Multi-agent learning

Uncoupled learning and NE

Gerard Vreeswijk, Intelligent Systems Group, Computer Science Department, Faculty of Sciences, Utrecht University, The Netherlands.

Tuesday 18th June, 2019

Contents

Contents

Contents

1. Finding Nash equilibria (not necessary through learning).

Contents

1. Finding Nash equilibria (not necessary through learning).
   - The computational complexity of finding Nash equilibria.

Contents

1. Finding Nash equilibria (not necessary through learning).
   - The computational complexity of finding Nash equilibria.
   - Some algorithms to find Nash equilibria.

Contents

1. Finding Nash equilibria (not necessary through learning).
   - The computational complexity of finding Nash equilibria.
   - Some algorithms to find Nash equilibria.
2. The possibility of finding Nash equilibria through uncoupled learning.

Contents

1. Finding Nash equilibria (not necessary through learning).
   - The computational complexity of finding Nash equilibria.
   - Some algorithms to find Nash equilibria.
2. The possibility of finding Nash equilibria through uncoupled learning.
   - Work of Sergiu Hart and Andreu Mas-Colell.

Contents

1. Finding Nash equilibria (not necessary through learning).
   - The computational complexity of finding Nash equilibria.
   - Some algorithms to find Nash equilibria.
2. The possibility of finding Nash equilibria through uncoupled learning.
   - Work of Sergiu Hart and Andreu Mas-Colell.
3. The possibility of finding Nash equilibria through completely uncoupled learning.

Contents

1. Finding Nash equilibria (not necessary through learning).
   - The computational complexity of finding Nash equilibria.
   - Some algorithms to find Nash equilibria.
2. The possibility of finding Nash equilibria through uncoupled learning.
   - Work of Sergiu Hart and Andreu Mas-Colell.
3. The possibility of finding Nash equilibria through completely uncoupled learning.
   - Work of Dean Foster, H. Peyton Young, and Yakov Babichenko.

Contents

1. Finding Nash equilibria (not necessary through learning).
   - The computational complexity of finding Nash equilibria.
   - Some algorithms to find Nash equilibria.
2. The possibility of finding Nash equilibria through uncoupled learning.
   - Work of Sergiu Hart and Andreu Mas-Colell.
3. The possibility of finding Nash equilibria through completely uncoupled learning.
   - Work of Dean Foster, H. Peyton Young, and Yakov Babichenko.

     — Regret testing.

Contents

1. Finding Nash equilibria (not necessary through learning).
   - The computational complexity of finding Nash equilibria.
   - Some algorithms to find Nash equilibria.
2. The possibility of finding Nash equilibria through uncoupled learning.
   - Work of Sergiu Hart and Andreu Mas-Colell.
3. The possibility of finding Nash equilibria through completely uncoupled learning.
   - Work of Dean Foster, H. Peyton Young, and Yakov Babichenko.

     — Regret testing.

     — Trial and error learning.

The computational complexity of finding Nash equilibria

The computational complexity of finding Nash equilibria (not necessary through learning)

Nash equilibria

A Nash equilibrium is a strategy profile where all strategies are best responses to their counter strategy profiles.

Nash equilibria

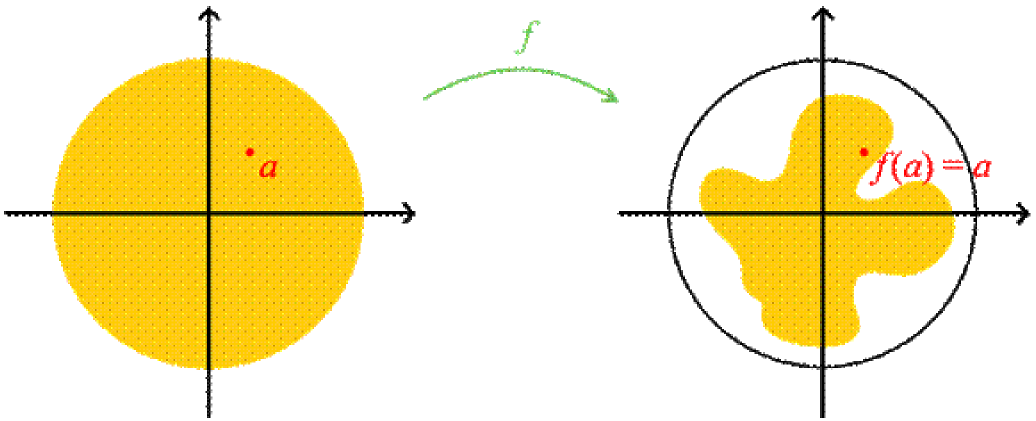
A Nash equilibrium is a strategy profile where all strategies are best responses to their counter strategy profiles.

Theorem (Nash, 1951). Every game with a finite number of players and action profiles possesses at least one Nash equilibrium.

Nash equilibria

A Nash equilibrium is a strategy profile where all strategies are best responses to their counter strategy profiles.

Theorem (Nash, 1951). Every game with a finite number of players and action profiles possesses at least one Nash equilibrium.

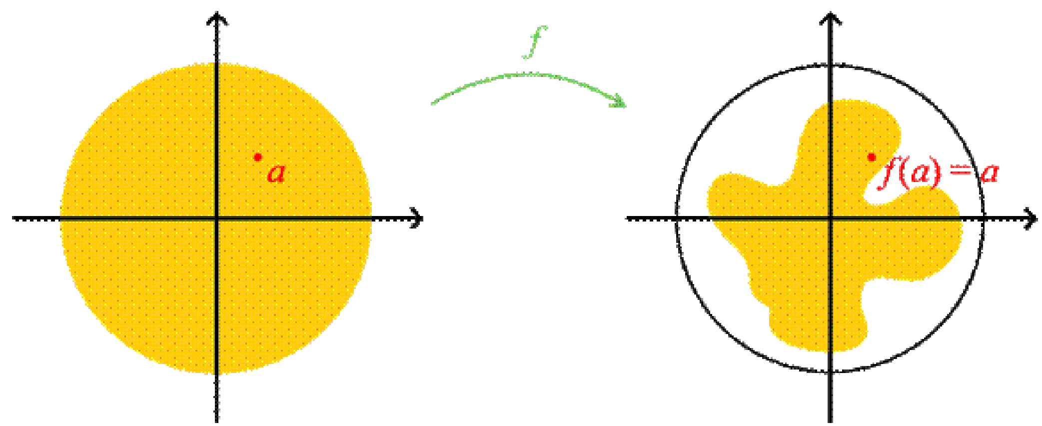


Nash equilibria

A Nash equilibrium is a strategy profile where all strategies are best responses to their counter strategy profiles.

Theorem (Nash, 1951). Every game with a finite number of players and action profiles possesses at least one Nash equilibrium.

Its proof uses Brouwer's fixed point theorem (1911): any continuous map from a compact and convex set to itself has a fixed point.

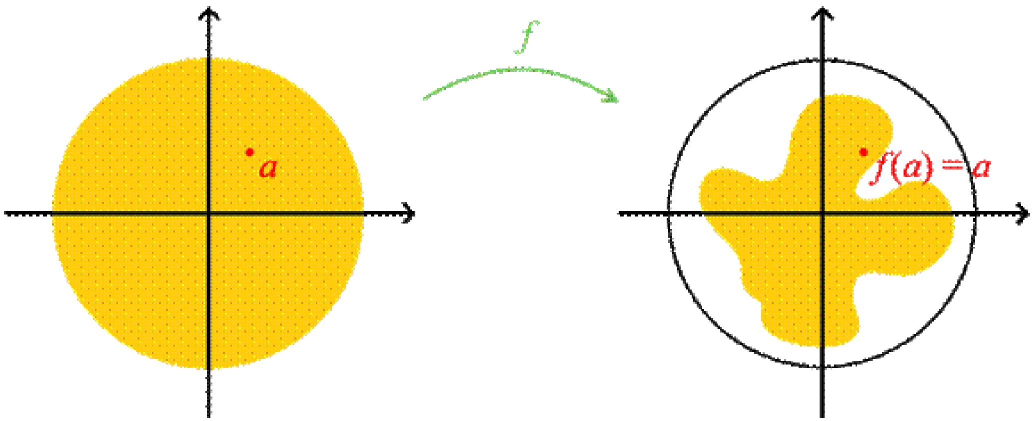


Nash equilibria

A Nash equilibrium is a strategy profile where all strategies are best responses to their counter strategy profiles.

Theorem (Nash, 1951). Every game with a finite number of players and action profiles possesses at least one Nash equilibrium.

Its proof uses Brouwer's fixed point theorem (1911): any continuous map from a compact and convex set to itself has a fixed point.



The proof of Brouwer's theorem is non-constructive. So is the proof of Nash's theorem!

How easy / difficult is it to find Nash equilibria?

How easy / difficult is it to find Nash equilibria?

- A mixed strategy is a best response if and only if all pure strategies in its support are best responses.

How easy / difficult is it to find Nash equilibria?

- A mixed strategy is a best response if and only if all pure strategies in its support are best responses.
- Finding a Nash equilibrium is therefore a combinatorial problem: it entails finding appropriate support sets for players.

How easy / difficult is it to find Nash equilibria?

- A mixed strategy is a best response if and only if all pure strategies in its support are best responses.
- Finding a Nash equilibrium is therefore a combinatorial problem: it entails finding appropriate support sets for players.
- Is finding Nash equilibria an NP-complete problem?

How easy / difficult is it to find Nash equilibria?

- A mixed strategy is a best response if and only if all pure strategies in its support are best responses.
- Finding a Nash equilibrium is therefore a combinatorial problem: it entails finding appropriate support sets for players.
- Is finding Nash equilibria an NP-complete problem? No. NP-complete problems are yes/no questions.

How easy / difficult is it to find Nash equilibria?

- A mixed strategy is a best response if and only if all pure strategies in its support are best responses.
- Finding a Nash equilibrium is therefore a combinatorial problem: it entails finding appropriate support sets for players.
- Is finding Nash equilibria an NP-complete problem? No. NP-complete problems are yes/no questions.
- Inspect Nash's proof to distill an algorithm?

How easy / difficult is it to find Nash equilibria?

- A mixed strategy is a best response if and only if all pure strategies in its support are best responses.
- Finding a Nash equilibrium is therefore a combinatorial problem: it entails finding appropriate support sets for players.
- Is finding Nash equilibria an NP-complete problem? No. NP-complete problems are yes/no questions.
- Inspect Nash's proof to distill an algorithm? Not much hope: Nash's proof uses Brouwer's FPT, the proof of which is non-constructive.

How easy / difficult is it to find Nash equilibria?

- A mixed strategy is a best response if and only if all pure strategies in its support are best responses.
- Finding a Nash equilibrium is therefore a combinatorial problem: it entails finding appropriate support sets for players.
- Is finding Nash equilibria an NP-complete problem? No. NP-complete problems are yes/no questions.
- Inspect Nash's proof to distill an algorithm? Not much hope: Nash's proof uses Brouwer's FPT, the proof of which is non-constructive.
- **Finding NE is PPAD-hard**. PPAD is a complexity class of search problems defined in 1994. (In 2005: finding NE is PPAD-complete.)

How easy / difficult is it to find Nash equilibria?

- A mixed strategy is a best response if and only if all pure strategies in its support are best responses.
- Finding a Nash equilibrium is therefore a combinatorial problem: it entails finding appropriate support sets for players.
- Is finding Nash equilibria an NP-complete problem? No. NP-complete problems are yes/no questions.
- Inspect Nash's proof to distill an algorithm? Not much hope: Nash's proof uses Brouwer's FPT, the proof of which is non-constructive.
- **Finding NE is PPAD-hard**. PPAD is a complexity class of search problems defined in 1994. (In 2005: finding NE is PPAD-complete.)

– C.H. Papadimitriou (2007): "The complexity of finding Nash equilibria" in: Algorithmic Game Theory.
– C. Daskalakis *et al.* (2009): "The complexity of computing a Nash equilibrium" in: SIAM Journal on Computing.

NP-complete questions about Nash equilibria

Theorem (Gilboa and Zemel, 1989). The following *are* NP-complete problems, even for symmetric games. Given a two-player game in strategic form, does it have

NP-complete questions about Nash equilibria

Theorem (Gilboa and Zemel, 1989). The following *are* NP-complete problems, even for symmetric games. Given a two-player game in strategic form, does it have

- at least two Nash equilibria?

NP-complete questions about Nash equilibria

Theorem (Gilboa and Zemel, 1989). The following *are* NP-complete problems, even for symmetric games. Given a two-player game in strategic form, does it have

- at least two Nash equilibria?
- a NE in which player 1 has utility at least a given amount?

NP-complete questions about Nash equilibria

Theorem (Gilboa and Zemel, 1989). The following *are* NP-complete problems, even for symmetric games. Given a two-player game in strategic form, does it have

- at least two Nash equilibria?
- a NE in which player 1 has utility at least a given amount?
- a NE in which the two players have total utility at least a given amount?

NP-complete questions about Nash equilibria

Theorem (Gilboa and Zemel, 1989). The following *are* NP-complete problems, even for symmetric games. Given a two-player game in strategic form, does it have

- at least two Nash equilibria?
- a NE in which player 1 has utility at least a given amount?
- a NE in which the two players have total utility at least a given amount?
- a NE with support of size greater than a given number?

NP-complete questions about Nash equilibria

Theorem (Gilboa and Zemel, 1989). The following *are* NP-complete problems, even for symmetric games. Given a two-player game in strategic form, does it have

- at least two Nash equilibria?
- a NE in which player 1 has utility at least a given amount?
- a NE in which the two players have total utility at least a given amount?
- a NE with support of size greater than a given number?
- a NE whose support contains strategy s_i?

NP-complete questions about Nash equilibria

Theorem (Gilboa and Zemel, 1989). The following *are* NP-complete problems, even for symmetric games. Given a two-player game in strategic form, does it have

- at least two Nash equilibria?
- a NE in which player 1 has utility at least a given amount?
- a NE in which the two players have total utility at least a given amount?
- a NE with support of size greater than a given number?
- a NE whose support contains strategy s_i?
- a NE whose support does not contain strategy s_i?

NP-complete questions about Nash equilibria

Theorem (Gilboa and Zemel, 1989). The following *are* NP-complete problems, even for symmetric games. Given a two-player game in strategic form, does it have

- at least two Nash equilibria?
- a NE in which player 1 has utility at least a given amount?
- a NE in which the two players have total utility at least a given amount?
- a NE with support of size greater than a given number?
- a NE whose support contains strategy s_i?
- a NE whose support does not contain strategy s_i? etc., etc.

NP-complete questions about Nash equilibria

Theorem (Gilboa and Zemel, 1989). The following *are* NP-complete problems, even for symmetric games. Given a two-player game in strategic form, does it have

- at least two Nash equilibria?
- a NE in which player 1 has utility at least a given amount?
- a NE in which the two players have total utility at least a given amount?
- a NE with support of size greater than a given number?
- a NE whose support contains strategy s_i?
- a NE whose support does not contain strategy s_i? etc., etc.

Conitzer and Sandholm (2008) give simple proofs by reducing (3-) satisfiability problems to such questions.

Conitzer, Vincent, and Tuomas Sandholm. "New complexity results about Nash equilibria." Games and Economic Behaviour 63.2 (2008): 621-641.

Gambit: software to find Nash equilibria

Gambit is a library of game theory software and tools.

```
 $ ls -l
total 42760
-rwxr-xr-x 1 Gerard None 3179712 May 15 2017 gambit-enummixed
-rwxr-xr-x 1 Gerard None 3818349 May 15 2017 gambit-enumpoly
-rwxr-xr-x 1 Gerard None 2724401 May 15 2017 gambit-enumpure
-rwxr-xr-x 1 Gerard None 2819270 May 15 2017 gambit-gnm
-rwxr-xr-x 1 Gerard None 2811320 May 15 2017 gambit-ipa
-rwxr-xr-x 1 Gerard None 3185637 May 15 2017 gambit-lcp
-rwxr-xr-x 1 Gerard None 2748940 May 15 2017 gambit-liap
-rwxr-xr-x 1 Gerard None 2813000 May 15 2017 gambit-logit
-rwxr-xr-x 1 Gerard None 3093175 May 15 2017 gambit-lp
-rwxr-xr-x 1 Gerard None 2752730 May 15 2017 gambit-simpdiv
```

Gambit: software to find Nash equilibria

Gambit is a library of game theory software and tools.

```
 $ ls -l
total 42760
-rwxr-xr-x 1 Gerard None 3179712 May 15 2017 gambit-enummixed
-rwxr-xr-x 1 Gerard None 3818349 May 15 2017 gambit-enumpoly
-rwxr-xr-x 1 Gerard None 2724401 May 15 2017 gambit-enumpure
-rwxr-xr-x 1 Gerard None 2819270 May 15 2017 gambit-gnm
-rwxr-xr-x 1 Gerard None 2811320 May 15 2017 gambit-ipa
-rwxr-xr-x 1 Gerard None 3185637 May 15 2017 gambit-lcp
-rwxr-xr-x 1 Gerard None 2748940 May 15 2017 gambit-liap
-rwxr-xr-x 1 Gerard None 2813000 May 15 2017 gambit-logit
-rwxr-xr-x 1 Gerard None 3093175 May 15 2017 gambit-lp
-rwxr-xr-x 1 Gerard None 2752730 May 15 2017 gambit-simpdiv
```

- Construct finite games in normal form or games in extensive form.

Gambit: software to find Nash equilibria

Gambit is a library of game theory software and tools.

```
 $ ls -l
total 42760
-rwxr-xr-x 1 Gerard None 3179712 May 15 2017 gambit-enummixed
-rwxr-xr-x 1 Gerard None 3818349 May 15 2017 gambit-enumpoly
-rwxr-xr-x 1 Gerard None 2724401 May 15 2017 gambit-enumpure
-rwxr-xr-x 1 Gerard None 2819270 May 15 2017 gambit-gnm
-rwxr-xr-x 1 Gerard None 2811320 May 15 2017 gambit-ipa
-rwxr-xr-x 1 Gerard None 3185637 May 15 2017 gambit-lcp
-rwxr-xr-x 1 Gerard None 2748940 May 15 2017 gambit-liap
-rwxr-xr-x 1 Gerard None 2813000 May 15 2017 gambit-logit
-rwxr-xr-x 1 Gerard None 3093175 May 15 2017 gambit-lp
-rwxr-xr-x 1 Gerard None 2752730 May 15 2017 gambit-simpdiv
```

- Construct finite games in normal form or games in extensive form.
- Compute (mixed) Nash equilibria.

Gambit: software to find Nash equilibria

Gambit is a library of game theory software and tools.

```
 $ ls -l
total 42760
-rwxr-xr-x 1 Gerard None 3179712 May 15 2017 gambit-enummixed
-rwxr-xr-x 1 Gerard None 3818349 May 15 2017 gambit-enumpoly
-rwxr-xr-x 1 Gerard None 2724401 May 15 2017 gambit-enumpure
-rwxr-xr-x 1 Gerard None 2819270 May 15 2017 gambit-gnm
-rwxr-xr-x 1 Gerard None 2811320 May 15 2017 gambit-ipa
-rwxr-xr-x 1 Gerard None 3185637 May 15 2017 gambit-lcp
-rwxr-xr-x 1 Gerard None 2748940 May 15 2017 gambit-liap
-rwxr-xr-x 1 Gerard None 2813000 May 15 2017 gambit-logit
-rwxr-xr-x 1 Gerard None 3093175 May 15 2017 gambit-lp
-rwxr-xr-x 1 Gerard None 2752730 May 15 2017 gambit-simpdiv
```

- Construct finite games in normal form or games in extensive form.
- Compute (mixed) Nash equilibria.
- Analyse dominance structures.

Gambit: software to find Nash equilibria

Gambit is a library of game theory software and tools.

```
 $ ls -l
total 42760
-rwxr-xr-x 1 Gerard None 3179712 May 15 2017 gambit-enummixed
-rwxr-xr-x 1 Gerard None 3818349 May 15 2017 gambit-enumpoly
-rwxr-xr-x 1 Gerard None 2724401 May 15 2017 gambit-enumpure
-rwxr-xr-x 1 Gerard None 2819270 May 15 2017 gambit-gnm
-rwxr-xr-x 1 Gerard None 2811320 May 15 2017 gambit-ipa
-rwxr-xr-x 1 Gerard None 3185637 May 15 2017 gambit-lcp
-rwxr-xr-x 1 Gerard None 2748940 May 15 2017 gambit-liap
-rwxr-xr-x 1 Gerard None 2813000 May 15 2017 gambit-logit
-rwxr-xr-x 1 Gerard None 3093175 May 15 2017 gambit-lp
-rwxr-xr-x 1 Gerard None 2752730 May 15 2017 gambit-simpdiv
```

- Construct finite games in normal form or games in extensive form.
- Compute (mixed) Nash equilibria.
- Analyse dominance structures.
- Cross-platform.

Gambit: software to find Nash equilibria

Gambit is a library of game theory software and tools.

```
 $ ls -l
total 42760
-rwxr-xr-x 1 Gerard None 3179712 May 15 2017 gambit-enummixed
-rwxr-xr-x 1 Gerard None 3818349 May 15 2017 gambit-enumpoly
-rwxr-xr-x 1 Gerard None 2724401 May 15 2017 gambit-enumpure
-rwxr-xr-x 1 Gerard None 2819270 May 15 2017 gambit-gnm
-rwxr-xr-x 1 Gerard None 2811320 May 15 2017 gambit-ipa
-rwxr-xr-x 1 Gerard None 3185637 May 15 2017 gambit-lcp
-rwxr-xr-x 1 Gerard None 2748940 May 15 2017 gambit-liap
-rwxr-xr-x 1 Gerard None 2813000 May 15 2017 gambit-logit
-rwxr-xr-x 1 Gerard None 3093175 May 15 2017 gambit-lp
-rwxr-xr-x 1 Gerard None 2752730 May 15 2017 gambit-simpdiv
```

- Construct finite games in normal form or games in extensive form.
- Compute (mixed) Nash equilibria.
- Analyse dominance structures.
- Cross-platform.
- It has a wxWidgets GUI, command-line tools and a scripting API.

Finding Nash equilibria through multi-agent learning

MAL and Nash equilibria

MAL and Nash equilibria

There are two important motives to study Nash equilibria in MAL.

MAL and Nash equilibria

There are two important motives to study Nash equilibria in MAL.

1. In MAL, convergence to a Nash equilibrium is a natural desideratum.

   Suppose strategies converge. If this would not be a Nash equilibrium, then at least one player does not play a best reply $\sim$ at least some learning algorithm is broken.

MAL and Nash equilibria

There are two important motives to study Nash equilibria in MAL.

1. In MAL, convergence to a Nash equilibrium is a natural desideratum.

   Suppose strategies converge. If this would not be a Nash equilibrium, then at least one player does not play a best reply $\sim$ at least some learning algorithm is broken.

2. MAL would be of great help if it can put to use to find Nash equilibria.

MAL and Nash equilibria

There are two important motives to study Nash equilibria in MAL.

1. In MAL, convergence to a Nash equilibrium is a natural desideratum.

   Suppose strategies converge. If this would not be a Nash equilibrium, then at least one player does not play a best reply $\sim$ at least some learning algorithm is broken.

2. MAL would be of great help if it can put to use to find Nash equilibria.

Unfortunately, so far it seems that MAL and Nash equilibria do not form a happy combination.

MAL and Nash equilibria

There are two important motives to study Nash equilibria in MAL.

1. In MAL, convergence to a Nash equilibrium is a natural desideratum.

   Suppose strategies converge. If this would not be a Nash equilibrium, then at least one player does not play a best reply $\sim$ at least some learning algorithm is broken.

2. MAL would be of great help if it can put to use to find Nash equilibria.

Unfortunately, so far it seems that MAL and Nash equilibria do not form a happy combination.

- The simplicity of many MAL algorithms does not seem in line with the complexity (i.e., PPAD) of finding NE.

MAL and Nash equilibria

There are two important motives to study Nash equilibria in MAL.

1. In MAL, convergence to a Nash equilibrium is a natural desideratum.

   Suppose strategies converge. If this would not be a Nash equilibrium, then at least one player does not play a best reply $\sim$ at least some learning algorithm is broken.

2. MAL would be of great help if it can put to use to find Nash equilibria.

Unfortunately, so far it seems that MAL and Nash equilibria do not form a happy combination.

- The simplicity of many MAL algorithms does not seem in line with the complexity (i.e., PPAD) of finding NE.
- MAL seems to yield correlated equilibria at best:

MAL and Nash equilibria

There are two important motives to study Nash equilibria in MAL.

1. In MAL, convergence to a Nash equilibrium is a natural desideratum.

   Suppose strategies converge. If this would not be a Nash equilibrium, then at least one player does not play a best reply $\sim$ at least some learning algorithm is broken.

2. MAL would be of great help if it can put to use to find Nash equilibria.

Unfortunately, so far it seems that MAL and Nash equilibria do not form a happy combination.

- The simplicity of many MAL algorithms does not seem in line with the complexity (i.e., PPAD) of finding NE.
- MAL seems to yield correlated equilibria at best: the history of play works as a correlation device.

The possibility to learn Nash equilibria

Is it possible to learn (pure/mixed/ϵ) Nash equilibria through learning?

The possibility to learn Nash equilibria

Is it possible to learn (pure/mixed/ϵ) Nash equilibria through learning?

- Coupled.
  If a player knows the entire game? In principle yes: let it use one of above-mentioned "1-player" algorithms.

The possibility to learn Nash equilibria

Is it possible to learn (pure/mixed/ϵ) Nash equilibria through learning?

- Coupled.
  If a player knows the entire game? In principle yes: let it use one of above-mentioned "1-player" algorithms.
  Is that what we mean by learning?

The possibility to learn Nash equilibria

Is it possible to learn (pure/mixed/ϵ) Nash equilibria through learning?

- Coupled.
  If a player knows the entire game? In principle yes: let it use one of above-mentioned “1-player” algorithms.
  Is that what we mean by learning? Probably not.

The possibility to learn Nash equilibria

Is it possible to learn (pure/mixed/ϵ) Nash equilibria through learning?

- Coupled.
  If a player knows the entire game? In principle yes: let it use one of above-mentioned "1-player" algorithms.
  Is that what we mean by learning? Probably not.

- Uncoupled.
  If a player doesn't know and can't observe the payoffs of its opponents?

The possibility to learn Nash equilibria

Is it possible to learn (pure/mixed/ϵ) Nash equilibria through learning?

- Coupled.
  If a player knows the entire game? In principle yes: let it use one of above-mentioned "1-player" algorithms.
  Is that what we mean by learning? Probably not.

- Uncoupled.
  If a player doesn't know and can't observe the payoffs of its opponents? Unlikely:

The possibility to learn Nash equilibria

Is it possible to learn (pure/mixed/ϵ) Nash equilibria through learning?

- Coupled.
  If a player knows the entire game? In principle yes: let it use one of above-mentioned "1-player" algorithms.
  Is that what we mean by learning? Probably not.

- Uncoupled.
  If a player doesn't know and can't observe the payoffs of its opponents? Unlikely: No-regret, fictitious play, replicator dynamic, etc. are examples of uncoupled learning

The possibility to learn Nash equilibria

Is it possible to learn (pure/mixed/ϵ) Nash equilibria through learning?

- Coupled.
  If a player knows the entire game? In principle yes: let it use one of above-mentioned "1-player" algorithms.
  Is that what we mean by learning? Probably not.

- Uncoupled.
  If a player doesn't know and can't observe the payoffs of its opponents? Unlikely: No-regret, fictitious play, replicator dynamic, etc. are examples of uncoupled learning—but they don't converge to Nash equilibria.

The possibility to learn Nash equilibria

Is it possible to learn (pure/mixed/ϵ) Nash equilibria through learning?

- Coupled.
  If a player knows the entire game? In principle yes: let it use one of above-mentioned "1-player" algorithms.
  Is that what we mean by learning? Probably not.

- Uncoupled.
  If a player doesn't know and can't observe the payoffs of its opponents? Unlikely: No-regret, fictitious play, replicator dynamic, etc. are examples of uncoupled learning—but they don't converge to Nash equilibria.

- Radically uncoupled (a.k.a. Radically uncoupled).
  If, in addition, a player doesn't know and can't observe the actions of its opponents?

The possibility to learn Nash equilibria

Is it possible to learn (pure/mixed/ϵ) Nash equilibria through learning?

- Coupled.
  If a player knows the entire game? In principle yes: let it use one of above-mentioned "1-player" algorithms.
  Is that what we mean by learning? Probably not.

- Uncoupled.
  If a player doesn't know and can't observe the payoffs of its opponents? Unlikely: No-regret, fictitious play, replicator dynamic, etc. are examples of uncoupled learning—but they don't converge to Nash equilibria.

- Radically uncoupled (a.k.a. Radically uncoupled).
  If, in addition, a player doesn't know and can't observe the actions of its opponents? Unlikely:

The possibility to learn Nash equilibria

Is it possible to learn (pure/mixed/ϵ) Nash equilibria through learning?

- Coupled.
  If a player knows the entire game? In principle yes: let it use one of above-mentioned "1-player" algorithms.
  Is that what we mean by learning? Probably not.

- Uncoupled.
  If a player doesn't know and can't observe the payoffs of its opponents? Unlikely: No-regret, fictitious play, replicator dynamic, etc. are examples of uncoupled learning—but they don't converge to Nash equilibria.

- Radically uncoupled (a.k.a. Radically uncoupled).
  If, in addition, a player doesn't know and can't observe the actions of its opponents? Unlikely: Reinforcement learning and satisficing play are examples of radically uncoupled learning

The possibility to learn Nash equilibria

Is it possible to learn (pure/mixed/ϵ) Nash equilibria through learning?

- Coupled.
  If a player knows the entire game? In principle yes: let it use one of above-mentioned "1-player" algorithms.
  Is that what we mean by learning? Probably not.

- Uncoupled.
  If a player doesn't know and can't observe the payoffs of its opponents? Unlikely: No-regret, fictitious play, replicator dynamic, etc. are examples of uncoupled learning—but they don't converge to Nash equilibria.

- Radically uncoupled (a.k.a. Radically uncoupled).
  If, in addition, a player doesn't know and can't observe the actions of its opponents? Unlikely: Reinforcement learning and satisficing play are examples of radically uncoupled learning—but they don't converge to Nash equilibria.

Notions and concepts

Notions and concepts

LEARNING

Notions and concepts

LEARNING

m-**Recall.** Players are of bounded memory m.

Notions and concepts

LEARNING

m**-Recall.** Players are of bounded memory m. May depend on ϵ, etc.

Notions and concepts

LEARNING

m**-Recall.** Players are of bounded memory m. May depend on ϵ, etc.

Finite memory. The learning algorithm can be implemented by a finite automaton with n states, which may depend on etc.

Notions and concepts

LEARNING

m**-Recall.** Players are of bounded memory m. May depend on ϵ, etc.

Finite memory. The learning algorithm can be implemented by a finite automaton with n states, which may depend on etc.

Uncoupled. No knowledge of opponent's payoffs and can't observe them.

Notions and concepts

LEARNING

m**-Recall.** Players are of bounded memory m. May depend on ϵ, etc.

Finite memory. The learning algorithm can be implemented by a finite automaton with n states, which may depend on etc.

Uncoupled. No knowledge of opponent's payoffs and can't observe them.

Completely uncoupled. Also no knowledge of opponent's actions and can't observe them.

Notions and concepts

LEARNING

m-Recall. Players are of bounded memory m. May depend on ϵ, etc.

Finite memory. The learning algorithm can be implemented by a finite automaton with n states, which may depend on etc.

Uncoupled. No knowledge of opponent's payoffs and can't observe them.

Completely uncoupled. Also no knowledge of opponent's actions and can't observe them.

RESTRICTIONS ON PAYOFFS

Notions and concepts

LEARNING

m-Recall. Players are of bounded memory m. May depend on ϵ, etc.

Finite memory. The learning algorithm can be implemented by a finite automaton with n states, which may depend on etc.

Uncoupled. No knowledge of opponent's payoffs and can't observe them.

Completely uncoupled. Also no knowledge of opponent's actions and can't observe them.

RESTRICTIONS ON PAYOFFS

Generic game. Payoffs can be perturbed without introducing new equilibria or removing existing ones.

Notions and concepts

LEARNING

m-Recall. Players are of bounded memory m. May depend on ϵ, etc.

Finite memory. The learning algorithm can be implemented by a finite automaton with n states, which may depend on etc.

Uncoupled. No knowledge of opponent's payoffs and can't observe them.

Completely uncoupled. Also no knowledge of opponent's actions and can't observe them.

RESTRICTIONS ON PAYOFFS

Generic game. Payoffs can be perturbed without introducing new equilibria or removing existing ones.

EQUILIBRIA

Notions and concepts

LEARNING

m-**Recall.** Players are of bounded memory m. May depend on ϵ, etc.

Finite memory. The learning algorithm can be implemented by a finite automaton with n states, which may depend on etc.

Uncoupled. No knowledge of opponent's payoffs and can't observe them.

Completely uncoupled. Also no knowledge of opponent's actions and can't observe them.

RESTRICTIONS ON PAYOFFS

Generic game. Payoffs can be perturbed without introducing new equilibria or removing existing ones.

EQUILIBRIA

Pure equilibria. Combination of pure strategies in a Nash-equilibrium.

Notions and concepts

LEARNING

m-**Recall.** Players are of bounded memory m. May depend on ϵ, etc.

Finite memory. The learning algorithm can be implemented by a finite automaton with n states, which may depend on etc.

Uncoupled. No knowledge of opponent's payoffs and can't observe them.

Completely uncoupled. Also no knowledge of opponent's actions and can't observe them.

RESTRICTIONS ON PAYOFFS

Generic game. Payoffs can be perturbed without introducing new equilibria or removing existing ones.

EQUILIBRIA

Pure equilibria. Combination of pure strategies in a Nash-equilibrium.

ϵ-**Equilibria.** Combination of mixed strategies not more than ϵ from a Nash-equilibrium.

Notions and concepts

LEARNING

m-Recall. Players are of bounded memory m. May depend on ϵ, etc.

Finite memory. The learning algorithm can be implemented by a finite automaton with n states, which may depend on etc.

Uncoupled. No knowledge of opponent's payoffs and can't observe them.

Completely uncoupled. Also no knowledge of opponent's actions and can't observe them.

RESTRICTIONS ON PAYOFFS

Generic game. Payoffs can be perturbed without introducing new equilibria or removing existing ones.

EQUILIBRIA

Pure equilibria. Combination of pure strategies in a Nash-equilibrium.

ϵ-Equilibria. Combination of mixed strategies not more than ϵ from a Nash-equilibrium.

Limit frequency. 1: all of the time, a.s; $1 - \epsilon$ of the time, a.s.

Overview of learning NE in MAL (column "Result")

	Conditions	Uncoupled	Restrictions	Equilibrium	Limit freq.	Result nr.	Result	Who	When	
1.	1-recall	yes		pure	1	Th. 1	$-$	Hart *et al.*	2006	
2.	1-recall, 2p	yes	generic	pure	1	Pr. 2	$+$	Hart *et al.*	2006	
3.	2-recall	yes		pure	1	Th. 3	$+$	Hart *et al.*	2006	
4.	fin. recall	yes		mix-ϵ	$1-\epsilon$ (m)	Pr. 4	$+$	Hart *et al.*	2006	
5.	fin. recall	yes		mix-ϵ	$1-\epsilon$ (j)	Th. 5	$+$	Hart *et al.*	2006	
6.	fin. recall	yes		mix-ϵ	$1-\epsilon$ (b)	Th. 6	$-$	Hart *et al.*	2006	
7.	fin. mem	yes		mix-ϵ	$1-\epsilon$ (b)	Th. 7	$+$	Hart *et al.*	2006	
8.	fin. mem	yes		mix-ϵ	1		$+$	slides Bab'ko	2013 –	Th. X of Hart *et al.*?
9.	fin. mem	radically		pure	1	Tb. 1	$-$	Babichenko	2012	
10.	fin. mem	radically	generic	pure	1	Tb. 3	$-$	Babichenko	2012	
11.	fin. mem	radically		pure	$1-\epsilon$	Tb. 5	$-$	Babichenko	2012	
12.	fin. mem	radically	generic	pure	$1-\epsilon$	Tb. 7	$+$	H.P. Young	2009 –	k. a. trial and error learning
13.	fin. mem	radically		mix-ϵ	1	Tb. 2	$-$	Babichenko	2012	
14.	fin. mem	radically	generic	mix-ϵ	1	Tb. 4	$+$	Babichenko	2012	
15.	fin. mem	radically		mix-ϵ	$1-\epsilon$	Tb. 6	$+$	Babichenko	2012	
16.	fin. mem	radically	generic	mix-ϵ	$1-\epsilon$	Tb. 8	$+$	Germano *et al.*	2007 –	regret testing (Foster *et al.*, 2006)

(m) marginal frequencies; (j) joint frequencies; (b) behaviour frequencies

Work of S. Hart and A. Mas-Colell

"Stochastic uncoupled dynamics and Nash equilibrium" in: *Games and Economic Behavior*, **57** (2006), pp 286-303.

The possibility of learning pure equilibria

Naive algorithm to learn pure equilibria

Naive algorithm to learn pure equilibria

Naive algorithm to learn pure equilibria

IDEA

- Let Row and Col play pure strategies x and y, and let A be the payoff matrix.
- The scalar $x^T Ay$ equals Row's expected payoff for x.
- The vector Ay equals Row's expected payoffs per action.
- If $x^T Ay < \max\{Ay\}$, then x wasn't optimal.

Naive algorithm to learn pure equilibria

IDEA

- Let Row and Col play pure strategies x and y, and let A be the payoff matrix.
- The scalar $x^T Ay$ equals Row's expected payoff for x.
- The vector Ay equals Row's expected payoffs per action.
- If $x^T Ay < \max\{Ay\}$, then x wasn't optimal.

ALGORITHM

Naive algorithm to learn pure equilibria

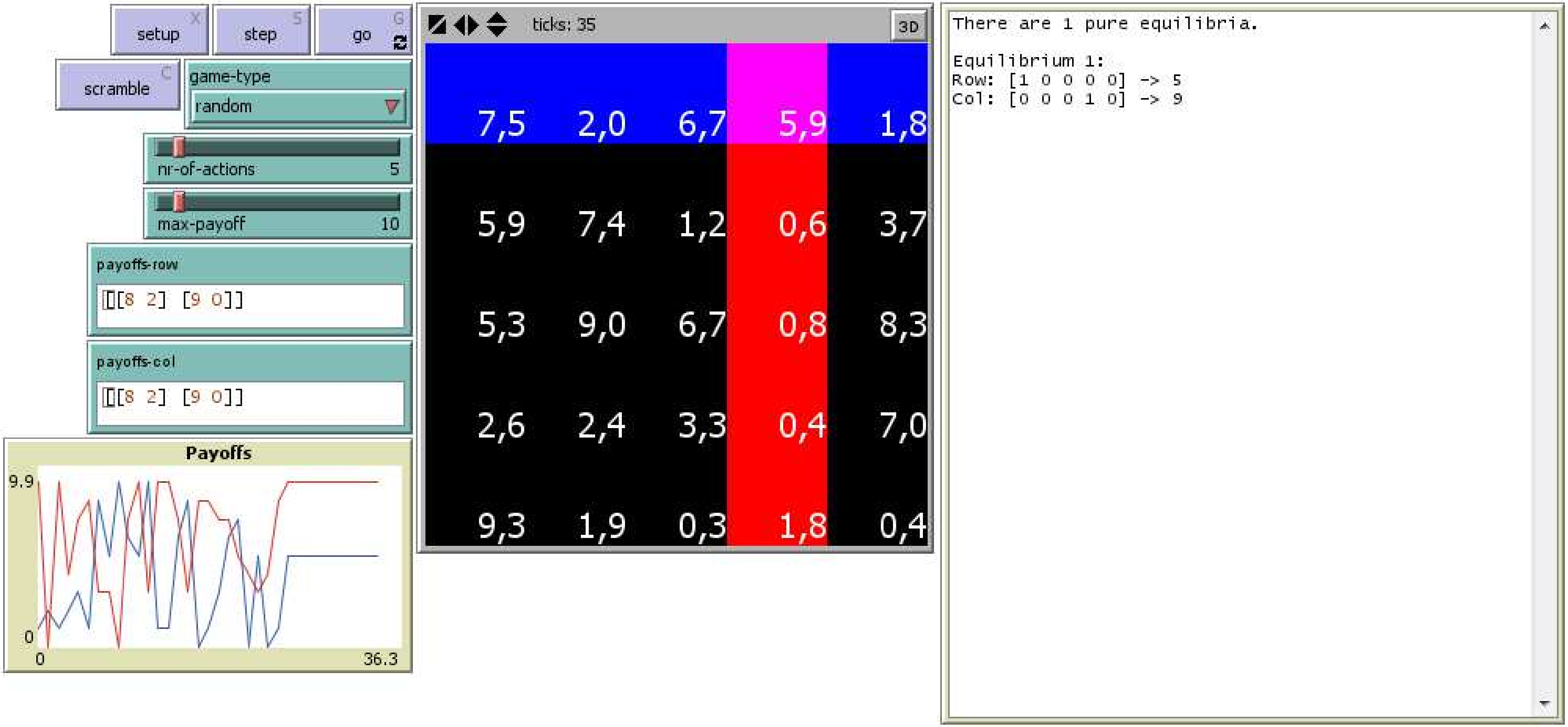
IDEA

- Let Row and Col play pure strategies x and y, and let A be the payoff matrix.
- The scalar $x^T Ay$ equals Row's expected payoff for x.
- The vector Ay equals Row's expected payoffs per action.
- If $x^T Ay < \max\{Ay\}$, then x wasn't optimal.

ALGORITHM

- Repeat: randomise current action if it wasn't a best reply.

Naive algorithm seems to work . . .



Learning pure equilibria with recall one (entry #1)

Learning pure equilibria with recall one (entry #1)

Theorem 1. There are no uncoupled, 1-recall strategies that guarantee a.s. convergence of play to pure NE of the stage game (in all games where such equilibria exist).

Learning pure equilibria with recall one (entry #1)

Theorem 1. There are no uncoupled, 1-recall strategies that guarantee a.s. convergence of play to pure NE of the stage game (in all games where such equilibria exist).

Proof. Consider

	A	B	C
A	1, 0	0, 1	1, 0
B	0, 1	1, 0	1, 0
C	0, 1	0, 1	1, 1

Learning pure equilibria with recall one (entry #1)

Theorem 1. There are no uncoupled, 1-recall strategies that guarantee a.s. convergence of play to pure NE of the stage game (in all games where such equilibria exist).

Proof. Consider

	A	B	C
A	1,0	0,1	1,0
B	0,1	1,0	1,0
C	0,1	0,1	1,1

The only pure NE is (C, C).

Learning pure equilibria with recall one (entry #1)

Theorem 1. There are no uncoupled, 1-recall strategies that guarantee a.s. convergence of play to pure NE of the stage game (in all games where such equilibria exist).

Proof. Consider

	A	B	C
A	1, 0	0, 1	1, 0
B	0, 1	1, 0	1, 0
C	0, 1	0, 1	1, 1

The only pure NE is (C, C). Claim: this equilibrium cannot be reached from any other action profile.

Learning pure equilibria with recall one (entry #1)

Theorem 1. There are no uncoupled, 1-recall strategies that guarantee a.s. convergence of play to pure NE of the stage game (in all games where such equilibria exist).

Proof. Consider

	A	B	C
A	1, 0	0, 1	1, 0
B	0, 1	1, 0	1, 0
C	0, 1	0, 1	1, 1

The only pure NE is (C, C). Claim: this equilibrium cannot be reached from any other action profile.

- For all action profiles, there is at least one player who best replies.

Learning pure equilibria with recall one (entry #1)

Theorem 1. There are no uncoupled, 1-recall strategies that guarantee a.s. convergence of play to pure NE of the stage game (in all games where such equilibria exist).

Proof. Consider

	A	B	C
A	1, 0	0, 1	1, 0
B	0, 1	1, 0	1, 0
C	0, 1	0, 1	1, 1

The only pure NE is (C, C). Claim: this equilibrium cannot be reached from any other action profile.

- For all action profiles, there is at least one player who best replies.
- If neither player plays C, then only one changes strategy.

Learning pure equilibria with recall one (entry #1)

Theorem 1. There are no uncoupled, 1-recall strategies that guarantee a.s. convergence of play to pure NE of the stage game (in all games where such equilibria exist).

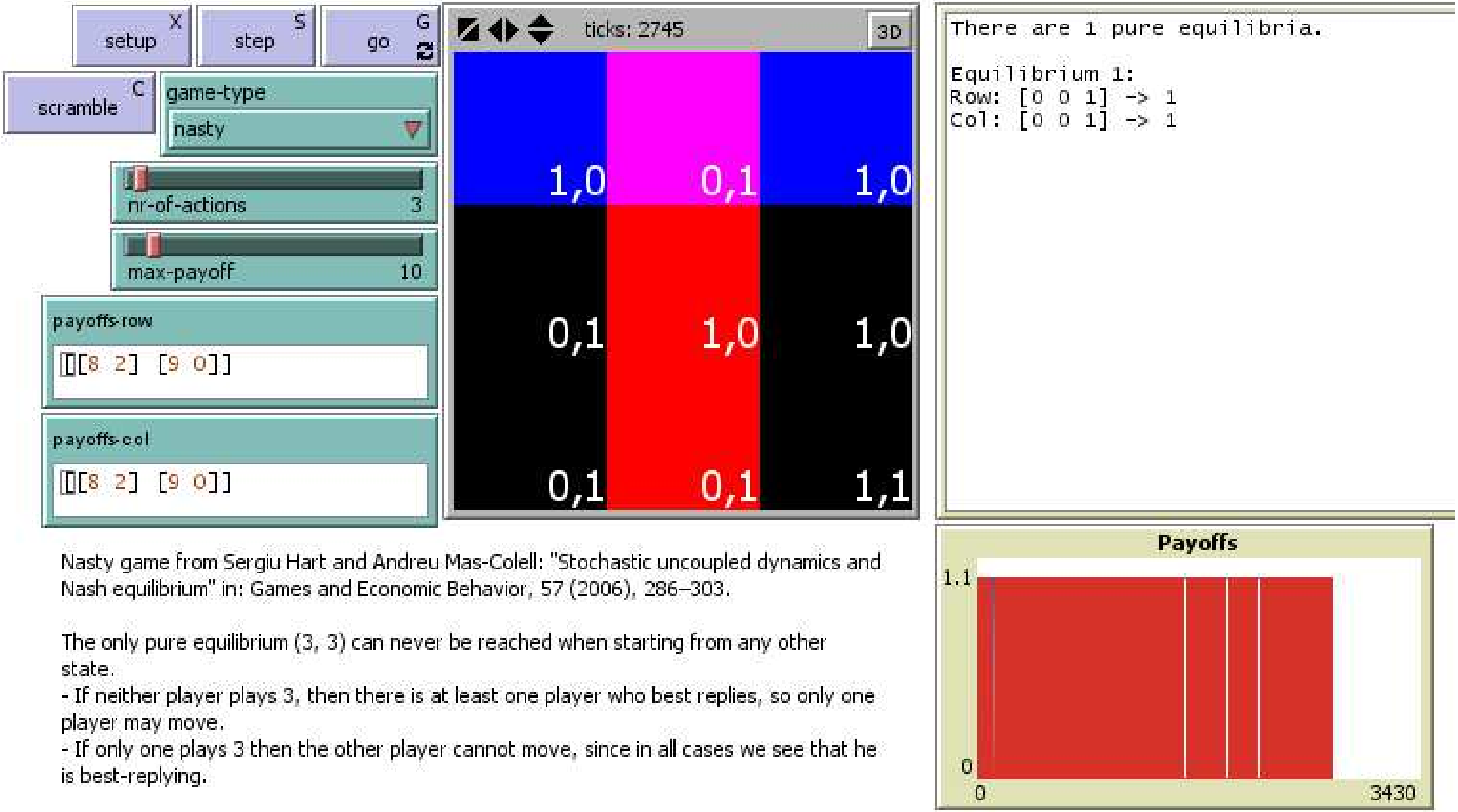
Proof. Consider

	A	B	C
A	1,0	0,1	1,0
B	0,1	1,0	1,0
C	0,1	0,1	1,1

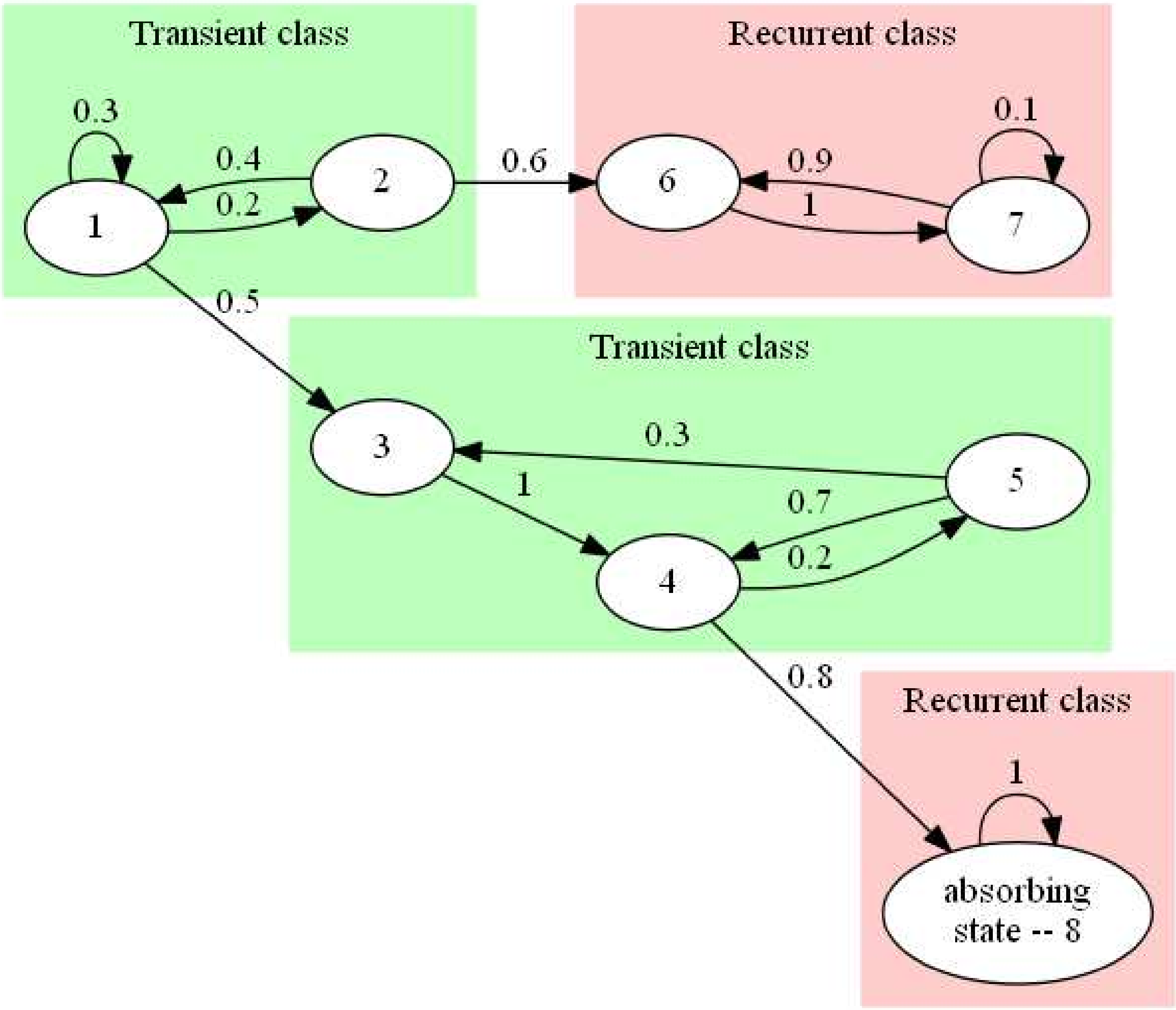
The only pure NE is (C, C). Claim: this equilibrium cannot be reached from any other action profile.

- For all action profiles, there is at least one player who best replies.
- If neither player plays C, then only one changes strategy.
- If one plays C, the other gives a best reply (and does not change strategy). □

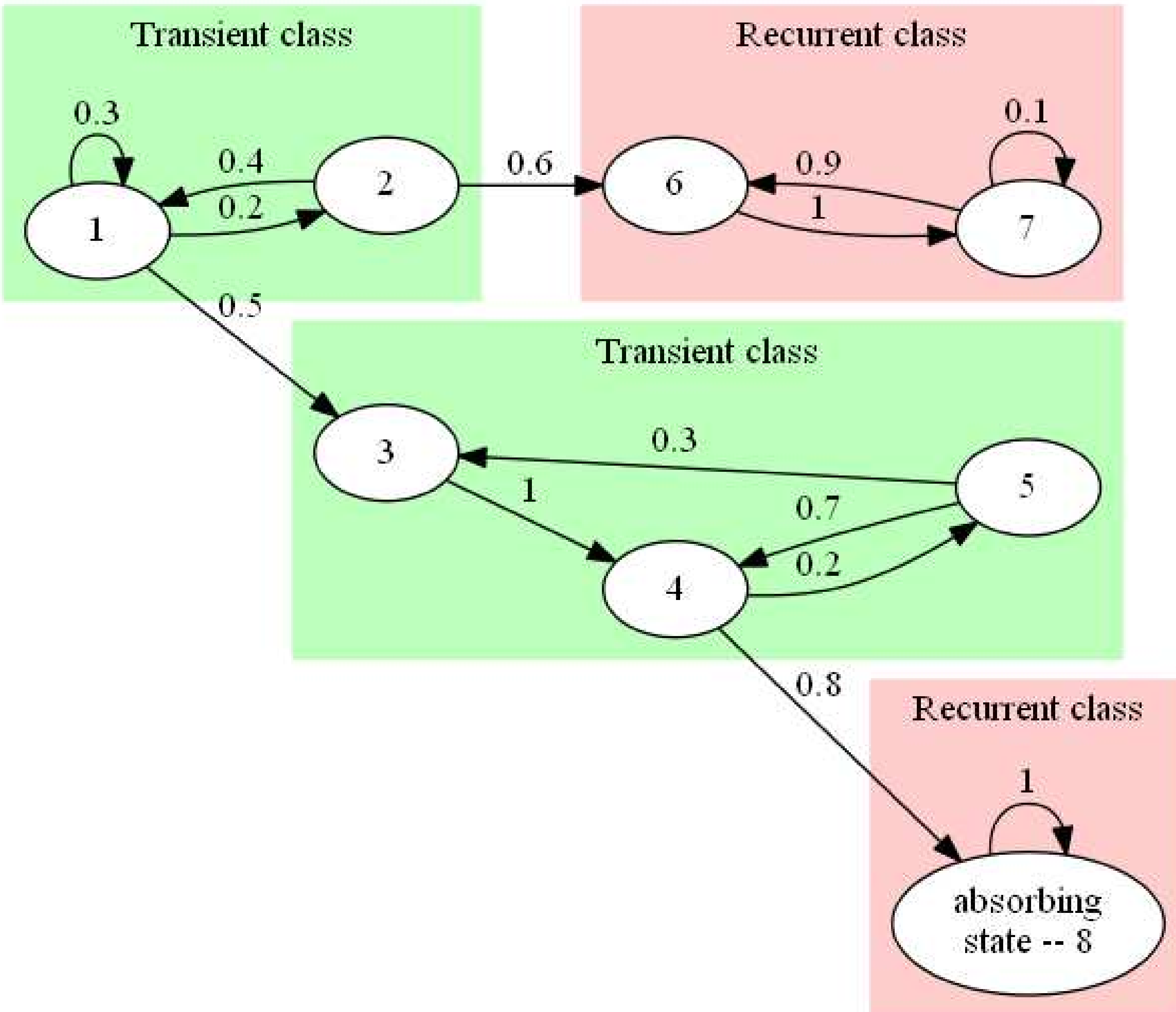
Impossibility of learning equilibria with recall one



Markov chain = a probabilistic process among states

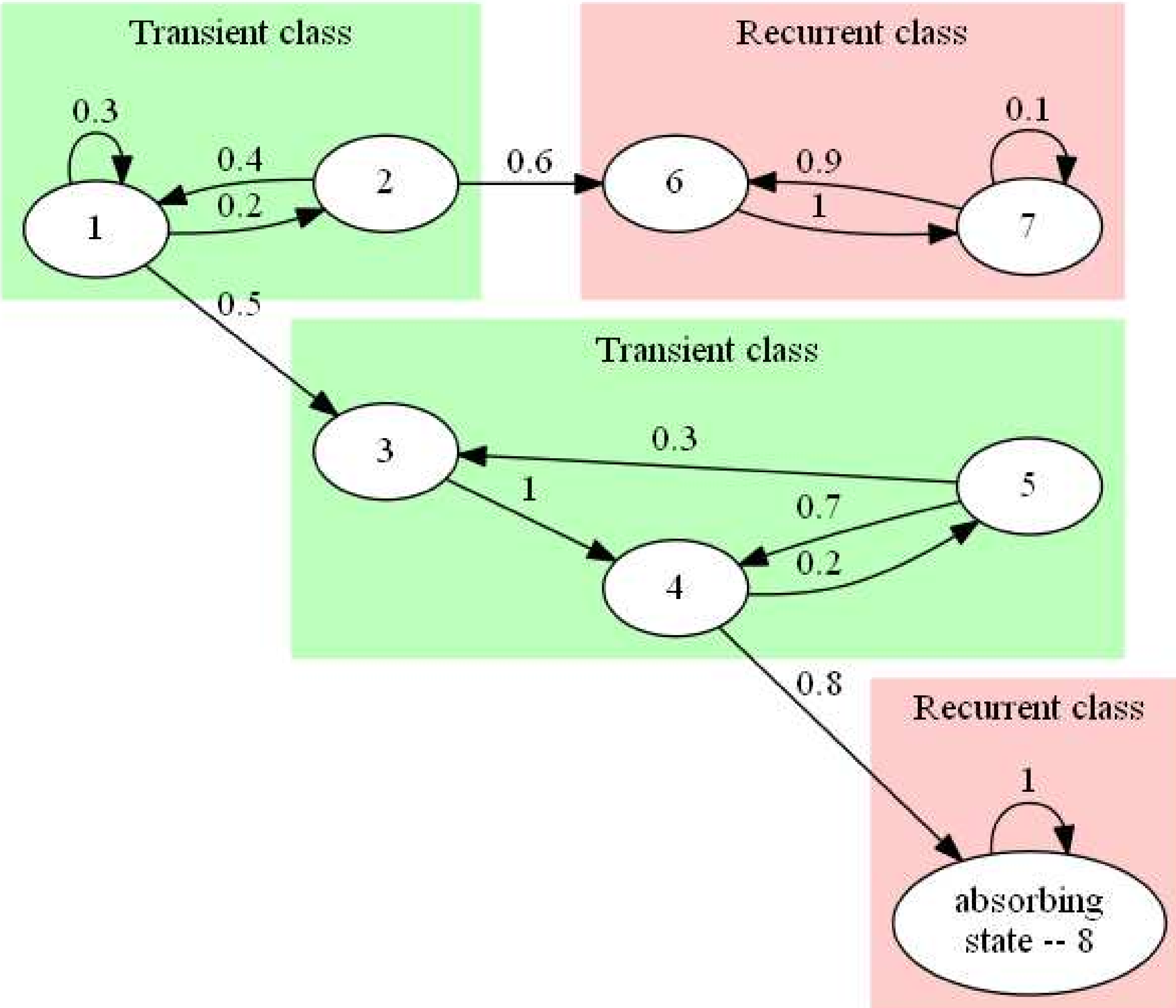


Markov chain = a probabilistic process among states



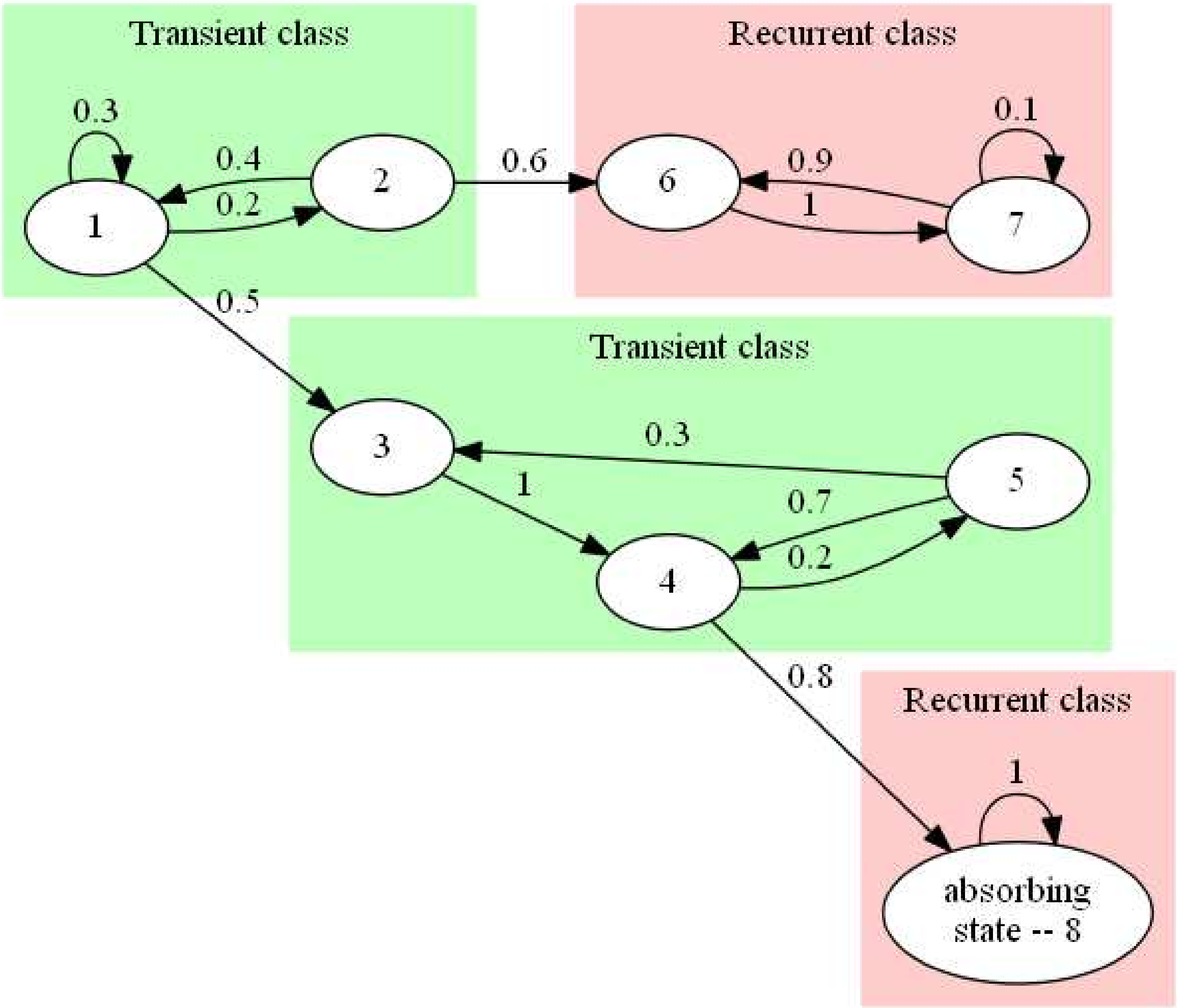
- Numbers represent probabilities.

Markov chain = a probabilistic process among states



- Numbers represent probabilities.
- Probabilities of outgoing edges should add up to one.

Markov chain = a probabilistic process among states



- Numbers represent probabili-ties.
- Probabilities of outgoing edges should add up to one.
- A process always ends up in one of the recurrent classes.

Generic and non-degenerate games

Generic and non-degenerate games

Definition. A game is called generic if its payoffs can be perturbed without introducing new equilibria or removing existing ones.

Generic and non-degenerate games

Definition. A game is called generic if its payoffs can be perturbed without introducing new equilibria or removing existing ones.

- Small changes in payoffs $\Rightarrow$ small changes in equilibria.

Generic and non-degenerate games

Definition. A game is called generic if its payoffs can be perturbed without introducing new equilibria or removing existing ones.

- Small changes in payoffs $\Rightarrow$ small changes in equilibria.
- In practice, generic games are games without ties in payoffs.

Generic and non-degenerate games

Definition. A game is called generic if its payoffs can be perturbed without introducing new equilibria or removing existing ones.

- Small changes in payoffs $\Rightarrow$ small changes in equilibria.
- In practice, generic games are games without ties in payoffs.
- Almost all games are generic ($\sim$ almost all real numbers are irrational).

Generic and non-degenerate games

Definition. A game is called generic if its payoffs can be perturbed without introducing new equilibria or removing existing ones.

- Small changes in payoffs $\Rightarrow$ small changes in equilibria.
- In practice, generic games are games without ties in payoffs.
- Almost all games are generic ($\sim$ almost all real numbers are irrational).

Definition. A game is called non-degenerate if the number of pure best responses to a mixed strategy never exceeds the size of its support.

Generic and non-degenerate games

Definition. A game is called generic if its payoffs can be perturbed without introducing new equilibria or removing existing ones.

- Small changes in payoffs $\Rightarrow$ small changes in equilibria.
- In practice, generic games are games without ties in payoffs.
- Almost all games are generic ($\sim$ almost all real numbers are irrational).

Definition. A game is called non-degenerate if the number of pure best responses to a mixed strategy never exceeds the size of its support.

- It can be proven that generic games are non-degenerate.

Generic and non-degenerate games

Definition. A game is called generic if its payoffs can be perturbed without introducing new equilibria or removing existing ones.

- Small changes in payoffs $\Rightarrow$ small changes in equilibria.
- In practice, generic games are games without ties in payoffs.
- Almost all games are generic ($\sim$ almost all real numbers are irrational).

Definition. A game is called non-degenerate if the number of pure best responses to a mixed strategy never exceeds the size of its support.

- It can be proven that generic games are non-degenerate.
- With non-degenerate games, it follows that pure strategies have unique (hence pure) best responses.

Learning pure equilibria with recall one (entry #2)

Learning pure equilibria with recall one (entry #2)

Theorem 2. There exist uncoupled, 1-recall response rules that guarantee a.s. convergence of play to pure NE of the stage game in every two-player generic game where such equilibria exist.

Learning pure equilibria with recall one (entry #2)

Theorem 2. There exist uncoupled, 1-recall response rules that guarantee a.s. convergence of play to pure NE of the stage game in every two-player generic game where such equilibria exist.

Proof.

Learning pure equilibria with recall one (entry #2)

Theorem 2. There exist uncoupled, 1-recall response rules that guarantee a.s. convergence of play to pure NE of the stage game in every two-player generic game where such equilibria exist.

Proof. Let players remember the last action profile played. Response rule: persist at best replies, randomise else.

Learning pure equilibria with recall one (entry #2)

Theorem 2. There exist uncoupled, 1-recall response rules that guarantee a.s. convergence of play to pure NE of the stage game in every two-player generic game where such equilibria exist.

Proof. Let players remember the last action profile played. Response rule: persist at best replies, randomise else. Gives a Markov proces.

Learning pure equilibria with recall one (entry #2)

Theorem 2. There exist uncoupled, 1-recall response rules that guarantee a.s. convergence of play to pure NE of the stage game in every two-player generic game where such equilibria exist.

Proof. Let players remember the last action profile played. Response rule: persist at best replies, randomise else. Gives a Markov proces.

- Each pure NE equilibrium is an absorbing state.

Learning pure equilibria with recall one (entry #2)

Theorem 2. There exist uncoupled, 1-recall response rules that guarantee a.s. convergence of play to pure NE of the stage game in every two-player generic game where such equilibria exist.

Proof. Let players remember the last action profile played. Response rule: persist at best replies, randomise else. Gives a Markov proces.

- Each pure NE equilibrium is an absorbing state.
- Otherwise, ≥ 1 player randomises, possibly into his part of a NE.

Learning pure equilibria with recall one (entry #2)

Theorem 2. There exist uncoupled, 1-recall response rules that guarantee a.s. convergence of play to pure NE of the stage game in every two-player generic game where such equilibria exist.

Proof. Let players remember the last action profile played. Response rule: persist at best replies, randomise else. Gives a Markov proces.

- Each pure NE equilibrium is an absorbing state.
- Otherwise, ≥ 1 player randomises, possibly into his part of a NE.
- Meanwhile, it is possible that the other players play the same action.

Learning pure equilibria with recall one (entry #2)

Theorem 2. There exist uncoupled, 1-recall response rules that guarantee a.s. convergence of play to pure NE of the stage game in every two-player generic game where such equilibria exist.

Proof. Let players remember the last action profile played. Response rule: persist at best replies, randomise else. Gives a Markov proces.

- Each pure NE equilibrium is an absorbing state.
- Otherwise, ≥ 1 player randomises, possibly into his part of a NE.
- Meanwhile, it is possible that the other players play the same action.

  If this same action is the other part of the NE $\Rightarrow$ in an absorbing state

Learning pure equilibria with recall one (entry #2)

Theorem 2. There exist uncoupled, 1-recall response rules that guarantee a.s. convergence of play to pure NE of the stage game in every two-player generic game where such equilibria exist.

Proof. Let players remember the last action profile played. Response rule: persist at best replies, randomise else. Gives a Markov proces.

- Each pure NE equilibrium is an absorbing state.
- Otherwise, ≥ 1 player randomises, possibly into his part of a NE.
- Meanwhile, it is possible that the other players play the same action.

  If this same action is the other part of the NE $\Rightarrow$ in an absorbing state
- If this same action is not part of the NE, then this is not a best reply, and by randomisation it is possible that the next state is a pure NE.

Learning pure equilibria with recall one (entry #2)

Theorem 2. There exist uncoupled, 1-recall response rules that guarantee a.s. convergence of play to pure NE of the stage game in every two-player generic game where such equilibria exist.

Proof. Let players remember the last action profile played. Response rule: persist at best replies, randomise else. Gives a Markov proces.

- Each pure NE equilibrium is an absorbing state.
- Otherwise, ≥ 1 player randomises, possibly into his part of a NE.
- Meanwhile, it is possible that the other players play the same action.

  If this same action is the other part of the NE $\Rightarrow$ in an absorbing state
- If this same action is not part of the NE, then this is not a best reply, and by randomisation it is possible that the next state is a pure NE.

$$(a_1, a_2) \rightsquigarrow_{\exists} (a_1, a_2^*) \rightsquigarrow_{\exists \text{ 1 randomizes if the previous was not a NE}} (a_1^*, a_2^*).$$

Learning pure equilibria with recall two (entry #3)

Learning pure equilibria with recall two (entry #3)

Theorem 3. There exist uncoupled 2-recall response rules that guarantee almost sure convergence of play to pure NE of the stage game in every game where such equilibria exist.

Learning pure equilibria with recall two (entry #3)

Theorem 3. There exist uncoupled 2-recall response rules that guarantee almost sure convergence of play to pure NE of the stage game in every game where such equilibria exist.

Proof.

Learning pure equilibria with recall two (entry #3)

Theorem 3. There exist uncoupled 2-recall response rules that guarantee almost sure convergence of play to pure NE of the stage game in every game where such equilibria exist.

Proof. Let players maintain the last two action profiles: (a_{-2}, a_{-1}).

Learning pure equilibria with recall two (entry #3)

Theorem 3. There exist uncoupled 2-recall response rules that guarantee almost sure convergence of play to pure NE of the stage game in every game where such equilibria exist.

Proof. Let players maintain the last two action profiles: (a_{-2}, a_{-1}). Response rule: persist at best replies when $a_{-2} = a_{-1}$, randomise else.

Learning pure equilibria with recall two (entry #3)

Theorem 3. There exist uncoupled 2-recall response rules that guarantee almost sure convergence of play to pure NE of the stage game in every game where such equilibria exist.

Proof. Let players maintain the last two action profiles: (a_{-2}, a_{-1}).
Response rule: persist at best replies when $a_{-2} = a_{-1}$, randomise else.
Four regions (= sets of states $\neq$ classes) in the Markov chain:

Learning pure equilibria with recall two (entry #3)

Theorem 3. There exist uncoupled 2-recall response rules that guarantee almost sure convergence of play to pure NE of the stage game in every game where such equilibria exist.

Proof. Let players maintain the last two action profiles: (a_{-2}, a_{-1}). Response rule: persist at best replies when $a_{-2} = a_{-1}$, randomise else. Four regions (= sets of states $\neq$ classes) in the Markov chain:

1. $S_1 =_{Def} \{ (a\ , a) \in A^2 \mid a \text{ is Nash} \}$.

Learning pure equilibria with recall two (entry #3)

Theorem 3. There exist uncoupled 2-recall response rules that guarantee almost sure convergence of play to pure NE of the stage game in every game where such equilibria exist.

Proof. Let players maintain the last two action profiles: (a_{-2}, a_{-1}).
Response rule: persist at best replies when $a_{-2} = a_{-1}$, randomise else.
Four regions (= sets of states $\neq$ classes) in the Markov chain:

1. $S_1 =_{Def} \{ (a\ , a) \in A^2 \mid a \text{ is Nash} \}$.

2. $S_2 =_{Def} \{ (a', a) \in A^2 \mid a \text{ is Nash} \}, a \neq a'$.

Learning pure equilibria with recall two (entry #3)

Theorem 3. There exist uncoupled 2-recall response rules that guarantee almost sure convergence of play to pure NE of the stage game in every game where such equilibria exist.

Proof. Let players maintain the last two action profiles: (a_{-2}, a_{-1}).
Response rule: persist at best replies when $a_{-2} = a_{-1}$, randomise else.
Four regions (= sets of states ≠ classes) in the Markov chain:

1. $S_1 =_{Def} \{ (a\ , a) \in A^2 \mid a \text{ is Nash} \}$.
2. $S_2 =_{Def} \{ (a', a) \in A^2 \mid a \text{ is Nash} \}$, $a \neq a'$.
3. $S_3 =_{Def} \{ (a', a) \in A^2 \mid a \text{ is not Nash} \}$, $a \neq a'$.

Learning pure equilibria with recall two (entry #3)

Theorem 3. There exist uncoupled 2-recall response rules that guarantee almost sure convergence of play to pure NE of the stage game in every game where such equilibria exist.

Proof. Let players maintain the last two action profiles: (a_{-2}, a_{-1}). Response rule: persist at best replies when $a_{-2} = a_{-1}$, randomise else. Four regions (= sets of states $\neq$ classes) in the Markov chain:

1. $S_1 =_{Def} \{ (a\ , a) \in A^2 \mid a \text{ is Nash} \}$.

2. $S_2 =_{Def} \{ (a', a) \in A^2 \mid a \text{ is Nash} \}$, $a \neq a'$.

3. $S_3 =_{Def} \{ (a', a) \in A^2 \mid a \text{ is not Nash} \}$, $a \neq a'$.

4. $S_4 =_{Def} \{ (a\ , a) \in A^2 \mid a \text{ is not Nash} \}$.

Learning pure equilibria with recall two (entry #3)

Theorem 3. There exist uncoupled 2-recall response rules that guarantee almost sure convergence of play to pure NE of the stage game in every game where such equilibria exist.

Proof. Let players maintain the last two action profiles: (a_{-2}, a_{-1}).
Response rule: persist at best replies when $a_{-2} = a_{-1}$, randomise else.
Four regions (= sets of states $\neq$ classes) in the Markov chain:

1. $S_1 =_{Def} \{ (a\ , a) \in A^2 \mid a \text{ is Nash} \}$.
2. $S_2 =_{Def} \{ (a', a) \in A^2 \mid a \text{ is Nash} \}$, $a \neq a'$.
3. $S_3 =_{Def} \{ (a', a) \in A^2 \mid a \text{ is not Nash} \}$, $a \neq a'$.
4. $S_4 =_{Def} \{ (a\ , a) \in A^2 \mid a \text{ is not Nash} \}$.

Claim: all states in S_1 are absorbing, all other states are transient:

Learning pure equilibria with recall two (entry #3)

Theorem 3. There exist uncoupled 2-recall response rules that guarantee almost sure convergence of play to pure NE of the stage game in every game where such equilibria exist.

Proof. Let players maintain the last two action profiles: (a_{-2}, a_{-1}).
Response rule: persist at best replies when $a_{-2} = a_{-1}$, randomise else.
Four regions (= sets of states $\neq$ classes) in the Markov chain:

1. $S_1 =_{Def} \{ (a\ , a) \in A^2 \mid a \text{ is Nash} \}$.
2. $S_2 =_{Def} \{ (a', a) \in A^2 \mid a \text{ is Nash} \}$, $a \neq a'$.
3. $S_3 =_{Def} \{ (a', a) \in A^2 \mid a \text{ is not Nash} \}$, $a \neq a'$.
4. $S_4 =_{Def} \{ (a\ , a) \in A^2 \mid a \text{ is not Nash} \}$.

Claim: all states in S_1 are absorbing, all other states are transient:
$S_2 \rightsquigarrow_{\exists} S_1$

Learning pure equilibria with recall two (entry #3)

Theorem 3. There exist uncoupled 2-recall response rules that guarantee almost sure convergence of play to pure NE of the stage game in every game where such equilibria exist.

Proof. Let players maintain the last two action profiles: (a_{-2}, a_{-1}).
Response rule: persist at best replies when $a_{-2} = a_{-1}$, randomise else.
Four regions (= sets of states $\neq$ classes) in the Markov chain:

1. $S_1 =_{Def} \{ (a\ , a) \in A^2 \mid a \text{ is Nash} \}$.
2. $S_2 =_{Def} \{ (a', a) \in A^2 \mid a \text{ is Nash} \}$, $a \neq a'$.
3. $S_3 =_{Def} \{ (a', a) \in A^2 \mid a \text{ is not Nash} \}$, $a \neq a'$.
4. $S_4 =_{Def} \{ (a\ , a) \in A^2 \mid a \text{ is not Nash} \}$.

Claim: all states in S_1 are absorbing, all other states are transient:
$S_2 \rightsquigarrow_{\exists} S_1$, $S_3 \rightsquigarrow_{\exists} S_2$

Learning pure equilibria with recall two (entry #3)

Theorem 3. There exist uncoupled 2-recall response rules that guarantee almost sure convergence of play to pure NE of the stage game in every game where such equilibria exist.

Proof. Let players maintain the last two action profiles: (a_{-2}, a_{-1}). Response rule: persist at best replies when $a_{-2} = a_{-1}$, randomise else. Four regions (= sets of states $\neq$ classes) in the Markov chain:

1. $S_1 =_{Def} \{ (a\ , a) \in A^2 \mid a \text{ is Nash} \}$.
2. $S_2 =_{Def} \{ (a', a) \in A^2 \mid a \text{ is Nash} \}, a \neq a'$.
3. $S_3 =_{Def} \{ (a', a) \in A^2 \mid a \text{ is not Nash} \}, a \neq a'$.
4. $S_4 =_{Def} \{ (a\ , a) \in A^2 \mid a \text{ is not Nash} \}$.

Claim: all states in S_1 are absorbing, all other states are transient: $S_2 \rightsquigarrow_{\exists} S_1$, $S_3 \rightsquigarrow_{\exists} S_2$, $S_4 \rightsquigarrow_{\exists} S_2 \cup S_3$.

Learning pure equilibria with recall two (entry #3)

Theorem 3. There exist uncoupled 2-recall response rules that guarantee almost sure convergence of play to pure NE of the stage game in every game where such equilibria exist.

Proof. Let players maintain the last two action profiles: (a_{-2}, a_{-1}).
Response rule: persist at best replies when $a_{-2} = a_{-1}$, randomise else.
Four regions (= sets of states $\neq$ classes) in the Markov chain:

1. $S_1 =_{Def} \{ (a\ , a) \in A^2 \mid a \text{ is Nash} \}$.
2. $S_2 =_{Def} \{ (a', a) \in A^2 \mid a \text{ is Nash} \}$, $a \neq a'$.
3. $S_3 =_{Def} \{ (a', a) \in A^2 \mid a \text{ is not Nash} \}$, $a \neq a'$.
4. $S_4 =_{Def} \{ (a\ , a) \in A^2 \mid a \text{ is not Nash} \}$.

Claim: all states in S_1 are absorbing, all other states are transient:
$S_2 \rightsquigarrow_{\exists} S_1$, $S_3 \rightsquigarrow_{\exists} S_2$, $S_4 \rightsquigarrow_{\exists} S_2 \cup S_3$. $\square$

The possibility of learning mixed equilibria

Definitions

Definitions

- The profile frequency: for each profile $a \in A$:

$$\Phi_t[a] =_{Def} \frac{|\{s \leq t \mid a(s) = a\}|}{t}.$$

Definitions

- The profile frequency: for each profile $a \in A$:

$$\Phi_t[a] =_{Def} \frac{|\{s \leq t \mid a(s) = a\}|}{t}.$$

- The empirical *joint* distribution of play or the empirical frequencies of play:

$$\{\, \Phi_t[a] \mid a \in A \,\}.$$

Definitions

- The profile frequency: for each profile $a \in A$:

$$\Phi_t[a] =_{Def} \frac{|\{s \leq t \mid a(s) = a\}|}{t}.$$

- The empirical *joint* distribution of play or the empirical frequencies of play:

$$\{\, \Phi_t[a] \mid a \in A \,\}.$$

- The action frequency: for each profile a, for each player i:

$$\Phi_t[a^i] =_{Def} \frac{|\{s \leq t \mid a^i(s) = a^i\}|}{t}.$$

Definitions

- The profile frequency: for each profile $a \in A$:

$$\Phi_t[a] =_{Def} \frac{|\{s \leq t \mid a(s) = a\}|}{t}.$$

- The empirical *joint* distribution of play or the empirical frequencies of play:

$$\{\, \Phi_t[a] \mid a \in A \,\}.$$

- The action frequency: for each profile a, for each player i:

$$\Phi_t[a^i] =_{Def} \frac{|\{s \leq t \mid a^i(s) = a^i\}|}{t}.$$

- The empirical *marginal* distribution of play of Player i: $\{\, \Phi_t[a^i] \mid a^i \in A^i \,\}$.

Definitions

- The profile frequency: for each profile $a \in A$:

$$\Phi_t[a] =_{Def} \frac{|\{s \leq t \mid a(s) = a\}|}{t}.$$

- The empirical *joint* distribution of play or the empirical frequencies of play:

$$\{ \Phi_t[a] \mid a \in A \}.$$

- The action frequency: for each profile a, for each player i:

$$\Phi_t[a^i] =_{Def} \frac{|\{s \leq t \mid a^i(s) = a^i\}|}{t}.$$

- The empirical *marginal* distribution of play of Player i: $\{ \Phi_t[a^i] \mid a^i \in A^i \}$.

Check: $\Phi_t[a] \in [0,1]$, $\Phi_t[a^i] \in [0,1]$, joint $\in \Delta(A)$, marginal $\in \Delta(A^i)$.

Learning ϵ-Nash with bounded recall (entry #4)

Learning ϵ-Nash with bounded recall (entry #4)

Theorem 4. For every game with bounded payoffs and for every $\epsilon > 0$ there exists an integer R and a family of uncoupled R-recall response rules, such that the empirical marginal distributions of play converge to some ϵ-equilibrium a.s.

Learning ϵ-Nash with bounded recall (entry #4)

Theorem 4. For every game with bounded payoffs and for every $\epsilon > 0$ there exists an integer R and a family of uncoupled R-recall response rules, such that the empirical marginal distributions of play converge to some ϵ-equilibrium a.s.

I.e., for some ϵ-equilibrium x of the stage game, for every player i and every action $a^i \in A^i$,

$$\lim_{t \to \infty} \Phi_t[a^i] = x^i \text{ a.s.}$$

Learning ϵ-Nash with bounded recall (entry #4)

Theorem 4. For every game with bounded payoffs and for every $\epsilon > 0$ there exists an integer R and a family of uncoupled R-recall response rules, such that the empirical marginal distributions of play converge to some ϵ-equilibrium a.s.

I.e., for some ϵ-equilibrium x of the stage game, for every player i and every action $a^i \in A^i$,

$$\lim_{t \to \infty} \Phi_t[a^i] = x^i \text{ a.s.}$$

Proof.

Learning ϵ-Nash with bounded recall (entry #4)

Theorem 4. For every game with bounded payoffs and for every $\epsilon > 0$ there exists an integer R and a family of uncoupled R-recall response rules, such that the empirical marginal distributions of play converge to some ϵ-equilibrium a.s.

I.e., for some ϵ-equilibrium x of the stage game, for every player i and every action $a^i \in A^i$,

$$\lim_{t\to\infty} \Phi_t[a^i] = x^i \text{ a.s.}$$

Proof. Let $\epsilon > 0$ be given.

Learning ϵ-Nash with bounded recall (entry #4)

Theorem 4. For every game with bounded payoffs and for every $\epsilon > 0$ there exists an integer R and a family of uncoupled R-recall response rules, such that the empirical marginal distributions of play converge to some ϵ-equilibrium a.s.

I.e., for some ϵ-equilibrium x of the stage game, for every player i and every action $a^i \in A^i$,

$$\lim_{t\to\infty} \Phi_t[a^i] = x^i \text{ a.s.}$$

Proof. Let $\epsilon > 0$ be given. Since u^i is linear hence continuous, there exists an integer K such that

$$\|x^i - y^i\| \leq \frac{1}{K} \text{ for all } i \quad \Rightarrow \quad |u^i(x) - u^i(y)| \leq \frac{\epsilon}{2} \text{ for all } i \tag{1}$$

for all mixed strategies x^i, y^i of player i.[1]

[1] The authors use the max-norm: $\|x^i - y^i\| =_{Def} \max_k\{|x^i_k - y^i_k|\}$.

Proof of theorem

Proof of theorem

- Using (1), let y be a ϵ-NE such that all probabilities are multiples of $1/K$.

Proof of theorem

- Using (1), let y be a ϵ-NE such that all probabilities are multiples of $1/K$.

  Now every such $y = (y^1, \dots, y^N)$ can be realised by a sequence of action profiles $\bar{a}_y \in A^K$.

Proof of theorem

- Using (1), let y be a ϵ-NE such that all probabilities are multiples of $1/K$.

  Now every such $y = (y^1, \dots, y^N)$ can be realised by a sequence of action profiles $\bar{a}_y \in A^K$.

- Define a state to be a history of play of length $R = 2K$.

Proof of theorem

- Using (1), let y be a ϵ-NE such that all probabilities are multiples of $1/K$.

  Now every such $y = (y^1, \dots, y^N)$ can be realised by a sequence of action profiles $\bar{a}_y \in A^K$.

- Define a state to be a history of play of length $R = 2K$. Call a state K-periodic if $a_{K+k} = a_k$ for $1 \leq k \leq K$.

Proof of theorem

- Using (1), let y be a ϵ-NE such that all probabilities are multiples of $1/K$.

  Now every such $y = (y^1, \dots, y^N)$ can be realised by a sequence of action profiles $\bar{a}_y \in A^K$.

- Define a state to be a history of play of length $R = 2K$. Call a state K-periodic if $a_{K+k} = a_k$ for $1 \leq k \leq K$.

- $z^i \in \Delta(A^i)$ is the empirical frequency of i in the last K rounds.

Proof of theorem

- Using (1), let y be a ϵ-NE such that all probabilities are multiples of $1/K$.

  Now every such $y = (y^1, \dots, y^N)$ can be realised by a sequence of action profiles $\bar{a}_y \in A^K$.

- Define a state to be a history of play of length $R = 2K$. Call a state K-periodic if $a_{K+k} = a_k$ for $1 \leq k \leq K$.

- $z^i \in \Delta(A^i)$ is the empirical frequency of i in the last K rounds. Response rule: player i persists in every K-periodic state where z^i is an ϵ-best reply to z^{-i}, else he randomises.

Proof of theorem

- Using (1), let y be a ϵ-NE such that all probabilities are multiples of $1/K$.

  Now every such $y = (y^1, \dots, y^N)$ can be realised by a sequence of action profiles $\bar{a}_y \in A^K$.

- Define a state to be a history of play of length $R = 2K$. Call a state K-periodic if $a_{K+k} = a_k$ for $1 \le k \le K$.

- $z^i \in \Delta(A^i)$ is the empirical frequency of i in the last K rounds. Response rule: player i persists in every K-periodic state where z^i is an ϵ-best reply to z^{-i}, else he randomises.

- We now have a Markov chain with four regions:

Proof of theorem

- Using (1), let y be a ϵ-NE such that all probabilities are multiples of $1/K$.

  Now every such $y = (y^1, \ldots, y^N)$ can be realised by a sequence of action profiles $\bar{a}_y \in A^K$.

- Define a state to be a history of play of length $R = 2K$. Call a state K-periodic if $a_{K+k} = a_k$ for $1 \leq k \leq K$.

- $z^i \in \Delta(A^i)$ is the empirical frequency of i in the last K rounds. Response rule: player i persists in every K-periodic state where z^i is an ϵ-best reply to z^{-i}, else he randomises.

- We now have a Markov chain with four regions:

  1. $S_1 =_{Def} \{\, s$ is K-periodic and z is an ϵ-NE $\}$.

Proof of theorem

- Using (1), let y be a ϵ-NE such that all probabilities are multiples of $1/K$.

  Now every such $y = (y^1, \dots, y^N)$ can be realised by a sequence of action profiles $\bar{a}_y \in A^K$.

- Define a state to be a history of play of length $R = 2K$. Call a state K-periodic if $a_{K+k} = a_k$ for $1 \leq k \leq K$.

- $z^i \in \Delta(A^i)$ is the empirical frequency of i in the last K rounds. Response rule: player i persists in every K-periodic state where z^i is an ϵ-best reply to z^{-i}, else he randomises.

- We now have a Markov chain with four regions:

  1. $S_1 =_{Def} \{\, s$ is K-periodic and z is an ϵ-NE $\}$.
  2. $S_2 =_{Def} \{\, s$ is not K-periodic and z is an ϵ-NE $\}$.

Proof of theorem

- Using (1), let y be a ϵ-NE such that all probabilities are multiples of $1/K$.

  Now every such $y = (y^1, \dots, y^N)$ can be realised by a sequence of action profiles $\bar{a}_y \in A^K$.

- Define a state to be a history of play of length $R = 2K$. Call a state K-periodic if $a_{K+k} = a_k$ for $1 \leq k \leq K$.

- $z^i \in \Delta(A^i)$ is the empirical frequency of i in the last K rounds. Response rule: player i persists in every K-periodic state where z^i is an ϵ-best reply to z^{-i}, else he randomises.

- We now have a Markov chain with four regions:

  1. $S_1 =_{Def} \{ s$ is K-periodic and z is an ϵ-NE $\}$.
  2. $S_2 =_{Def} \{ s$ is not K-periodic and z is an ϵ-NE $\}$.
  3. $S_3 =_{Def} \{ s$ is not K-periodic, z is not an ϵ-NE $\}$.

Proof of theorem

- Using (1), let y be a ϵ-NE such that all probabilities are multiples of $1/K$.

  Now every such $y = (y^1, \dots, y^N)$ can be realised by a sequence of action profiles $\bar{a}_y \in A^K$.
- Define a state to be a history of play of length $R = 2K$. Call a state K-periodic if $a_{K+k} = a_k$ for $1 \leq k \leq K$.
- $z^i \in \Delta(A^i)$ is the empirical frequency of i in the last K rounds. Response rule: player i persists in every K-periodic state where z^i is an ϵ-best reply to z^{-i}, else he randomises.
- We now have a Markov chain with four regions:
  1. $S_1 =_{Def} \{\, s$ is K-periodic and z is an ϵ-NE $\}$.
  2. $S_2 =_{Def} \{\, s$ is not K-periodic and z is an ϵ-NE $\}$.
  3. $S_3 =_{Def} \{\, s$ is not K-periodic, z is not an ϵ-NE $\}$.
  4. $S_4 =_{Def} \{\, s$ is K-periodic and z is not an ϵ-NE $\}$.

Proof of theorem

- Using (1), let y be a ϵ-NE such that all probabilities are multiples of $1/K$.

  Now every such $y = (y^1, \dots, y^N)$ can be realised by a sequence of action profiles $\bar{a}_y \in A^K$.

- Define a state to be a history of play of length $R = 2K$. Call a state K-periodic if $a_{K+k} = a_k$ for $1 \leq k \leq K$.

- $z^i \in \Delta(A^i)$ is the empirical frequency of i in the last K rounds. Response rule: player i persists in every K-periodic state where z^i is an ϵ-best reply to z^{-i}, else he randomises.

- We now have a Markov chain with four regions:

  1. $S_1 =_{Def} \{ s$ is K-periodic and z is an ϵ-NE $\}$.
  2. $S_2 =_{Def} \{ s$ is not K-periodic and z is an ϵ-NE $\}$.
  3. $S_3 =_{Def} \{ s$ is not K-periodic, z is not an ϵ-NE $\}$.
  4. $S_4 =_{Def} \{ s$ is K-periodic and z is not an ϵ-NE $\}$.

  The proof now proceeds as the proof of Theorem 3.

Proof of theorem

- Using (1), let y be a ϵ-NE such that all probabilities are multiples of $1/K$.

  Now every such $y = (y^1, \ldots, y^N)$ can be realised by a sequence of action profiles $\bar{a}_y \in A^K$.

- Define a state to be a history of play of length $R = 2K$. Call a state K-periodic if $a_{K+k} = a_k$ for $1 \leq k \leq K$.

- $z^i \in \Delta(A^i)$ is the empirical frequency of i in the last K rounds. Response rule: player i persists in every K-periodic state where z^i is an ϵ-best reply to z^{-i}, else he randomises.

- We now have a Markov chain with four regions:

  1. $S_1 =_{Def} \{ s$ is K-periodic and z is an ϵ-NE $\}$.
  2. $S_2 =_{Def} \{ s$ is not K-periodic and z is an ϵ-NE $\}$.
  3. $S_3 =_{Def} \{ s$ is not K-periodic, z is not an ϵ-NE $\}$.
  4. $S_4 =_{Def} \{ s$ is K-periodic and z is not an ϵ-NE $\}$.

  The proof now proceeds as the proof of Theorem 3. (Region 3. needs the $\bar{a}_y$.)

Uncoupled learning of ϵ-Nash (entries #5 and #6)

Uncoupled learning of ϵ-Nash (entries #5 and #6)

It is even possible to prove the following:

Theorem 5. For every $\epsilon > 0$ there exists an R and an uncoupled, R-recall, family of response rules that guarantees, in every game, the a.s. convergence of the empirical joint distributions of play to Nash ϵ-equilibria.

(It's proof is involved, see the article of Hart and Mas-Colell.)

Uncoupled learning of ϵ-Nash (entries #5 and #6)

It is even possible to prove the following:

Theorem 5. For every $\epsilon > 0$ there exists an R and an uncoupled, R-recall, family of response rules that guarantees, in every game, the a.s. convergence of the empirical joint distributions of play to Nash ϵ-equilibria.

(It's proof is involved, see the article of Hart and Mas-Colell.)

What about behaviour probabilities (a.k.a. actual play)

$$\mathbf{Pr}[\, a_t^i = a^i \mid h_{t-1} \,] \quad \Big(= \; g_i(h_{t-1})(a^i) \Big) \quad ?$$

Uncoupled learning of ϵ-Nash (entries #5 and #6)

It is even possible to prove the following:

Theorem 5. For every $\epsilon > 0$ there exists an R and an uncoupled, R-recall, family of response rules that guarantees, in every game, the a.s. convergence of the empirical joint distributions of play to Nash ϵ-equilibria.

(It's proof is involved, see the article of Hart and Mas-Colell.)

What about behaviour probabilities (a.k.a. actual play)

$$\mathbf{Pr}[\, a^i_t = a^i \mid h_{t-1} \,] \quad \Big(= \; g_i(h_{t-1})(a^i) \Big) \quad ?$$

Theorem 6. For every small enough $\epsilon > 0$, there are no uncoupled, finite recall response rules that guarantee, in every game, the almost sure convergence of the behaviour probabilities to ϵ-equilibria of the stage game.

Impossibility of learning behaviour prob's (entry #6)

Impossibility of learning behaviour prob's (entry #6)

- Choose a stage game U with a unique equilibrium x which is fully mixed.

Impossibility of learning behaviour prob's (entry #6)

- Choose a stage game U with a unique equilibrium x which is fully mixed. (Matching p's.)

Impossibility of learning behaviour prob's (entry #6)

- Choose a stage game U with a unique equilibrium x which is fully mixed. (Matching p's.)
- Assume an action profile, a, such that $\{a_1\} = \mathrm{BR}(a_{-1})$.

Impossibility of learning behaviour prob's (entry #6)

- Choose a stage game U with a unique equilibrium x which is fully mixed. (Matching p's.)
- Assume an action profile, a, such that $\{a_1\} = \mathrm{BR}(a_{-1})$.
- Let U' be U where payoffs for Player 1 are the same as in U, and maximal for others at a_{-1}.

Impossibility of learning behaviour prob's (entry #6)

- Choose a stage game U with a unique equilibrium x which is fully mixed. (Matching p's.)
- Assume an action profile, a, such that $\{a_1\} = \mathrm{BR}(a_{-1})$.
- Let U' be U where payoffs for Player 1 are the same as in U, and maximal for others at a_{-1}. Then a is the unique NE of U'.

Impossibility of learning behaviour prob's (entry #6)

- Choose a stage game U with a unique equilibrium x which is fully mixed. (Matching p's.)
- Assume an action profile, a, such that $\{a_1\} = \mathrm{BR}(a_{-1})$.
- Let U' be U where payoffs for Player 1 are the same as in U, and maximal for others at a_{-1}. Then a is the unique NE of U'.
- Take $\epsilon > 0$ such that

Impossibility of learning behaviour prob's (entry #6)

- Choose a stage game U with a unique equilibrium x which is fully mixed. (Matching p's.)
- Assume an action profile, a, such that $\{a_1\} = \mathrm{BR}(a_{-1})$.
- Let U' be U where payoffs for Player 1 are the same as in U, and maximal for others at a_{-1}. Then a is the unique NE of U'.
- Take $\epsilon > 0$ such that
  — All ϵ-NE in U are fully mixed.

Impossibility of learning behaviour prob's (entry #6)

- Choose a stage game U with a unique equilibrium x which is fully mixed. (Matching p's.)

- Assume an action profile, a, such that $\{a_1\} = \text{BR}(a_{-1})$.

- Let U' be U where payoffs for Player 1 are the same as in U, and maximal for others at a_{-1}. Then a is the unique NE of U'.

- Take $\epsilon > 0$ such that
  — All ϵ-NE in U are fully mixed.
  — For all ϵ-NE x and y in U and U' there is a ρ such that $x(a_i) < \rho < y(a_i)$.

Impossibility of learning behaviour prob's (entry #6)

- Choose a stage game U with a unique equilibrium x which is fully mixed. (Matching p's.)
- Assume an action profile, a, such that $\{a_1\} = \mathrm{BR}(a_{-1})$.
- Let U' be U where payoffs for Player 1 are the same as in U, and maximal for others at a_{-1}. Then a is the unique NE of U'.
- Take $\epsilon > 0$ such that
  — All ϵ-NE in U are fully mixed.
  — For all ϵ-NE x and y in U and U' there is a ρ such that $x(a_i) < \rho < y(a_i)$.
- The behaviour probabilities have been assumed to converge to ϵ-Nash.

Impossibility of learning behaviour prob's (entry #6)

- Choose a stage game U with a unique equilibrium x which is fully mixed. (Matching p's.)
- Assume an action profile, a, such that $\{a_1\} = \mathrm{BR}(a_{-1})$.
- Let U' be U where payoffs for Player 1 are the same as in U, and maximal for others at a_{-1}. Then a is the unique NE of U'.
- Take $\epsilon > 0$ such that
  — All ϵ-NE in U are fully mixed.
  — For all ϵ-NE x and y in U and U' there is a ρ such that $x(a_i) < \rho < y(a_i)$.
- The behaviour probabilities have been assumed to converge to ϵ-Nash.
- x as well as y give positive probabilities to a in U and U' resp.

Impossibility of learning behaviour prob's (entry #6)

- Choose a stage game U with a unique equilibrium x which is fully mixed. (Matching p's.)
- Assume an action profile, a, such that $\{a_1\} = \mathrm{BR}(a_{-1})$.
- Let U' be U where payoffs for Player 1 are the same as in U, and maximal for others at a_{-1}. Then a is the unique NE of U'.
- Take $\epsilon > 0$ such that
  — All ϵ-NE in U are fully mixed.
  — For all ϵ-NE x and y in U and U' there is a ρ such that $x(a_i) < \rho < y(a_i)$.
- The behaviour probabilities have been assumed to converge to ϵ-Nash.
- x as well as y give positive probabilities to a in U and U' resp. So, there is a positive probability that a occurs R times in succession: $\bar{a} = \underbrace{(a, \dots, a)}_{R}$.

Impossibility of learning behaviour prob's (entry #6)

- Choose a stage game U with a unique equilibrium x which is fully mixed. (Matching p's.)

- Assume an action profile, a, such that $\{a_1\} = \mathrm{BR}(a_{-1})$.

- Let U' be U where payoffs for Player 1 are the same as in U, and maximal for others at a_{-1}. Then a is the unique NE of U'.

- Take $\epsilon > 0$ such that
  — All ϵ-NE in U are fully mixed.
  — For all ϵ-NE x and y in U and U' there is a ρ such that $x(a_i) < \rho < y(a_i)$.

- The behaviour probabilities have been assumed to converge to ϵ-Nash.

- x as well as y give positive probabilities to a in U and U' resp. So, there is a positive probability that a occurs R times in succession: $\bar{a} = \underbrace{(a, \dots, a)}_{R}$.

- x is ϵ-Nash in U, so $g_1(\bar{a})(a_1) < \rho$.

Impossibility of learning behaviour prob's (entry #6)

- Choose a stage game U with a unique equilibrium x which is fully mixed. (Matching p's.)
- Assume an action profile, a, such that $\{a_1\} = \mathrm{BR}(a_{-1})$.
- Let U' be U where payoffs for Player 1 are the same as in U, and maximal for others at a_{-1}. Then a is the unique NE of U'.
- Take $\epsilon > 0$ such that
  — All ϵ-NE in U are fully mixed.
  — For all ϵ-NE x and y in U and U' there is a ρ such that $x(a_i) < \rho < y(a_i)$.
- The behaviour probabilities have been assumed to converge to ϵ-Nash.
- x as well as y give positive probabilities to a in U and U' resp. So, there is a positive probability that a occurs R times in succession: $\bar{a} = \underbrace{(a, \dots, a)}_{R}$.
- x is ϵ-Nash in U, so $g_1(\bar{a})(a_1) < \rho$.
- y is ϵ-Nash in U', so $g_1(\bar{a})(a_1) > \rho$.

Contradiction. $\square$

Work of D. Foster and H. Peyton Young on completely uncoupled learning

Work of D. Foster and H. Peyton Young on completely uncoupled learning

Young, H. Peyton. "Learning by trial and error." *Games and economic behaviour* **65**(2) (2009), pp. 626-643.

Work of D. Foster and H. Peyton Young on completely uncoupled learning

Young, H. Peyton. "Learning by trial and error." *Games and economic behaviour* **65**(2) (2009), pp. 626-643.

Foster, Dean P., and H. Peyton Young. “Regret testing: Learning to play Nash equilibrium without knowing you have an opponent” in: *Theoretical Economics* **1**(3) (2006), pp. 341-367.

Learning ϵ-equilibria by regret testing (entry #16)

Learning ϵ-equilibria by regret testing (entry #16)

- Before exposing (2006), the authors go to great lengths to acknowledge the impossibility results of Hart and Mas-Colell (2003/6).

Learning ϵ-equilibria by regret testing (entry #16)

- Before exposing (2006), the authors go to great lengths to acknowledge the impossibility results of Hart and Mas-Colell (2003/6).
- Intuitive idea: play a fixed (possibly mixed) strategy and choose a new one if some individual action did significantly better.

Learning ϵ-equilibria by regret testing (entry #16)

- Before exposing (2006), the authors go to great lengths to acknowledge the impossibility results of Hart and Mas-Colell (2003/6).
- Intuitive idea: play a fixed (possibly mixed) strategy and choose a new one if some individual action did significantly better.
- Germano and Lugosi (2007) extended the result from 2 to $n \geq 2$ players.

Learning ϵ-equilibria by regret testing (entry #16)

- Before exposing (2006), the authors go to great lengths to acknowledge the impossibility results of Hart and Mas-Colell (2003/6).
- Intuitive idea: play a fixed (possibly mixed) strategy and choose a new one if some individual action did significantly better.
- Germano and Lugosi (2007) extended the result from 2 to $n \geq 2$ players.

Germano, F. and Lugosi, G.. "Global Nash convergence of Foster and Young's regret testing" in *Games and Economic Behaviour*, **60**(1), pp. 135-154.

Algorithm

IDEA

Algorithm

IDEA

- Let Row and Col play mixed strategies x and y, and let A be the payoff matrix.

Algorithm

IDEA

- Let Row and Col play mixed strategies x and y, and let A be the payoff matrix.
- The scalar $x^T Ay$ equals Row's expected payoff for x.

Algorithm

IDEA

- Let Row and Col play mixed strategies x and y, and let A be the payoff matrix.
- The scalar $x^T Ay$ equals Row's expected payoff for x.
- The vector Ay equals Row's expected payoffs per action.

Algorithm

IDEA

- Let Row and Col play mixed strategies x and y, and let A be the payoff matrix.
- The scalar $x^T Ay$ equals Row's expected payoff for x.
- The vector Ay equals Row's expected payoffs per action.
- If $x^T Ay < \max\{Ay\}$, then x wasn't optimal.

Algorithm

IDEA

- Let Row and Col play mixed strategies x and y, and let A be the payoff matrix.
- The scalar $x^T Ay$ equals Row's expected payoff for x.
- The vector Ay equals Row's expected payoffs per action.
- If $x^T Ay < \max\{Ay\}$, then x wasn't optimal.

ALGORITHM

Algorithm

IDEA

- Let Row and Col play mixed strategies x and y, and let A be the payoff matrix.
- The scalar $x^T Ay$ equals Row's expected payoff for x.
- The vector Ay equals Row's expected payoffs per action.
- If $x^T Ay < \max\{Ay\}$, then x wasn't optimal.

ALGORITHM

- Parameters:

Algorithm

IDEA

- Let Row and Col play mixed strategies x and y, and let A be the payoff matrix.
- The scalar $x^T Ay$ equals Row's expected payoff for x.
- The vector Ay equals Row's expected payoffs per action.
- If $x^T Ay < \max\{Ay\}$, then x wasn't optimal.

ALGORITHM

- Parameters:
  — mixed strategy x

Algorithm

IDEA

- Let Row and Col play mixed strategies x and y, and let A be the payoff matrix.
- The scalar $x^T Ay$ equals Row's expected payoff for x.
- The vector Ay equals Row's expected payoffs per action.
- If $x^T Ay < \max\{Ay\}$, then x wasn't optimal.

ALGORITHM

- Parameters:
  — mixed strategy x (the "hypothesis")

Algorithm

IDEA

- Let Row and Col play mixed strategies x and y, and let A be the payoff matrix.
- The scalar $x^T Ay$ equals Row's expected payoff for x.
- The vector Ay equals Row's expected payoffs per action.
- If $x^T Ay < \max\{Ay\}$, then x wasn't optimal.

ALGORITHM

- Parameters:
  — mixed strategy x (the "hypothesis")
  — length of sample period s

Algorithm

IDEA

- Let Row and Col play mixed strategies x and y, and let A be the payoff matrix.
- The scalar $x^T Ay$ equals Row's expected payoff for x.
- The vector Ay equals Row's expected payoffs per action.
- If $x^T Ay < \max\{Ay\}$, then x wasn't optimal.

ALGORITHM

- Parameters:
  — mixed strategy x (the "hypothesis")
  — length of sample period s
  — probability of exploration ϵ

Algorithm

IDEA

- Let Row and Col play mixed strategies x and y, and let A be the payoff matrix.
- The scalar $x^T Ay$ equals Row's expected payoff for x.
- The vector Ay equals Row's expected payoffs per action.
- If $x^T Ay < \max\{Ay\}$, then x wasn't optimal.

ALGORITHM

- Parameters:
  — mixed strategy x (the "hypothesis")
  — length of sample period s
  — probability of exploration ϵ
  — tolerance level τ.

Algorithm

IDEA

- Let Row and Col play mixed strategies x and y, and let A be the payoff matrix.
- The scalar $x^T Ay$ equals Row's expected payoff for x.
- The vector Ay equals Row's expected payoffs per action.
- If $x^T Ay < \max\{Ay\}$, then x wasn't optimal.

ALGORITHM

- Parameters:
  — mixed strategy x (the "hypothesis")
  — length of sample period s
  — probability of exploration ϵ
  — tolerance level τ.
- Initialise x randomly.

Algorithm

IDEA

- Let Row and Col play mixed strategies x and y, and let A be the payoff matrix.
- The scalar $x^T Ay$ equals Row's expected payoff for x.
- The vector Ay equals Row's expected payoffs per action.
- If $x^T Ay < \max\{Ay\}$, then x wasn't optimal.

ALGORITHM

- Parameters:
  — mixed strategy x (the "hypothesis")
  — length of sample period s
  — probability of exploration ϵ
  — tolerance level τ.
- Initialise x randomly.

Repeat:

Algorithm

IDEA

- Let Row and Col play mixed strategies x and y, and let A be the payoff matrix.
- The scalar $x^T Ay$ equals Row's expected payoff for x.
- The vector Ay equals Row's expected payoffs per action.
- If $x^T Ay < \max\{Ay\}$, then x wasn't optimal.

ALGORITHM

- Parameters:
  — mixed strategy x (the "hypothesis")
  — length of sample period s
  — probability of exploration ϵ
  — tolerance level τ.
- Initialise x randomly.

Repeat:

1. Play x for s rounds, experimenting ϵ of the time.

Algorithm

IDEA

- Let Row and Col play mixed strategies x and y, and let A be the payoff matrix.
- The scalar $x^T Ay$ equals Row's expected payoff for x.
- The vector Ay equals Row's expected payoffs per action.
- If $x^T Ay < \max\{Ay\}$, then x wasn't optimal.

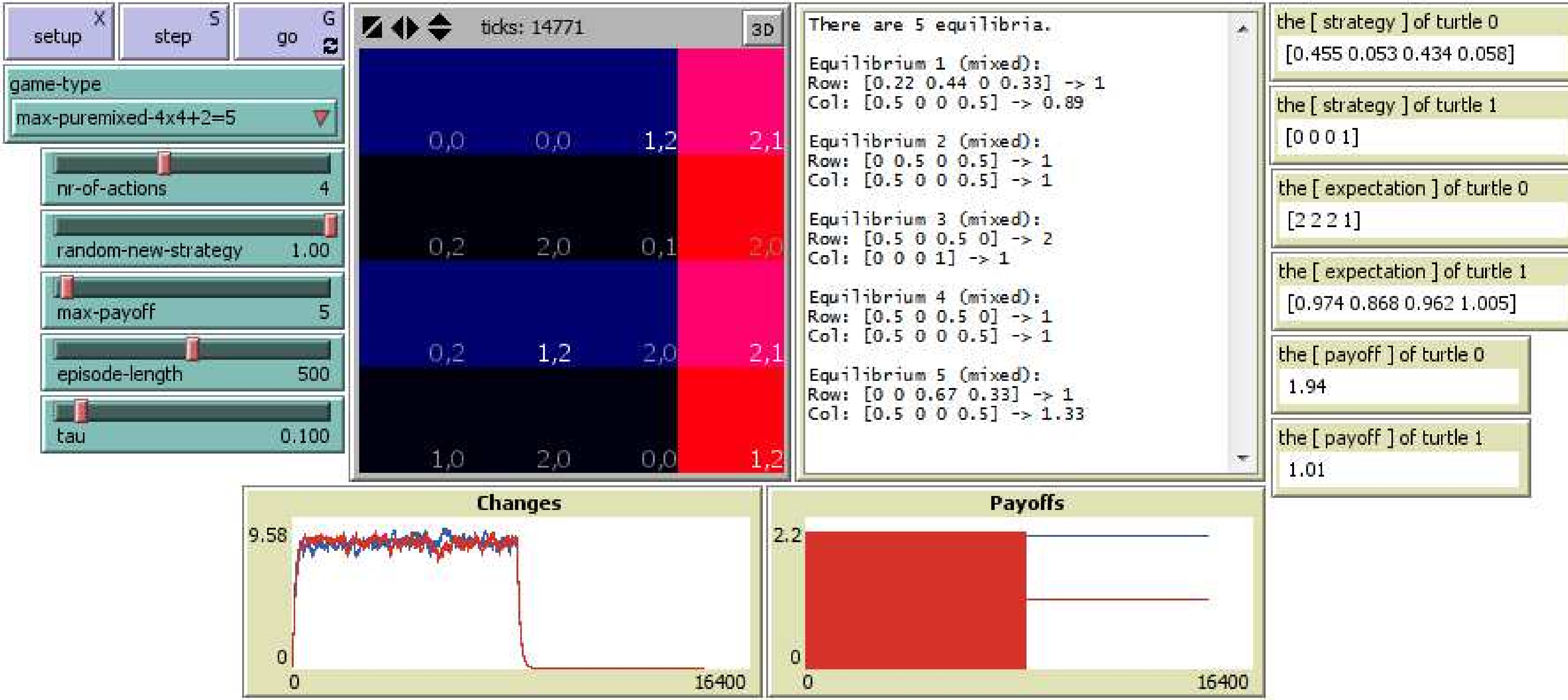
ALGORITHM

- Parameters:
  — mixed strategy x (the "hypothesis")
  — length of sample period s
  — probability of exploration ϵ
  — tolerance level τ.
- Initialise x randomly.

Repeat:

1. Play x for s rounds, experimenting ϵ of the time.
2. If, at the end of sampling, the revenue plus τ does not exceed the yield of every individual action when experimenting, adopt a new random x.

Demo: regret testing



Row player: blue; column player: white. To test the algorithm, a game with mixed equilibria only is chosen. Pareto sub-optimal action profiles are grey. In experiments, the strategy profiles are near the third equilibrium most of the time.

The following slides were not used.

Some algorithms to find Nash equilibria

Some algorithms to find Nash equilibria

1. Compute all the supports which could, in principle, be the support of a Nash equilibrium. Then search for a mixed equilibrium on that support by solving a system of simple equalities and inequalities.

Some algorithms to find Nash equilibria

1. Compute all the supports which could, in principle, be the support of a Nash equilibrium. Then search for a mixed equilibrium on that support by solving a system of simple equalities and inequalities.

2. Two-player games can be reduced to linear complementarity problems.

Some algorithms to find Nash equilibria

1. Compute all the supports which could, in principle, be the support of a Nash equilibrium. Then search for a mixed equilibrium on that support by solving a system of simple equalities and inequalities.
2. Two-player games can be reduced to linear complementarity problems.
3. In a two-player strategic game, the set of Nash equilibria can be expressed as extreme points of the union of convex sets.

Generate all the extreme points of those convex sets and test.

Some algorithms to find Nash equilibria

1. Compute all the supports which could, in principle, be the support of a Nash equilibrium. Then search for a mixed equilibrium on that support by solving a system of simple equalities and inequalities.
2. Two-player games can be reduced to linear complementarity problems.
3. In a two-player strategic game, the set of Nash equilibria can be expressed as extreme points of the union of convex sets.

Generate all the extreme points of those convex sets and test.

O. Mangaanknol (1964): "Equilibrium points in bi-matrix games, in: Journal of the Society for Industrial and Applied Mathematics, Vol. 12.

Some algorithms to find Nash equilibria

1. Compute all the supports which could, in principle, be the support of a Nash equilibrium. Then search for a mixed equilibrium on that support by solving a system of simple equalities and inequalities.

2. Two-player games can be reduced to linear complementarity problems.

3. In a two-player strategic game, the set of Nash equilibria can be expressed as extreme points of the union of convex sets. Generate all the extreme points of those convex sets and test.

   O. Mangaanknol (1964): "Equilibrium points in bi-matrix games, in: Journal of the Society for Industrial and Applied Mathematics, Vol. 12.

4. Use a global Newton method.

Some algorithms to find Nash equilibria

1. Compute all the supports which could, in principle, be the support of a Nash equilibrium. Then search for a mixed equilibrium on that support by solving a system of simple equalities and inequalities.
2. Two-player games can be reduced to linear complementarity problems.
3. In a two-player strategic game, the set of Nash equilibria can be expressed as extreme points of the union of convex sets.

   Generate all the extreme points of those convex sets and test.

   O. Mangaanknol (1964): "Equilibrium points in bi-matrix games, in: Journal of the Society for Industrial and Applied Mathematics, Vol. 12.
4. Use a global Newton method.
5. Compute equilibria via simplicial subdivision.

Some algorithms to find Nash equilibria

1. Compute all the supports which could, in principle, be the support of a Nash equilibrium. Then search for a mixed equilibrium on that support by solving a system of simple equalities and inequalities.

2. Two-player games can be reduced to linear complementarity problems.

3. In a two-player strategic game, the set of Nash equilibria can be expressed as extreme points of the union of convex sets.

   Generate all the extreme points of those convex sets and test.

   O. Mangaanknol (1964): "Equilibrium points in bi-matrix games, in: Journal of the Society for Industrial and Applied Mathematics, Vol. 12.

4. Use a global Newton method.

5. Compute equilibria via simplicial subdivision.

6. Form a differentiable function from mixed strategy profiles to $[0, \rightarrow)$ that is zero if and only if its argument is a Nash-equilibrium.

A differentiable function from strategy profiles to $\mathbb{R}$

A differentiable function from strategy profiles to $\mathbb{R}$

Define

$$f : S \to \mathbb{R}$$

A differentiable function from strategy profiles to $\mathbb{R}$

Define

$$f : S \to \mathbb{R} : s \mapsto$$

A differentiable function from strategy profiles to $\mathbb{R}$

Define

$$f : S \to \mathbb{R} : s \mapsto \sum_{i=1}^{n}$$

A differentiable function from strategy profiles to $\mathbb{R}$

Define

$$f : S \to \mathbb{R} : s \mapsto \sum_{i=1}^{n} \sum_{a_i \in A_i}$$

A differentiable function from strategy profiles to $\mathbb{R}$

Define

$$f : S \to \mathbb{R} : s \mapsto \sum_{i=1}^{n} \sum_{a_i \in A_i} \left([u_i(a_i, s_{-i}) - u_i(s)]_+ \right)^2$$

A differentiable function from strategy profiles to $\mathbb{R}$

Define

$$f : S \rightarrow \mathbb{R} : s \mapsto \sum_{i=1}^{n} \sum_{a_i \in A_i} \left([u_i(a_i, s_{-i}) - u_i(s)]_+ \right)^2$$

- f is non-negative.

A differentiable function from strategy profiles to $\mathbb{R}$

Define

$$f : S \to \mathbb{R} : s \mapsto \sum_{i=1}^{n} \sum_{a_i \in A_i} \left([u_i(a_i, s_{-i}) - u_i(s)]_+ \right)^2$$

- f is non-negative.
- $f = 0 \Leftrightarrow s$ is a (possibly mixed) Nash equilibrium.

A differentiable function from strategy profiles to $\mathbb{R}$

Define

$$f : S \to \mathbb{R} : s \mapsto \sum_{i=1}^{n} \sum_{a_i \in A_i} \left([u_i(a_i, s_{-i}) - u_i(s)]_+ \right)^2$$

- f is non-negative.
- $f = 0 \Leftrightarrow s$ is a (possibly mixed) Nash equilibrium.
- f is everywhere differentiable.

A differentiable function from strategy profiles to $\mathbb{R}$

Define

$$f : S \to \mathbb{R} : s \mapsto \sum_{i=1}^{n} \sum_{a_i \in A_i} \left([u_i(a_i, s_{-i}) - u_i(s)]_+ \right)^2$$

- f is non-negative.
- $f = 0 \Leftrightarrow s$ is a (possibly mixed) Nash equilibrium.
- f is everywhere differentiable. (Verification is tedious but doable.)

A differentiable function from strategy profiles to $\mathbb{R}$

Define

$$f : S \to \mathbb{R} : s \mapsto \sum_{i=1}^{n} \sum_{a_i \in A_i} \left([u_i(a_i, s_{-i}) - u_i(s)]_+ \right)^2$$

- f is non-negative.
- $f = 0 \Leftrightarrow s$ is a (possibly mixed) Nash equilibrium.
- f is everywhere differentiable. (Verification is tedious but doable.)
- Now minimise with gradient descent!

A differentiable function from strategy profiles to $\mathbb{R}$

Define

$$f : S \to \mathbb{R} : s \mapsto \sum_{i=1}^{n} \sum_{a_i \in A_i} \left([u_i(a_i, s_{-i}) - u_i(s)]_+ \right)^2$$

- f is non-negative.
- $f = 0 \Leftrightarrow s$ is a (possibly mixed) Nash equilibrium.
- f is everywhere differentiable. (Verification is tedious but doable.)
- Now minimise with gradient descent!
  – Classical Newton doesn't work because f isn't twice differentiable.
  Use the BFGS (Broyden-Fletcher-Goldfarb-Shanno) algorithm.

A differentiable function from strategy profiles to $\mathbb{R}$

Define

$$f : S \to \mathbb{R} : s \mapsto \sum_{i=1}^{n} \sum_{a_i \in A_i} \left([u_i(a_i, s_{-i}) - u_i(s)]_+ \right)^2$$

- f is non-negative.
- $f = 0 \Leftrightarrow s$ is a (possibly mixed) Nash equilibrium.
- f is everywhere differentiable. (Verification is tedious but doable.)
- Now minimise with gradient descent!
  – Classical Newton doesn't work because f isn't twice differentiable. Use the BFGS (Broyden-Fletcher-Goldfarb-Shanno) algorithm.
  – f is Lyapunov. So trajectories are bounded, convergent and stable.

A differentiable function from strategy profiles to $\mathbb{R}$

Define

$$f : S \to \mathbb{R} : s \mapsto \sum_{i=1}^{n} \sum_{a_i \in A_i} \left([u_i(a_i, s_{-i}) - u_i(s)]_+ \right)^2$$

- f is non-negative.
- $f = 0 \Leftrightarrow s$ is a (possibly mixed) Nash equilibrium.
- f is everywhere differentiable. (Verification is tedious but doable.)
- Now minimise with gradient descent!
  – Classical Newton doesn't work because f isn't twice differentiable. Use the BFGS (Broyden-Fletcher-Goldfarb-Shanno) algorithm.
  – f is Lyapunov. So trajectories are bounded, convergent and stable.

– McKelvey, R. D., A Liapunov function for Nash equilibria, mimeo, California Institute of Technology (1992).
– Judd, Kenneth L. Numerical methods in economics. MIT press, 1998. (Sec. 4.9: Computing Nash equilibrium.)

Learning pure equilibria by trial and error (entry #12)

Learning pure equilibria by trial and error (entry #12)

- An agent is said to learn by trial and error if he occasionally tries out new strategies

Learning pure equilibria by trial and error (entry #12)

- An agent is said to learn by trial and error if he occasionally tries out new strategies, rejecting choices below his aspiration level.

Learning pure equilibria by trial and error (entry #12)

- An agent is said to learn by trial and error if he occasionally tries out new strategies, rejecting choices below his aspiration level.
- Strategies can become erroneous due to a change of behaviour by someone else.

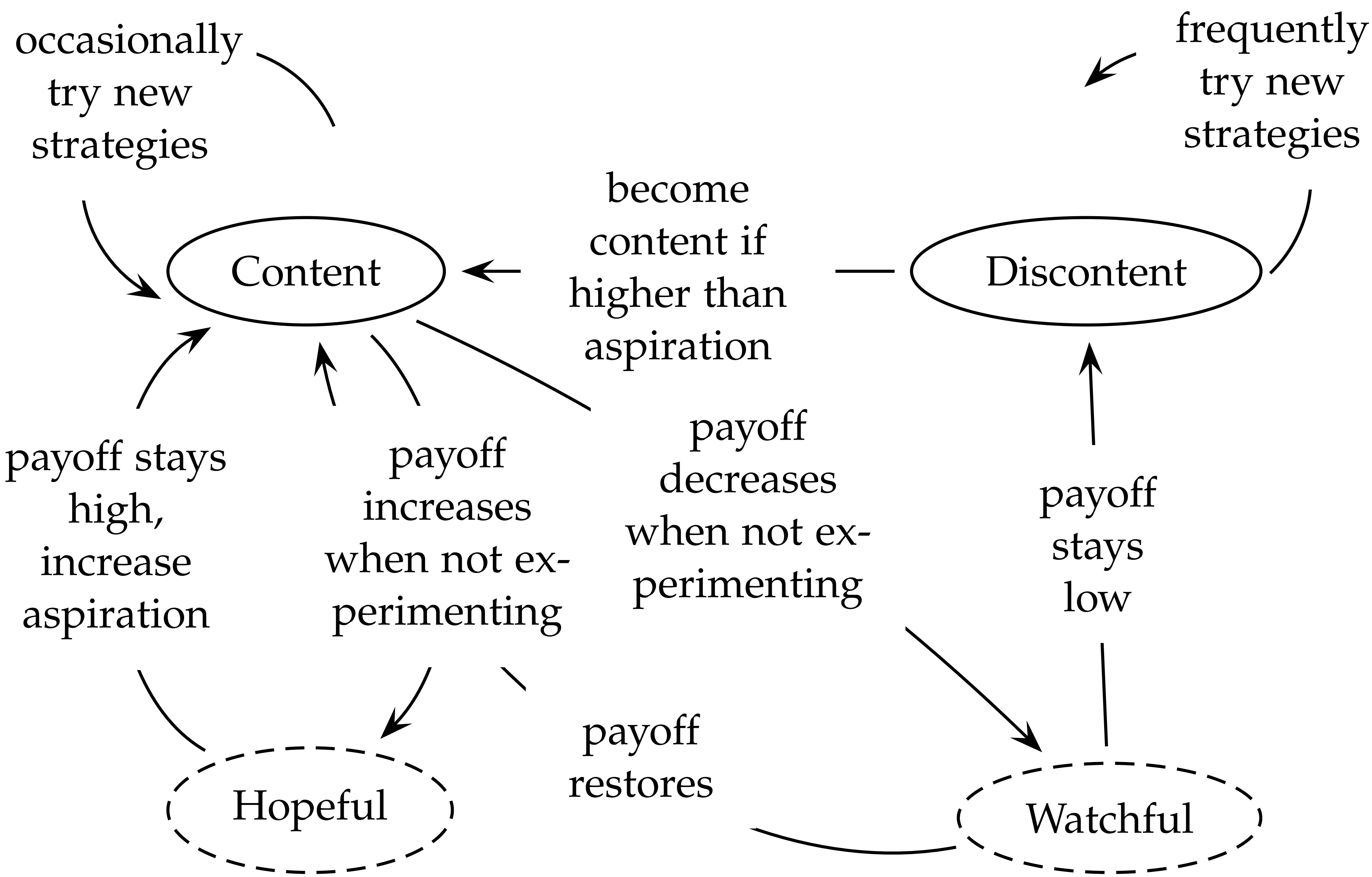
Learning pure equilibria by trial and error (entry #12)

- An agent is said to learn by trial and error if he occasionally tries out new strategies, rejecting choices below his aspiration level.
- Strategies can become erroneous due to a change of behaviour by someone else.
- Suppose a learning rule that conditionalises on whether a player experiences an error of the first or second type.

Learning pure equilibria by trial and error (entry #12)

- An agent is said to learn by trial and error if he occasionally tries out new strategies, rejecting choices below his aspiration level.
- Strategies can become erroneous due to a change of behaviour by someone else.
- Suppose a learning rule that conditionalises on whether a player experiences an error of the first or second type.
- This rule leads strategies into pure Nash equilibria most of the time in generic games that have pure Nash equilibria.

Idea: four states ("moods")



Why does this process lead to equilibrium?

Why does this process lead to equilibrium?

- **Intuition**:

Why does this process lead to equilibrium?

- **Intuition**: occasional search leads players toward progressively higher payoffs and higher aspiration levels until

Why does this process lead to equilibrium?

- **Intuition**: occasional search leads players toward progressively higher payoffs and higher aspiration levels until

  1. An equilibrium is reached.

Why does this process lead to equilibrium?

- **Intuition**: occasional search leads players toward progressively higher payoffs and higher aspiration levels until

  1. An equilibrium is reached.
  2. Someone's aspirations are disappointed before an equilibrium is reached.

Why does this process lead to equilibrium?

- **Intuition**: occasional search leads players toward progressively higher payoffs and higher aspiration levels until

  1. An equilibrium is reached.
  2. Someone's aspirations are disappointed before an equilibrium is reached.

- In the latter case the disappointed player starts searching at random

Why does this process lead to equilibrium?

- **Intuition**: occasional search leads players toward progressively higher payoffs and higher aspiration levels until

  1. An equilibrium is reached.
  2. Someone's aspirations are disappointed before an equilibrium is reached.

- In the latter case the disappointed player starts searching at random, which causes the other players to become disappointed with positive probability

Why does this process lead to equilibrium?

- **Intuition**: occasional search leads players toward progressively higher payoffs and higher aspiration levels until

  1. An equilibrium is reached.
  2. Someone's aspirations are disappointed before an equilibrium is reached.

- In the latter case the disappointed player starts searching at random, which causes the other players to become disappointed with positive probability, which leads to a full-scale random search by everyone.

Why does this process lead to equilibrium?

- **Intuition**: occasional search leads players toward progressively higher payoffs and higher aspiration levels until
  1. An equilibrium is reached.
  2. Someone's aspirations are disappointed before an equilibrium is reached.
- In the latter case the disappointed player starts searching at random, which causes the other players to become disappointed with positive probability, which leads to a full-scale random search by everyone.

  This phase concludes when everyone calms down and they start building a new monotone-payoff path.

Why does this process lead to equilibrium?

- **Intuition**: occasional search leads players toward progressively higher payoffs and higher aspiration levels until

  1. An equilibrium is reached.
  2. Someone's aspirations are disappointed before an equilibrium is reached.

- In the latter case the disappointed player starts searching at random, which causes the other players to become disappointed with positive probability, which leads to a full-scale random search by everyone.

  This phase concludes when everyone calms down and they start building a new monotone-payoff path.

- It can be shown that, *when the probability of calming down is sufficiently large relative to the probability of "thrashing around"*, the process is in a pure Nash equilibrium state much more often than in a disequilibrium state.