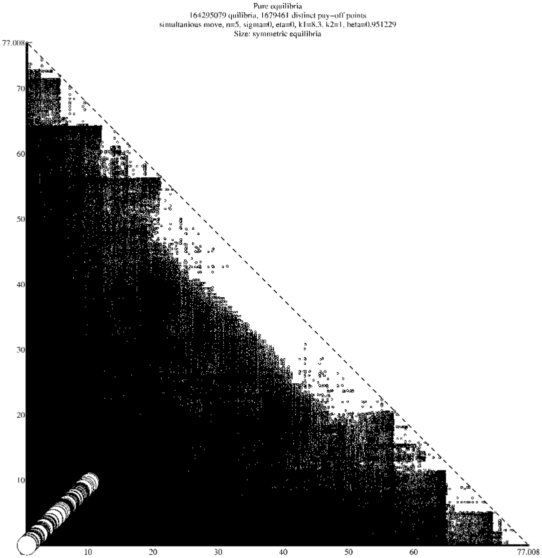
has a unique MPE with

- *preemptive investments: by only one firm and no investment in equilibrium by its opponent.*
- *rent dissipation: discounted payoffs of both firms in equilibrium is* 0*, so the entire surplus is wasted on excessively frequent investments by the preempting firm.*

We show by computing examples and counterexamples

1. Confirm R&S the result with high $K$and small dt
2. Underinvestment: Rent dissapation is not a general outcome - disappears when K is low relative dt
3. Leapfrogging: Preemption is not the general outcome - disappears when K is even lower
4. Random move alternation $\rightarrow$ Leapfrogging
5. Random onestep technology $\rightarrow$ Leapfrogging
6. Random multistep technology $\rightarrow$ Leapfrogging
7. Simultaneous moves $\rightarrow$ Leapfrogging

Symmetric equilibria: $V_1(c_1, c_2, c) = V_2(c_2, c_1, c)$



Failure of homotopy approach

Homotopy parameter: η

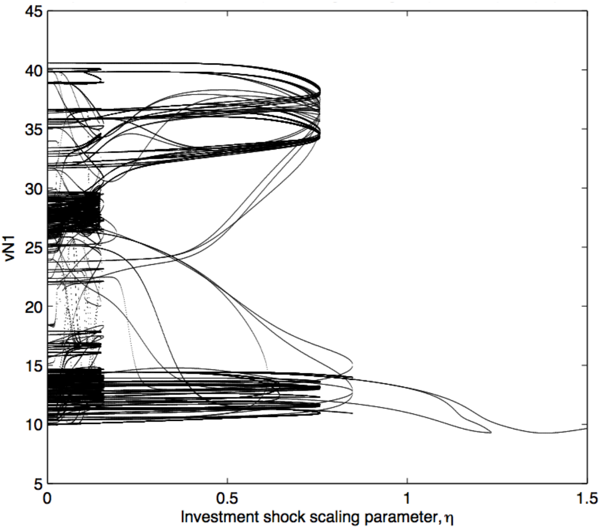
- In each period each firm incurs additive random costs/benefit from not investing and investing
- η is a scaling parameter that index variance of idiosyncratic shocks to investment
- High $\eta \rightarrow$ unique equilibrium $\eta \rightarrow 0 \rightarrow$ multiple equilibria

Problems:

- Multiplicity of equilibria $\rightarrow$ too many bifurcations along the path
- Equilibrium correspondence is not lower hemi-continuous

Failure of homotopy approach

Equilibrium correspondance, alternating move game: $V_{N,1}(c_0, c_0, c_0)$ vs. η



Estimation of directional dynamic games: Full solution nested MLE estimation

Nested Recursive Lexicographic Search algorithm

Markov Perfect Equilibria

- MPE is a pair of strategy profile and value functions
- In compact notation

$$\mathsf{V} = \Psi^{\mathsf{V}}(\mathsf{V}, \mathsf{P}, \theta)$$
$$\mathsf{P} = \Psi^{\mathsf{P}}(\mathsf{V}, \mathsf{P}, \theta)$$

- Set of all Markov Perfect Equilibria

$$SOL(\Psi, \theta) = \left\{ (\mathsf{P}, \mathsf{V}) \,\middle|\, \begin{array}{l} \mathsf{V} = \Psi^{\mathsf{V}}(\mathsf{V}, \mathsf{P}, \theta) \\ \mathsf{P} = \Psi^{\mathsf{P}}(\mathsf{V}, \mathsf{P}, \theta) \end{array} \right\}$$

- $\Psi^{\mathsf{V}}: \ \mathsf{V}, \mathsf{P} \longrightarrow \mathsf{V}$ Bellman operator
- $\Psi^{\mathsf{P}}: \ \mathsf{V}, \mathsf{P} \longrightarrow \mathsf{P}$ Choice probability formulas (logit)
- $\Gamma: \ \mathsf{P} \longrightarrow \mathsf{V}$ Hotz-Miller inversion

Estimation methods for *dynamic* stochastic games

- Two step (CCP) estimators
  - Fast, potentially large finite sample biases

Hotz, Miller (1993); Altug, Miller (1998); Pakes, Ostrovsky, and Berry (2007); Pesendorfer, Schmidt-Dengler (2008)

1. Estimate CCP $\to \hat{\mathrm{P}}$
2. Method of moments • Minimal distance • Pseudo likelihood

$$\min_{\theta} \left[\hat{\mathrm{P}} - \Psi^{\mathrm{P}}(\Gamma(\theta, \hat{\mathrm{P}}), \hat{\mathrm{P}}, \theta)\right]' W \left[\hat{\mathrm{P}} - \Psi^{\mathrm{P}}(\Gamma(\theta, \hat{\mathrm{P}}), \hat{\mathrm{P}}, \theta)\right]$$

$$\max_{\theta} \mathcal{L}(\mathrm{Z}, \Psi^{\mathrm{P}}(\Gamma(\theta, \hat{\mathrm{P}}), \hat{\mathrm{P}}, \theta))$$

- Nested pseudo-likelihood (NPL)
  - Recursive two step pseudo-likelihood
  - Bridges the gap between efficiency and tractability
  - Unstable under multiplicity

Aguirregabiria, Mira (2007); Pesendorfer, Schmidt-Dengler (2010); Kasahara and Shimotsu (2012); Aguirregabiria, Marcoux (2021)

Estimation methods for *dynamic* stochastic games

- Equilibrium inequalities (BBL)
  - Minimize the one-sided discrepancies
  - Computationally feasible in large models

  Bajari, Benkard, Levin (2007)

- Math programming with equilibrium constraints (MPEC)
  - MLE as constrained optimization
  - Does not rely on the structure of the problem
  - Much bigger computational problem

  Su (2013); Egesdal, Lai and Su (2015)

$$\max_{(\theta, \mathsf{P}, \mathsf{V})} \mathcal{L}(\mathsf{Z}, \mathsf{P}) \text{ subject to } \mathsf{V} = \Psi^{\mathsf{V}}(\mathsf{V}, \mathsf{P}, \theta), \mathsf{P} = \Psi^{\mathsf{P}}(\mathsf{V}, \mathsf{P}, \theta)$$

- All solution homotopy MLE

  Borkovsky, Doraszelsky and Kryukov (2010)

Overview of NRLS

- Robust and *computationally feasible*$^{(?)}$ MLE estimator for directional dynamic games (DDG)
- Rely of full solution algorithm that provably computes all MPE under certain regularity conditions
- Employ smart discrete programming method to maximize likelihood function over the finite set of equilibria
- Fully robust to multiplicity of MPE
- Relax single-equilibrium-in-data assumption
- Path to estimation of equilibrium selection rules

Nested Recursive Lexicographical Search (NRLS)

- Data from M independent markets from T periods
  $\mathrm{Z} = \{\bar{\mathrm{a}}^{mt}, \bar{\mathrm{x}}^{mt}\}_{m \in \mathcal{M}, t \in \mathcal{T}}$
  Usually assume only one equilibrium is played in the data.
- Denote θ $\left(\mathrm{P}^{\ell}(\theta), \mathrm{V}^{\ell}(\theta)\right) \in SOL(\Psi, \theta)$ the ℓ-the equilibrium

1. Outer loop
   Maximization of the likelihood function w.r.t. to structural parameters θ
   $$\theta^{ML} = \arg\max_{\theta} \mathcal{L}(Z, \theta)$$
2. Inner loop
   Maximization of the likelihood function w.r.t. equilibrium selection
   $$\mathcal{L}(Z, \theta) = \max_{(\mathrm{P}^{\ell}(\theta), \mathrm{V}^{\ell}(\theta) \in SOL(\Psi, \theta)} \frac{1}{M} \sum_{i=1}^{N} \sum_{m=1}^{M} \sum_{t=1}^{T} \log P_i^{\ell}(\bar{a}_i{}^{mt} | \bar{\mathrm{x}}^{mt}; \theta)$$

Max of a function on a discrete set organized into RLS tree

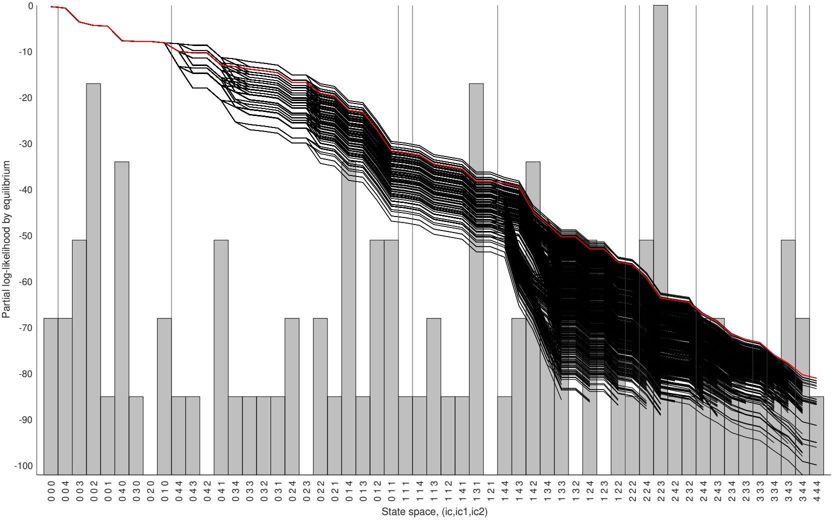
Branch and bound (BnB) method

Land and Doig, 1960 *Econometrica*

- Old method for solving discrete programming problems

1. Form a tree of subdivisions of the set of admissible plans
2. Specify a bounding function representing the best attainable objective on a given subset (branch)
3. Dismiss the subsets of the plans where the bound is below the current best attained value of the objective

BnB and numerical performance of NRLS estimator



Numerical performance and refinements of NRLS estimator

- Numerical performance of NRLS estimator depends crucially on how the data is able to distinguish between different equilibria

- Bounding criterion is deterministic $\rightarrow$ may use statistical criterion to decide whether to extend a given branch or not
- Have to assess potential likelihood contribution of the branches that are not fully extended $\rightarrow$ Vuong closeness test (LR-type test to assess how different two equilibria are given already computed partial likelihood)

$\Rightarrow$ Poly-algorithm with statistical decision rule

Monte Carlo simulations

A

Single equilibrium in the model
Single equilibrium in the data

B

Multiple equilibria in the model
Single equilibrium in the data

C

Multiple equilibria in the model
Multiple equilibria in the data

1. Two-step CCP estimator
2. Nested pseudo-likelihood vs. NRLS estimator
3. MPEC

Implementation details

- Two-step estimator and NPL
  - Matlab unconstraint optimizer (numerical derivatives)
  - CCPs from frequency estimators
  - For NPL max 30 iterations
- MPEC
  - Matlab constraint optimizer (interior-point algorithm)
  - MPEC-VP: Constraints on both values and choice probabilities (as in Egesdal, Lai and Su, 2015)
  - MPEC-P: Constraints in terms of choice probabilities + Hotz-Miller inversion
  - Starting values from two-step estimator
- Estimated parameters $\theta = (k_1, k_2)$
- Sample size: 1000 markets in 5 time periods
- Initial state drawn uniformly over the state space

Monte Carlo A, run 1: no multiplicity

Maximum number of equilibria in the model: 1

Number of equilibria in the data: 1

	PML2step	NPL	MPEC-VP	MPEC-P	NRLS
k1=3.5	3.51893	3.51022	3.50380	3.50380	3.50380
Bias	0.01893	0.01022	0.00380	0.00380	0.00380
MCSD	0.12087	0.12635	0.11573	0.11573	0.11573
k2=0.5	0.50860	0.50658	0.50452	0.50452	0.50452
Bias	0.00860	0.00658	0.00452	0.00452	0.00452
MCSD	0.06460	0.06247	0.05939	0.05939	0.05939
log-likelihood	-1958.176	-1953.406	-1953.327	-1953.327	-1953.327
$\|\|\Psi^{\mathbf{P}}(P) - P\|\|$	0.25285	0.00001	0.00000	0.00000	0.00000
$\|\|\Psi^{\mathbf{V}}(v) - v\|\|$	0.50038	0.00001	0.00000	0.00000	0.00000
Converged,%	100	100	100	100	100
K-L divergence	0.131139	0.005020	0.006770	0.006770	0.006770

- All MLE estimators identical to the last digit
- NPL estimator is approaching MLE

Monte Carlo A, run 2: no multiplicity at true parameter

Maximum number of equilibria in the model: 3

Number of equilibria at true parameter value: 1

Number of equilibria in the data: 1

	PML2step	NPL	MPEC-VP	MPEC-P	NRLS
k1=3.5	3.50467	3.51307	3.49485	3.49318	3.49318
Bias	0.00467	0.01307	-0.00515	-0.00682	-0.00682
MCSD	0.11252	0.00000	0.10193	0.10177	0.10177
k2=0.5	0.50035	0.47394	0.50265	0.50157	0.50157
Bias	0.00035	-0.02606	0.00265	0.00157	0.00157
MCSD	0.05009	0.00000	0.04154	0.04205	0.04205
log-likelihood	-4106.771	-3940.158	-4091.873	-4093.040	-4093.04
$\|\|\Psi^{\mathbf{P}}(P) - P\|\|$	0.41453	0.00001	0.00000	0.00000	0.00000
$\|\|\Psi^{\mathbf{V}}(v) - v\|\|$	1.90182	0.00005	0.00000	0.00000	0.00000
Converged,%	100	1	98	100	100
K-L divergence	0.188551	0.004546	0.002921	0.002921	0.002920

- NPL estimator fails to converge
- MPEC is not affected by “nearby” equilibria with good starting values (PML2step)

Monte Carlo B, run 1: moderate multiplicity

Number of equilibria in the model (at true parameter): 3
Number of equilibria in the data: 1

	PML2step	NPL	MPEC-VP	MPEC-P	NRLS
k1=3.5	3.50081	-	3.72713	3.94941	3.49624
Bias	0.00081	-	0.22713	0.44941	-0.00376
MCSD	0.12050	-	0.85934	1.16633	0.09537
k2=0.5	0.49478	-	0.56166	0.62361	0.49381
Bias	-0.00522	-	0.06166	0.12361	-0.00619
MCSD	0.04317	-	0.25552	0.32488	0.03510
log-likelihood	-4070.035	-	-4080.989	-4121.102	-4049.647
$\|\|\Psi^{\mathbf{P}}(P) - P\|\|$	0.50375	-	0.00000	0.00000	0.00000
$\|\|\Psi^{\mathbf{V}}(v) - v\|\|$	2.83611	-	0.00000	0.00000	0.00000
Converged,%	100	0	100	100	100
K-L divergence	0.304411	-	0.018636	2.302525	0.006314

- NPL estimator fails to converge
- MPEC fails to identify the equilibrium that generated the data (converges to a different MPE) as seen from MCSD and K-L divergence

Monte Carlo B, run 2: higher multiplicity

Number of equilibria in the model (at true parameter): 81

Number of equilibria in the data: 1

	PML2step	NPL	MPEC-VP	MPEC-P	NRLS
k1=3.5	3.51468	-	3.48740	3.49007	3.47786
Bias	0.01468	-	-0.01260	-0.00993	-0.02214
MCSD	0.04844	-	0.02802	0.02929	0.02731
k2=0.5	0.53780	-	0.49197	0.48944	0.49252
Bias	0.03780	-	-0.00803	-0.01056	-0.00748
MCSD	0.03894	-	0.00850	0.01033	0.00404
log-likelihood	-4038.78471	-	-4007.45663	-4010.18139	-3996.45223
$\|\|\Psi^{\mathbf{P}}(P) - P\|\|$	0.68907	-	0.00000	0.00000	0.00000
$\|\|\Psi^{\mathbf{V}}(v) - v\|\|$	5.44052	-	0.00000	0.00000	0.00000
Converged,%	100	0	100	100	100
K-L divergence	0.453917	-	0.278263	0.356678	0.000750

- NPL estimator fails to converge
- MPEC fails to identify the DGP equilibrium (converges to a different MPE)
- With good starting values, does not suffer more with higher multiplicity

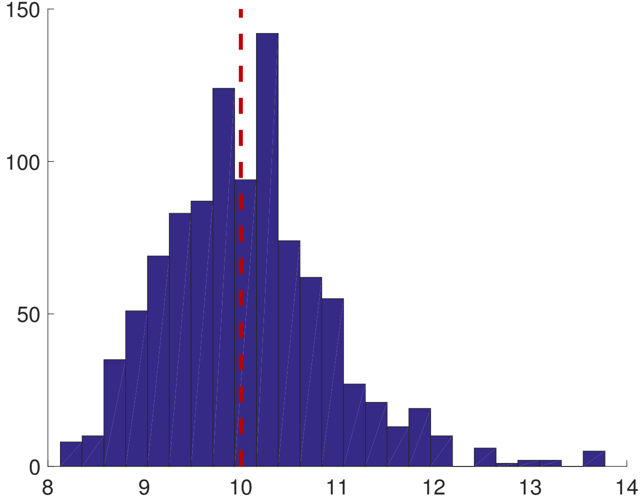
NRLS Monte Carlo setup (C)

- $n = 3$ points on the grid on the grid of costs
- 14 points in state space of the model
- 109 MPE in total

- 1000 random samples from 3 different equilibria (3 markets)
- 100 observations per market/equilibrium
- Uniform distribution over state space $\leftrightarrow$ “ideal” data

- Estimating one parameter in cost function

Distribution of estimated k_1 parameter



MC results and numerical performance of NRLS

1. Average bias and RMSE of the estimates of the cost of investment parameter (true value is 10.0)

$$\text{Bias} = 0.0737$$
$$\text{RMSE} = 0.8712$$

2. Average fraction of MPE computed by BnB relative to RLS

   0.321 (std=0.11635)

3. Average fraction of stages solved by BnB relative to RLS

   0.263 (std=0.09725)

4. All 3 MPE correctly identified by BnB in

   98.4% of runs

Identification of multiple equilibria in the data (C)

- 100 random samples
- 3 market clusters with different equilibria
- 1000 observations per market cluster/equilibrium in 3 time periods

- Among all runs, 93% of equilibria were pin-pointed exactly
- Among the misidentified equilibria, all had deviation in one point of the state space

Conclusions: Bertrand investments model

- Many types of endogenous coordination is possible in equilibrium
  - Leapfrogging (alternating investments)
  - Preemption (investment by cost leader)
  - Duplicative (simultaneous investments)
- Full rent dissipation and monopoly outcomes are supported as MPE.
- Numerous MPE equilibria and "Folk theorem"-like result
- The equilibria are generally inefficient due to over-investment
  - Duplicative or excessively frequent investments

Conclusions: Solution of dynamic games

- When equilibrium is not unique the computation algorithm inadvertently acts as an equilibrium selection mechanism
- When directionality in the state space is present, state recursion algorithm is preferred to time iterations
- Plethora of Markov perfect equilibria poses new challenges:
  - How firms manage to coordinate on a particular equilibrium?
  - Increased difficulties for empirical applications.
  - Daunting perspectives for identification of equilibrium selection rule from the data.
- Estimation of dynamic games with multiple equilibria
  Nested Recursive Lexicographical Search (NRLS)

Contributions and further developments

- NRLS is MLE estimator for dynamic games of a particular type, directional dynamic games (DDGs)
  - Fully robust to multiplicity of equilibria
  - Able to identify multiple equilibria in the data
- Further work on and tests of numerical performance
  - Refinements of the implementation of NRLS (optimization of BnB algorithm)
  - Statistical bounding criterion
- More detailed comparison of existing estimators using leapfrogging game
  - Refine the implementation of MPEC
  - Include recent estimators into the battery (Aguirregabiria and Marcoux, 2019, Bugni and Bunting, 2020)