Recap	Iterated Removal	Computation	Rationalizability	Fun Game

Iterated Dominance, Rationalizability, Correlated Equilibrium

CPSC 532A Lecture 7

October 3, 2006

Iterated Dominance, Rationalizability, Correlated Equilibrium

CPSC 532A Lecture 7, Slide 1

Recap	Iterated Removal	Computation	Rationalizability	Fun Game
Lecture	Overview			

Recap

Iterated Removal

Computation

Rationalizability

Fun Game

Iterated Dominance, Rationalizability, Correlated Equilibrium

CPSC 532A Lecture 7, Slide 2

æ

イロン イヨン イヨン イヨン

 Recap
 Iterated Removal
 Computation
 Rationalizability
 Fun Game

 Max-Min Strategies
 Fun Game
 F

- Player i's maxmin strategy is a strategy that maximizes i's worst-case payoff, in the situation where all the other players (whom we denote -i) happen to play the strategies which cause the greatest harm to i.
- The maxmin value (or safety level) of the game for player i is that minimum amount of payoff guaranteed by a maxmin strategy.
- Why would i want to play a maxmin strategy?

 Recap
 Iterated Removal
 Computation
 Rationalizability
 Fun Game

 Max-Min Strategies
 Fun Game
 F

- Player i's maxmin strategy is a strategy that maximizes i's worst-case payoff, in the situation where all the other players (whom we denote -i) happen to play the strategies which cause the greatest harm to i.
- The maxmin value (or safety level) of the game for player i is that minimum amount of payoff guaranteed by a maxmin strategy.
- Why would i want to play a maxmin strategy?
 - a conservative agent maximizing worst-case payoff
 - a paranoid agent who believes everyone is out to get him

Definition

The maxmin strategy for player *i* is $\arg \max_{s_i} \min_{s_{-i}} u_i(s_1, s_2)$, and the maxmin value for player *i* is $\max_{s_i} \min_{s_{-i}} u_i(s_1, s_2)$.

3

・ロン ・回 と ・ ヨ と ・ ヨ と

Theorem (Minmax theorem (von Neumann, 1928))

In any finite, two-player, zero-sum game it is the case that:

- 1. The maxmin value for one player is equal to the minmax value for the other player. By convention, the maxmin value for player 1 is called the value of the game.
- 2. For both players, the set of maxmin strategies coincides with the set of minmax strategies.
- 3. Any maxmin strategy profile (or, equivalently, minmax strategy profile) is a Nash equilibrium. Furthermore, these are all the Nash equilibria. Consequently, all Nash equilibria have the same payoff vector (namely, those in which player 1 gets the value of the game).

3

▲ 理 ▶ | ▲ 理 ▶ …

$$\begin{array}{ll} \text{maximize} & \displaystyle \sum_{i} w_{i} x_{i} \\ \text{subject to} & \displaystyle \sum_{i} w_{i}^{c} x_{i} \geq b^{c} & \quad \forall c \in C \\ & \displaystyle x_{i} \geq 0 & \quad \forall x_{i} \in X \end{array}$$

- These problems can be solved in polynomial time using interior point methods.
 - Interestingly, the (worst-case exponential) simplex method is often faster in practice.

 Recap
 Iterated Removal
 Computation
 Rationalizability
 Fun Game

Computing equilibria of zero-sum games

$$\begin{array}{ll} \mbox{minimize} & U_1^* \\ \mbox{subject to} & \sum_{a_2 \in A_2} u_1(a_1,a_2) \cdot s_2^{a_2} \leq U_1^* & \forall a_1 \in A_1 \\ & \sum_{a_2 \in A_2} s_2^{a_2} = 1 \\ & s_2^{a_2} \geq 0 & \forall a_2 \in A_2 \end{array}$$

 This program can also be used on a modified version of a general-sum two-player game to compute maxmin and minmax strategies.

 Recap
 Iterated Removal
 Computation
 Rationalizability
 Fun Game

Computing equilibria of zero-sum games

$$\begin{array}{ll} \mbox{minimize} & U_1^*\\ \mbox{subject to} & \sum_{a_2 \in A_2} u_1(a_1,a_2) \cdot s_2^{a_2} \leq U_1^* \qquad \forall a_1 \in A_1\\ & \sum_{a_2 \in A_2} s_2^{a_2} = 1\\ & s_2^{a_2} \geq 0 \qquad \qquad \forall a_2 \in A_2 \end{array}$$

 This program can also be used on a modified version of a general-sum two-player game to compute maxmin and minmax strategies.

Recap Iterated Removal Computation Ratio

Computing equilibria of zero-sum games

$$\begin{array}{ll} \mbox{minimize} & U_1^*\\ \mbox{subject to} & \sum_{a_2 \in A_2} u_1(a_1,a_2) \cdot s_2^{a_2} \leq U_1^* \qquad \forall a_1 \in A_1\\ & \sum_{a_2 \in A_2} s_2^{a_2} = 1\\ & s_2^{a_2} \geq 0 \qquad \qquad \forall a_2 \in A_2 \end{array}$$

 This program can also be used on a modified version of a general-sum two-player game to compute maxmin and minmax strategies.

 Recap
 Iterated Removal
 Computation
 Rationalizability
 Fun Game

Computing equilibria of zero-sum games

$$\begin{array}{ll} \text{minimize} & U_1^*\\ \text{subject to} & \displaystyle\sum_{a_2 \in A_2} u_1(a_1, a_2) \cdot s_2^{a_2} \leq U_1^* \qquad \forall a_1 \in A_1\\ & \displaystyle\sum_{a_2 \in A_2} s_2^{a_2} = 1\\ & s_2^{a_2} \geq 0 \qquad \qquad \forall a_2 \in A_2 \end{array}$$

 This program can also be used on a modified version of a general-sum two-player game to compute maxmin and minmax strategies.

Recap	Iterated Removal	Computation	Rationalizability	Fun Game
Dominat	tion			

▶ Let s_i and s'_i be two strategies for player i, and let S_{-i} be is the set of all possible strategy profiles for the other players

Definition s_i strictly dominates s'_i if $\forall s_{-i} \in S_{-i}$, $u_i(s_i, s_{-i}) > u_i(s'_i, s_{-i})$

Definition

 s_i weakly dominates s_i' if $\forall s_{-i} \in S_{-i}, u_i(s_i, s_{-i}) \geq u_i(s_i', s_{-i})$ and $\exists s_{-i} \in S_{-i}, u_i(s_i, s_{-i}) > u_i(s_i', s_{-i})$

Definition

 s_i very weakly dominates s'_i if $\forall s_{-i} \in S_{-i}$, $u_i(s_i, s_{-i}) \ge u_i(s'_i, s_{-i})$

同 と く ヨ と く ヨ と … ヨ

Recap	Iterated Removal	Computation	Rationalizability	Fun Game
Lecture	Overview			

Recap

Iterated Removal

Computation

Rationalizability

Fun Game

Iterated Dominance, Rationalizability, Correlated Equilibrium

CPSC 532A Lecture 7, Slide 8

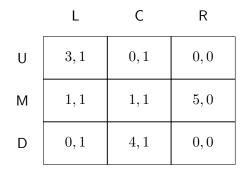
æ

イロン イヨン イヨン イヨン

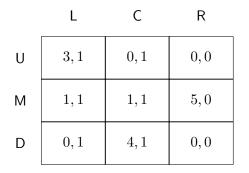
Recap	Iterated Removal	Computation	Rationalizability	Fun Game

Dominated strategies

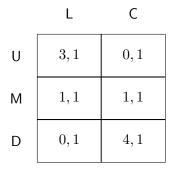
- No equilibrium can involve a strictly dominated strategy
 - Thus we can remove it, and end up with a strategically equivalent game
 - This might allow us to remove another strategy that wasn't dominated before
 - Running this process to termination is called iterated removal of dominated strategies.



< A

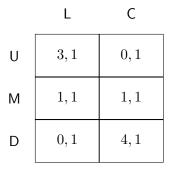


▶ *R* is dominated by *L*.

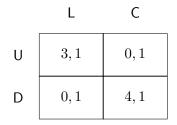


Iterated Dominance, Rationalizability, Correlated Equilibrium

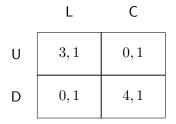
→ < ≧ → < ≧ → E → Q (CPSC 532A Lecture 7, Slide 11



M is dominated by the mixed strategy that selects U and D with equal probability.



Iterated Dominance, Rationalizability, Correlated Equilibrium



No other strategies are dominated.

→ < ≧ → < ≧ → E → Q (CPSC 532A Lecture 7, Slide 12

Recap	Iterated Removal	Computation	Rationalizability	Fun Game
Itorotod	Pomoval of Do	minated St.	votorios	

Iterated Removal of Dominated Strategies

- This process preserves Nash equilibria.
 - strict dominance: all equilibria preserved.
 - weak or very weak dominance: at least one equilibrium preserved.
- Thus, it can be used as a preprocessing step before computing an equilibrium
 - Some games are solvable using this technique.
 - Example: Traveler's Dilemma!
- What about the order of removal when there are multiple dominated strategies?
 - strict dominance: doesn't matter.
 - weak or very weak dominance: can affect which equilibria are preserved.

Recap	Iterated Removal	Computation	Rationalizability	Fun Game
Lecture	Overview			

Recap

Iterated Removal

Computation

Rationalizability

Fun Game

Iterated Dominance, Rationalizability, Correlated Equilibrium

CPSC 532A Lecture 7, Slide 14

æ

イロン イヨン イヨン イヨン

Computational Problems in Domination

- Identifying strategies dominated by a pure strategy
- Identifying strategies dominated by a mixed strategy
- Identifying strategies that survive iterated elimination
- Asking whether a strategy survives iterated elimination under all elimination orderings
- We'll assume that i's utility function is strictly positive everywhere (why is this OK?)

Computation Recap Iterated Removal Rationalizability Fun Game Is s_i strictly dominated by any pure strategy? for all pure strategies $a_i \in A_i$ for player *i* where $a_i \neq s_i$ do $dom \leftarrow true$ for all pure strategy profiles $a_{-i} \in A_{-i}$ for the players other than i do if $u_i(s_i, a_{-i}) > u_i(a_i, a_{-i})$ then $dom \leftarrow false$ break end if end for if dom = true then return trueend for return false

・ 戸 ト ・ ヨ ト ・ ヨ ト ・

Computation Recap Iterated Removal Rationalizability Fun Game Is s_i strictly dominated by any pure strategy? for all pure strategies $a_i \in A_i$ for player i where $a_i \neq s_i$ do $dom \leftarrow true$ for all pure strategy profiles $a_{-i} \in A_{-i}$ for the players other than i do if $u_i(s_i, a_{-i}) > u_i(a_i, a_{-i})$ then $dom \leftarrow false$ break end if end for if dom = true then return trueend for return false What is the complexity of this procedure?

- Why don't we have to check mixed strategies of the other players?
- What would we have to change to test for weak or very weak dominance?

Constraints for determining whether s_i is strictly dominated by any mixed strategy

$$\sum_{\substack{j \in A_i \\ p_j \ge 0}} p_j u_i(a_j, a_{-i}) > u_i(s_i, a_{-i}) \qquad \forall a_{-i} \in A_{-i}$$
$$p_j \ge 0 \qquad \forall j \in A_i$$
$$\sum_{j \in A_i} p_j = 1$$

Iterated Dominance, Rationalizability, Correlated Equilibrium

▶ < 콜 ▶ < 콜 ▶ 로 ∽ Q (CPSC 532A Lecture 7, Slide 17 Constraints for determining whether s_i is strictly dominated by any mixed strategy

$$\begin{split} \sum_{j \in A_i} p_j u_i(a_j, a_{-i}) > u_i(s_i, a_{-i}) & \forall a_{-i} \in A_{-i} \\ p_j \ge 0 & \forall j \in A_i \\ \sum_{j \in A_i} p_j = 1 \end{split}$$

What's wrong with this program?

Constraints for determining whether s_i is strictly dominated by any mixed strategy

$$\begin{split} \sum_{j \in A_i} p_j u_i(a_j, a_{-i}) > u_i(s_i, a_{-i}) & \forall a_{-i} \in A_{-i} \\ p_j \ge 0 & \forall j \in A_i \\ \sum_{j \in A_i} p_j = 1 \end{split}$$

- What's wrong with this program?
 - strict inequality in the first constraint means we don't have an LP

Recap Iterated Removal Computation Rationalizability Fun Game LP for determining whether s_i is strictly dominated by any mixed strategy

$$\begin{array}{ll} \mbox{minimize} & \displaystyle \sum_{j \in A_i} p_j \\ \mbox{subject to} & \displaystyle \sum_{j \in A_i} p_j u_i(a_j, a_{-i}) \geq u_i(s_i, a_{-i}) & \forall a_{-i} \in A_{-i} \end{array}$$

This is clearly an LP. Why is it a solution to our problem?

Iterated Dominance, Rationalizability, Correlated Equilibrium

Recap Iterated Removal Computation Rationalizability Fun Game LP for determining whether s_i is strictly dominated by any mixed strategy

$$\begin{array}{ll} \mbox{minimize} & \displaystyle \sum_{j \in A_i} p_j \\ \mbox{subject to} & \displaystyle \sum_{j \in A_i} p_j u_i(a_j, a_{-i}) \geq u_i(s_i, a_{-i}) & \forall a_{-i} \in A_{-i} \end{array}$$

• This is clearly an LP. Why is it a solution to our problem?

- ▶ if a solution exists with ∑_j p_j < 1 then we can add 1 ∑_j p_j to some p_k and we'll have a dominating mixed strategy (since utility was assumed to be positive everywhere)
- Our original program (weak inequality) works for very weak domination
- For weak domination we can use that program with a different objective function trick.

- This can be done by repeatedly solving our LPs: solving a polynomial number of LPs is still in *P*.
 - Checking whether every pure strategy of every player is dominated by any other mixed strategy requires us to solve at worst $\sum_{i \in N} |A_i|$ linear programs.
 - ▶ Each step removes one pure strategy for one player, so there can be at most $\sum_{i \in N} (|A_i| 1)$ steps.
 - Thus we need to solve $O((n \cdot a^*)^2)$ linear programs, where $a^* = \max_i |A_i|$.

・ 同 ト ・ ヨ ト ・ ヨ ト …

RecapIterated RemovalComputationRationalizabilityFun GameFurther questions about iterated elimination

- 1. (Strategy Elimination) Does there exist some elimination path under which the strategy s_i is eliminated?
- 2. (Reduction Identity) Given action subsets $A'_i \subseteq A_i$ for each player i, does there exist a maximally reduced game where each player i has the actions A'_i ?
- 3. **(Uniqueness)** Does every elimination path lead to the same reduced game?
- 4. (Reduction Size) Given constants k_i for each player *i*, does there exist a maximally reduced game where each player *i* has exactly k_i actions?

(4) (5) (4) (5) (4)

RecapIterated RemovalComputationRationalizabilityFun GameFurther questions about iterated elimination

- 1. (Strategy Elimination) Does there exist some elimination path under which the strategy s_i is eliminated?
- 2. (Reduction Identity) Given action subsets $A'_i \subseteq A_i$ for each player i, does there exist a maximally reduced game where each player i has the actions A'_i ?
- 3. **(Uniqueness)** Does every elimination path lead to the same reduced game?
- 4. (Reduction Size) Given constants k_i for each player *i*, does there exist a maximally reduced game where each player *i* has exactly k_i actions?
- For iterated strict dominance these problems are all in \mathcal{P} .
- ► For iterated weak or very weak dominance these problems are all *NP*-complete.

· < @ > < 문 > < 문 > · · 문

Recap	Iterated Removal	Computation	Rationalizability	Fun Game
Lecture	Overview			

Recap

Iterated Removal

Computation

Rationalizability

Fun Game

Iterated Dominance, Rationalizability, Correlated Equilibrium

CPSC 532A Lecture 7, Slide 21

æ

イロン イヨン イヨン イヨン



- assumes opponent is rational
- assumes opponent knows that you and the others are rational
- ► ...
- Examples
 - is *heads* rational in matching pennies?

< 注→ < 注→



- assumes opponent is rational
- assumes opponent knows that you and the others are rational
- <u>ا...</u>
- Examples
 - is *heads* rational in matching pennies?
 - is cooperate rational in prisoner's dilemma?

(3)



- assumes opponent is rational
- assumes opponent knows that you and the others are rational
- ...
- Examples
 - is *heads* rational in matching pennies?
 - is cooperate rational in prisoner's dilemma?
- Will there always exist a rationalizable strategy?

< ∃ >



- assumes opponent is rational
- assumes opponent knows that you and the others are rational
- <u>ا...</u>
- Examples
 - is *heads* rational in matching pennies?
 - is cooperate rational in prisoner's dilemma?
- Will there always exist a rationalizable strategy?
 - > Yes, equilibrium strategies are always rationalizable.

(4) (E) (A) (E) (A)



- assumes opponent is rational
- assumes opponent knows that you and the others are rational
- ▶ ...
- Examples
 - is *heads* rational in matching pennies?
 - is cooperate rational in prisoner's dilemma?
- Will there always exist a rationalizable strategy?
 - > Yes, equilibrium strategies are always rationalizable.
- ► Furthermore, in two-player games, rationalizable ⇔ survives iterated removal of strictly dominated strategies.

・ 戸 ト ・ ヨ ト ・ ヨ ト

Recap	Iterated Removal	Computation	Rationalizability	Fun Game
Lecture	Overview			

Recap

Iterated Removal

Computation

Rationalizability

Fun Game

Iterated Dominance, Rationalizability, Correlated Equilibrium

CPSC 532A Lecture 7, Slide 23

æ

イロン イヨン イヨン イヨン

Recap	Iterated Removal	Computation	Rationalizability	Fun Game
Fun game				

$$\begin{array}{c|ccccccc} L & H & S \\ L & 90,90 & 0,0 & 0,40 \\ B & 0,0 & 180,180 & 0,40 \end{array}$$

Iterated Dominance, Rationalizability, Correlated Equilibrium

CPSC 532A Lecture 7, Slide 24

◆□ → ◆□ → ◆目 → ◆目 → ◆□ →

Recap	Iterated Removal	Computation	Rationalizability	Fun Game
Fun game				

$$\begin{array}{c|cccccc} L & H & S \\ L & 90,90 & 0,0 & 400,40 \\ B & 0,0 & 180,180 & 0,40 \end{array}$$

Iterated Dominance, Rationalizability, Correlated Equilibrium

CPSC 532A Lecture 7, Slide 24

◆□ → ◆□ → ◆目 → ◆目 → ◆□ →

Recap	Iterated Removal	Computation	Rationalizability	Fun Game
Fun game				

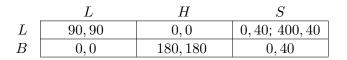
	L	H	S
L	90,90	0, 0	0,40;400,40
B	0,0	180, 180	0,40

What's the equilibrium?

Iterated Dominance, Rationalizability, Correlated Equilibrium

▶ < 콜 ▶ < 콜 ▶ 콜 → 오 (CPSC 532A Lecture 7, Slide 24

Recap	Iterated Removal	Computation	Rationalizability	Fun Game
Fun game				



- What's the equilibrium?
 - ► 50-50 L-H dominates S for column, so we have a standard coordination game.

Recap	Iterated Removal	Computation	Rationalizability	Fun Game
Fun game				

	L	H	S
L	90,90	0, 0	0,40;400,40
B	0,0	180, 180	0,40

- What's the equilibrium?
 - ► 50-50 L-H dominates S for column, so we have a standard coordination game.
- What happens when people play?

Recap	Iterated Removal	Computation	Rationalizability	Fun Game
Fun game				

	L	H	S
L	90,90	0, 0	0,40;400,40
B	0,0	180, 180	0,40

- What's the equilibrium?
 - ► 50-50 L-H dominates S for column, so we have a standard coordination game.

What happens when people play?

- with 0, 40, 96% row and 84% column choose the high payoff *H*, coordination occurs 80% of the time.
- with 400, 40, 64% row and 76% column chose H; coordination on H,H 32% of the time, coordination on L,L 16% of the time, uncoordinated over half the time