Algorithmic Collusion and Best-Response Dynamics

Clemens Possnig

University of British Columbia

July 29, 2021

Motivation

- Reinforcement Learning (RL): Algorithm adapts a policy online, while being used, and feedbacks are accrued.
    - Adaptation towards an objective prescribed by designer (minimize some gradient, no regret, etc.)

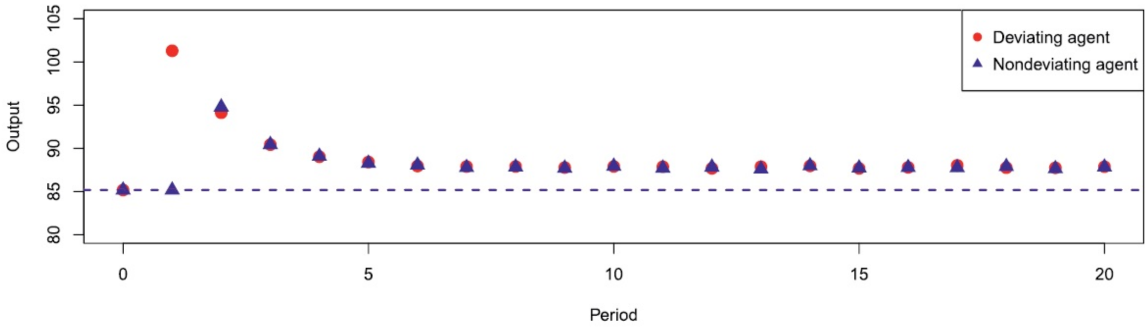
Motivation

- Reinforcement Learning (RL): Algorithm adapts a policy online, while being used, and feedbacks are accrued.
  - Adaptation towards an objective prescribed by designer (minimize some gradient, no regret, etc.)
- Prevalent decision making tool for complex environments.
- Receiving increasing interest from economists: algorithmic pricing.

Motivation

- Reinforcement Learning (RL): Algorithm adapts a policy online, while being used, and feedbacks are accrued.
  - Adaptation towards an objective prescribed by designer (minimize some gradient, no regret, etc.)
- Prevalent decision making tool for complex environments.
- Receiving increasing interest from economists: algorithmic pricing.
- Calzolari et al (2020 AER, 2021 xx): Simulations show algorithmic collusion
- Empirical studies support this (Assad et al 2020).

Calvano et al. 2021: Limiting strategies



Quantity competition with 1-period price memory: impulse response of limiting strategies.

Motivation

So we know algorithms may learn to collude, but how and why?

Research Question

What details of the game and algorithm allow algorithms to learn collusion?

Answering this will

- Give regulators a framework: which parts of the market can be adjusted to deter collusion?
- Aid the design and understanding of cooperative AI in general.

Research Question

What details of the game and algorithm allow algorithms to learn collusion?

Answering this will

- Give regulators a framework: which parts of the market can be adjusted to deter collusion?
- Aid the design and understanding of cooperative AI in general.

Research Question

What details of the game and algorithm allow algorithms to learn collusion?

Answering this will

- Give regulators a framework: which parts of the market can be adjusted to deter collusion?
- Aid the design and understanding of cooperative AI in general.

"The problem of algorithmic collusion:
Analytic approaches are untractable"–G. Calzolari

Research Question

What details of the game and algorithm allow algorithms to learn collusion?

To answer analytically:

Research Question

What details of the game and algorithm allow algorithms to learn collusion?

To answer analytically:

- Focus on specific class of RL: Actor-Critic
- Imperfect Public Monitoring Games.

Contribution: Methodology

A new methodology to study algorithmic collusion analytically:

3 Steps.

1. Stochastic Approximation: Connect strategies learned in the long term to stable equilibria of best-response dynamics.

Contribution: Methodology

A new methodology to study algorithmic collusion analytically:

3 Steps.

1. Stochastic Approximation: Connect strategies learned in the long term to stable equilibria of best-response dynamics.
2. State-space Reduction: General rest points live in a large space.
   Reduce state space, gain clean insights.

Contribution: Methodology

A new methodology to study algorithmic collusion analytically:

3 Steps.

1. Stochastic Approximation: Connect strategies learned in the long term to stable equilibria of best-response dynamics.
2. State-space Reduction: General rest points live in a large space.
   Reduce state space, gain clean insights.
3. Comparative statics: vary a shape parameter and study stability of resulting equilibria.

Contribution: Literature

- Algorithmic Collusion: Calvano et al. (2020, 2021), McKay et al. 2021.

Contribution: Literature

- Algorithmic Collusion: Calvano et al. (2020, 2021), McKay et al. 2021.
- Heuristic learning in Games: Leslie et al. (2020, 2006), Foster and Vohra (1997), many more

Contribution: Literature

- Algorithmic Collusion: Calvano et al. (2020, 2021), McKay et al. 2021.
- Heuristic learning in Games: Leslie et al. (2020, 2006), Foster and Vohra (1997), many more
  - Contribution: State-dependent Best-response Dynamics

Contribution: Literature

- Algorithmic Collusion: Calvano et al. (2020, 2021), McKay et al. 2021.
- Heuristic learning in Games: Leslie et al. (2020, 2006), Foster and Vohra (1997), many more
  - Contribution: State-dependent Best-response Dynamics
- Stochastic Approximation Theory: Leslie et al. (2012), Benaim et al. (2012), etc.

Contribution: Literature

- Algorithmic Collusion: Calvano et al. (2020, 2021), McKay et al. 2021.
- Heuristic learning in Games: Leslie et al. (2020, 2006), Foster and Vohra (1997), many more
  - Contribution: State-dependent Best-response Dynamics
- Stochastic Approximation Theory: Leslie et al. (2012), Benaim et al. (2012), etc.
  - Contribution: Asynchronous stochastic approximation for imperfect public info games

Contribution: Literature

- Algorithmic Collusion: Calvano et al. (2020, 2021), McKay et al. 2021.
- Heuristic learning in Games: Leslie et al. (2020, 2006), Foster and Vohra (1997), many more
  - Contribution: State-dependent Best-response Dynamics
- Stochastic Approximation Theory: Leslie et al. (2012), Benaim et al. (2012), etc.
  - Contribution: Asynchronous stochastic approximation for imperfect public info games
  - Approximation from discrete to continuous action spaces

Contribution: Literature

- Algorithmic Collusion: Calvano et al. (2020, 2021), McKay et al. 2021.
- Heuristic learning in Games: Leslie et al. (2020, 2006), Foster and Vohra (1997), many more
  - Contribution: State-dependent Best-response Dynamics
- Stochastic Approximation Theory: Leslie et al. (2012), Benaim et al. (2012), etc.
  - Contribution: Asynchronous stochastic approximation for imperfect public info games
  - Approximation from discrete to continuous action spaces
  - Connection between asymptotic strategies and stable equilibria

Roadmap

1. The Game
2. Stochastic Approximation
3. Reduction
4. Comparative Statics

Cournot à la Green-Porter

The GP-game is a tuple $\Gamma = \langle I, u_i, A, \Theta, G, \delta \rangle$,

- I is the set of 2 Firms,

Cournot à la Green-Porter

The GP-game is a tuple $\Gamma = \langle I, u_i, A, \Theta, G, \delta \rangle$,

- I is the set of 2 Firms,
- Quantity choice $q_i \in A = [0, M]$.

Cournot à la Green-Porter

The GP-game is a tuple $\Gamma = \langle I, u_i, A, \Theta, G, \delta \rangle$,

- I is the set of 2 Firms,
- Quantity choice $q_i \in A = [0, M]$.
- Demand shocks $\theta \sim G$: $\quad \theta \in \Theta \subseteq \mathbb{R}_+$, G twice differentiable.

Cournot à la Green-Porter

The GP-game is a tuple $\Gamma = \langle I, u_i, A, \Theta, G, \delta \rangle$,

- I is the set of 2 Firms,
- Quantity choice $q_i \in A = [0, M]$.
- Demand shocks $\theta \sim G$: $\theta \in \Theta \subseteq \mathbb{R}_+$, G twice differentiable.
- Price as the public outcome:

$$p(Q, \theta) = \theta + h(Q),$$

where $Q = \sum_i q_i$ and $h : \mathbb{R}_+ \mapsto \mathbb{R}$ s.t. $h' < 0$, $h'' \leq 0$.

Cournot à la Green-Porter

The GP-game is a tuple $\Gamma = \langle I, u_i, A, \Theta, G, \delta \rangle$,

- I is the set of 2 Firms,
- Quantity choice $q_i \in A = [0, M]$.
- Demand shocks $\theta \sim G$: $\theta \in \Theta \subseteq \mathbb{R}_+$, G twice differentiable.
- Price as the public outcome:

$$p(Q, \theta) = \theta + h(Q),$$

  where $Q = \sum_i q_i$ and $h : \mathbb{R}_+ \mapsto \mathbb{R}$ s.t. $h' < 0$, $h'' \leq 0$.
- Symmetric profit functions:

$$u_i(q, p(Q, \theta)) = p(Q, \theta) q_i - c(q_i),$$

Cournot à la Green-Porter

The GP-game is a tuple $\Gamma = \langle I, u_i, A, \Theta, G, \delta \rangle$,

- I is the set of 2 Firms,
- Quantity choice $q_i \in A = [0, M]$.
- Demand shocks $\theta \sim G$: $\theta \in \Theta \subseteq \mathbb{R}_+$, G twice differentiable.
- Price as the public outcome:

$$p(Q, \theta) = \theta + h(Q),$$

  where $Q = \sum_i q_i$ and $h : \mathbb{R}_+ \mapsto \mathbb{R}$ s.t. $h' < 0$, $h'' \leq 0$.
- Symmetric profit functions:

$$u_i(q, p(Q, \theta)) = p(Q, \theta) q_i - c(q_i),$$

- c(q) twice differentiable cost function.

Imperfect Public Monitoring

Players can only observe their own action and the public outcome.

- $h_t = \{p_0, p_1, .., p_{t-1}\}$ is the history of public outcomes (prices), with H_t their set.
- Public Policies $\pi_i : H_t \mapsto A$,
  with $\pi = \times_i \pi_i$ a profile of policies, and $\pi_{-i} = \times_{j \neq i} \pi_j$.

Imperfect Public Monitoring

Players can only observe their own action and the public outcome.

- $h_t = \{p_0, p_1, .., p_{t-1}\}$ is the history of public outcomes (prices), with H_t their set.
- Public Policies $\pi_i : H_t \mapsto A$,
  with $\pi = \times_i \pi_i$ a profile of policies, and $\pi_{-i} = \times_{j \neq i} \pi_j$.
- Objective:

$$W^{(i)}(\pi_i, \pi_{-i}, p_0) = \mathbb{E}_{G,\pi} \sum_{t=0}^{\infty} \delta^t u_i(\pi(h_t))$$

- Value Function:

$$V_{\pi_{-i}}(p) = \max_{\pi_i} W^{(i)}(\pi_i, \pi_{-i}, p)$$

Finite Memory PPE

- Study game with algorithm players: Finite memory required.
- H^k: set of possible truncated public histories from $t-k, \ldots, t-1$ for any t.
- $\Pi_k = \{\pi : \pi_i : H_t^k \mapsto A \ \forall i\}$ set of k-memory public strategies.
- $$BR_i^k(\pi_{-i}) = \text{argmax}_{\pi \in \Pi_k} W^{(i)}(\pi_i, \pi_{-i}, y_0).$$

Finite Memory PPE

- Γ_k: restriction of Γ where strategies are constrained to lie in Π_k.
- Let E be the set of PPEs of Γ. Then $E_k \subseteq E$ is the set of k-memory PPEs.
- $SE_k \subseteq E_k$: Symmetric equi such that $\pi_i(h) = \pi_j(h) \ \ \forall h \in H^k$, all i, j.

Roadmap

1. The Game
2. Stochastic Approximation
3. Reduction
4. Comparative Statics

Stochastic Approximation

Two-Timescale Actor-Critic Algorithms (2AC): Useful for changing environments.

- Actor-Critic: At every period, update two objects:

Stochastic Approximation

Two-Timescale Actor-Critic Algorithms (2AC): Useful for changing environments.

- Actor-Critic: At every period, update two objects:
  - Policy π_t, mapping from states to actions.
    Update towards best action given current action-value estimates.

Stochastic Approximation

Two-Timescale Actor-Critic Algorithms (2AC): Useful for changing environments.

- Actor-Critic: At every period, update two objects:
  - Policy π_t, mapping from states to actions.
    Update towards best action given current action-value estimates.
  - Action-value function $Q_t(s, a)$, which estimates

$$Q_{\pi_{-i}}(s, a) = u(a, \pi_{-i}(s)) + \delta E_{G,(a,\pi_{-i}(s))} \left[V_{\pi_{-i}}(s')\right]$$

Stochastic Approximation

Two-Timescale Actor-Critic Algorithms (2AC): Useful for changing environments.

- Actor-Critic: At every period, update two objects:
  - Policy π_t, mapping from states to actions.
    Update towards best action given current action-value estimates.
  - Action-value function $Q_t(s, a)$, which estimates

$$Q_{\pi_{-i}}(s, a) = u(a, \pi_{-i}(s)) + \delta E_{G,(a,\pi_{-i}(s))}\left[V_{\pi_{-i}}(s')\right]$$

- Two-Timescale: Update policies *an order of magnitude slower* than Q-estimates

Stochastic Approximation

Two-Timescale Actor-Critic Algorithms (2AC): Useful for changing environments.

- Actor-Critic: At every period, update two objects:
  - Policy π_t, mapping from states to actions.
    Update towards best action given current action-value estimates.
  - Action-value function $Q_t(s, a)$, which estimates

$$Q_{\pi_{-i}}(s,a) = u(a, \pi_{-i}(s)) + \delta E_{G,(a,\pi_{-i}(s))}\left[V_{\pi_{-i}}(s')\right]$$

- Two-Timescale: Update policies *an order of magnitude slower* than Q-estimates
  - Policies appear almost stationary to the Q-estimator

Stochastic Approximation

Two-Timescale Actor-Critic Algorithms (2AC): Useful for changing environments.

- Actor-Critic: At every period, update two objects:
  - Policy π_t, mapping from states to actions.
    Update towards best action given current action-value estimates.
  - Action-value function $Q_t(s, a)$, which estimates

$$Q_{\pi_{-i}}(s, a) = u(a, \pi_{-i}(s)) + \delta E_{G,(a,\pi_{-i}(s))}\left[V_{\pi_{-i}}(s')\right]$$

- Two-Timescale: Update policies *an order of magnitude slower* than Q-estimates
  - Policies appear almost stationary to the Q-estimator
  - Given stationary policy, Q-estimator is unbiased and converges

Stochastic Approximation

- We say 2AC-algorithms are k-memory if their state space consists of past k periods prices.
- k-memory 2AC-algorithms induce a dynamic process $\{\pi_t^k\}$ in Π_k.

Stochastic Approximation

- We say 2AC-algorithms are k-memory if their state space consists of past k periods prices.
- k-memory 2AC-algorithms induce a dynamic process $\{\pi_t^k\}$ in Π_k.

Theorem

With positive probability, the limiting strategies of process π_t^k will be an asymptotically stable PPE of Γ_k.
Asymptotic stability is with respect to

$$\dot{\pi}(t) = BR^k(\pi(t)) - \pi(t).$$

Stability

Roadmap

1. The Game
2. Stochastic Approximation
3. Reduction
4. Comparative Statics

Reduction

- Let $\mathcal{R} = \{S_L, S_H\}$ be a partition of H^k and let π be a profile of policies s.t.

$$s_k \in S_L \Rightarrow \ \pi_1(s_k) = \alpha_L; \ \pi_2(s_k) = \beta_L,$$

$$s_k \in S_H \Rightarrow \ \pi_1(s_k) = \alpha_H; \ \pi_2(s_k) = \beta_H$$

Call $\pi = (\pi_1, \pi_2)$ a 2-qty profile.

General

Reduction

- Define a reduced state space $\bar{S} = \{L, H\}$ with strategy choices $\bar{\pi} : \bar{S} \mapsto A$.
- Define injective function $r : S \mapsto \mathcal{R}$, to associate reduced states with original partition.
- Given a game Γ and a partition $\mathcal{R}$ and function r, define a reduced game $\bar{\Gamma}(\mathcal{R})$ using the reduced state space, strategy choices, and transition function implied by the reduction.

Reduction

- Define a reduced state space $\bar{S} = \{L, H\}$ with strategy choices $\bar{\pi} : \bar{S} \mapsto A$.
- Define injective function $r : S \mapsto \mathcal{R}$, to associate reduced states with original partition.
- Given a game Γ and a partition $\mathcal{R}$ and function r, define a reduced game $\bar{\Gamma}(\mathcal{R})$ using the reduced state space, strategy choices, and transition function implied by the reduction.
- Given a 2-qty profile π, let $\bar{\pi}$ be the associated reduced strategy profile:

$$\bar{\pi}(s) = \pi(r(s)) \; \forall \, s \in S.$$

Reduction

Proposition (Equilibrium Reduction)

Suppose π is a PPE of Γ_k. Then the associated $\bar{\pi}$ is a PPE of the reduced game $\bar{\Gamma}$, and vice-versa.

Reduction: Stability

Proposition (Unstable $\Rightarrow$ Unstable)

Suppose $\bar{\pi}$ is a PPE of $\bar{\Gamma}$ and let π be the associated PPE in Γ_k. Then π is unstable if $\bar{\pi}$ is unstable.

Proposition (Uniform $\Rightarrow$ Equivalence)

Let Γ be the GP game with uniformly distributed shocks. Suppose π is a one-memory 2-qty SE in Γ_1, and $\bar{\pi}$ is the associated SE of $\bar{\Gamma}$. Then there exists $\bar{\delta}$ s.t. for all $\delta \geq \bar{\delta}$, π is asymptotically stable if and only if $\bar{\pi}$ is asymptotically stable.

Roadmap

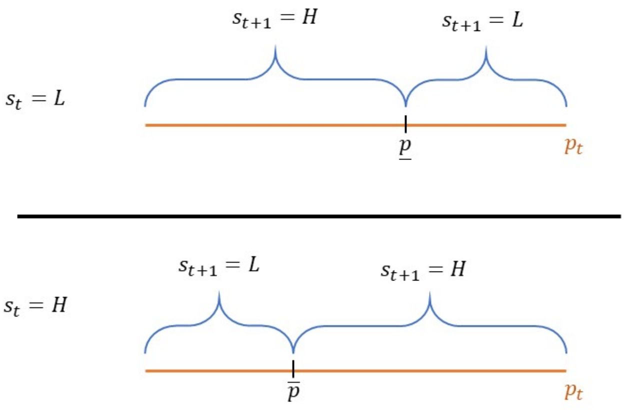
1. The Game
2. Stochastic Approximation
3. Reduction
4. Comparative Statics

Perturbations

Take a reduced state space $\bar{S} = \{L, H\}$ and define the class of threshold-equilibria TE:

$\pi^* \in TE$ if it is a 2-qty profile and there exist $\underline{p}, \overline{p}$ such that

$$
\begin{aligned}
s_t = L,\ p_t > \underline{p} \quad & \Rightarrow \quad s_{t+1} = L \\
s_t = L,\ p_t \leq \underline{p} \quad & \Rightarrow \quad s_{t+1} = H \\
s_t = H,\ p_t \leq \overline{p} \quad & \Rightarrow \quad s_{t+1} = L \\
s_t = H,\ p_t > \overline{p} \quad & \Rightarrow \quad s_{t+1} = H.
\end{aligned}
$$



$s_t = L$
$s_{t+1} = H$
$s_{t+1} = L$
$\underline{p}$
p_t
$s_t = H$
$s_{t+1} = L$
$s_{t+1} = H$
$\overline{p}$
p_t

Green-Porter Game: Perturbations

- Introduce shape parameters
$$\gamma_p, \gamma_c \in Z_x = [1 - x, 1 + x]$$
for some $x > 0$.
- $h(Q) = -Q^{\gamma_p}, \qquad c(q) = q^{\gamma_c}.$
- Index associated GP game Γ as $\Gamma_{\gamma_p,\gamma_c}$ and associated TE_{γ_p,γ_c}.

Green-Porter Game: Perturbations

Proposition

Let $\gamma_p = 1$*. There is* $\bar{x} > 0$ *such that for all* $\gamma_p \in Z_{\bar{x}}$*, there exist threshold-equilibria* $\pi^* \in TE_{1,\gamma_c}$ *of the GP game.*

1. $\gamma_c \in [1 - \bar{x}, 1) \Rightarrow \pi^*$ *is unstable for all* $\pi^* \in TE_{\gamma_p,1}$.
2. $\gamma_c \in (1, 1 + \bar{x}] \Rightarrow \pi^*$ *is stable for all* $\pi^* \in TE_{\gamma_p,1}$.
3. *Static Cournot is always stable.*

Green-Porter Game: Perturbations

Proposition

Let $\gamma_c = 1$. There is $\bar{x} > 0$ such that for all $\gamma_p \in Z_{\bar{x}}$, there exist threshold-equilibria $\pi^* \in TE_{\gamma_p,1}$ of the GP game.

1. $\gamma_p \in [1 - \bar{x}, 1) \Rightarrow \pi^*$ *is unstable for all* $\pi^* \in TE_{\gamma_p,1}$.
2. $\gamma_p \in (1, 1 + \bar{x}] \Rightarrow \pi^*$ *is stable for all* $\pi^* \in TE_{\gamma_p,1}$.
3. *Static Cournot is always stable.*

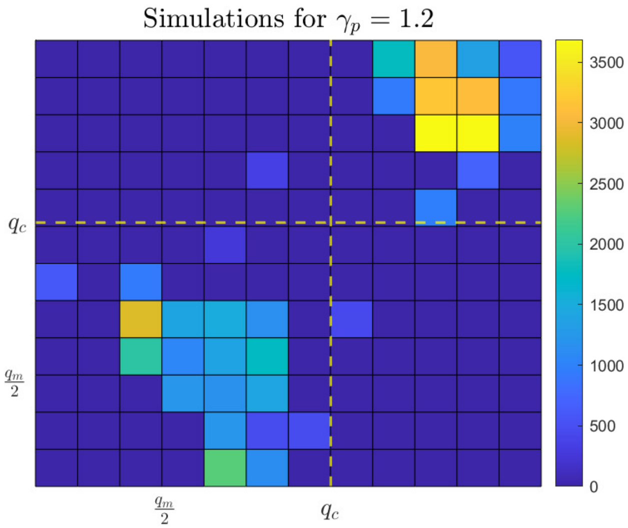


Figure: Last 1000 periods of 50 simulations runs

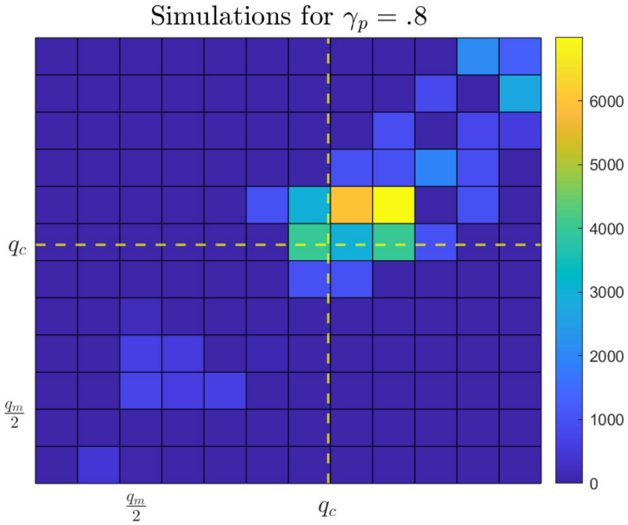


Figure: Last 1000 periods of 50 simulations runs

Conclusion

- An analytic study of algorithmic collusion.
- New methodology.
- Connection of stable equilibria to limiting behavior of algorithms.

Appendix

The Game: Imperfect Public Monitoring

A Game of Imperfect Public Monitoring is a tuple $\Gamma = \langle I, u_i, A, \mathcal{Y}, G, \delta \rangle$, where

- $I = \{1, ..., n\}$ is the set of n players
- $A \subseteq \mathbb{R}$ an interval action space
- $\mathcal{Y}$ is a space of public outcomes
- $u_i : A \times \mathcal{Y} \mapsto \mathbb{R}$ is i's payoff function
- $G(y; a)$ is the twice differentiable cdf of $y \in \mathcal{Y}$ given profile $a \in A^n$.
- $\delta \in (0, 1)$ a discount factor.

Stability

Suppose $\phi_t(x)$ is a solution to a differential system

$$\dot{x} = f(x(t)).$$

Suppose x_0 is a rest point, i.e. $f(x_0) = 0$. Then x_0 is

- Stable if for all $\varepsilon > 0$ there exists $\delta > 0$ s.t.

  $$|x - x_0| < \delta \Rightarrow |\phi_t(x) - x_0| < \varepsilon$$

  holds for all $t \geq 0$.
- Asymptotically stable if it is stable and there is a constant $a > 0$ s.t.

  $$|x - x_0| < a \Rightarrow \lim_{t\to\infty} |\phi_t(x) - x_0| = 0.$$

Back

Reduction: General

Suppose $\pi^* \in PPE_k$ is symmetric such that on path, only finitely many actions are ever played:

- there exist finite sets Z_i such that $\pi_i^*(h) \in Z_i \subseteq A$ for all i, $h \in H^k$.
- Let $R_i \subset \mathcal{P}[H^k]$ be the partition of the state space such that
$$R_i = \{\pi_i^{*-1}(z) : z \in Z_i\},$$
with
$$\pi_i^{*-1}(z) = \{h \in H^k : \pi_i^*(h) = z\}.$$

Back

Reduction: General

- Symmetry implies $R_i = R$ for all i.
- Define new game $\bar{\Gamma}$ with state space S such that $|S| = |R| < \infty$
- There exists an injective mapping $r : S \mapsto R$ associating reduced states with partition elements in R
- Define policy $\bar{\pi}$ as

$$\bar{\pi}(s) = \pi^*(r(s)), \ \forall\, s \in S.$$

- $\bar{\pi}$ is a PPE of $\bar{\Gamma}$.

Back