# Correlated Equilibria in Succinct Games

**BCTCS 2006 Swansea**

**06/04/2006**

**Matthias Mnich, London School of Economics**

# Representation of Games

- **strategic form** (normal form)

# Representation of Games

- strategic form (normal form)
- utility matrix for pure strategies

# Representation of Games

- strategic form (normal form)
- utility matrix for pure strategies
- example: chicken game

| | stop | go |
|---|---|---|
| stop | 4,4 | 1,5 |
| go | 5,1 | 0,0 |

## Correlated Equilibrium [AUMANN 1974, 1987]

- distribution over strategy profiles

## Correlated Equilibrium [Aumann 1974, 1987]

- distribution over strategy profiles
- trusted intermediary suggests strategy to each player

## Correlated Equilibrium [AUMANN 1974, 1987]

- distribution over strategy profiles
- trusted intermediary suggests strategy to each player
- no player has incentive to deviate

## Correlated Equilibrium [AUMANN 1974, 1987]

- distribution over strategy profiles
- trusted intermediary suggests strategy to each player
- no player has incentive to deviate
- under condition that all others also follow suggestion

## Correlated Equilibrium [AUMANN 1974, 1987]

- distribution over strategy profiles
- trusted intermediary suggests strategy to each player
- no player has incentive to deviate
- under condition that all others also follow suggestion
- chicken game:

| | |
|---|---|
| 0 | 1 |
| 0 | 0 |

| | |
|---|---|
| 0 | 0 |
| 1 | 0 |

| | |
|---|---|
| 1/4 | 1/4 |
| 1/4 | 1/4 |

| | |
|---|---|
| 0 | 1/2 |
| 1/2 | 0 |

| | |
|---|---|
| 1/3 | 1/3 |
| 1/3 | 0 |

## Succinct Games [Papadimitriou 2005]

- **strategy profile**: one strategy for each player

## Succinct Games [PAPADIMITRIOU 2005]

- strategy profile: one strategy for each player
- number of strategy profiles exponential in number of players

# Succinct Games [Papadimitriou 2005]

- **strategy profile**: one strategy for each player
- number of strategy profiles **exponential** in number of players
- need **succinct representation** of game

## Succinct Games [Papadimitriou 2005]

- strategy profile: one strategy for each player
- number of strategy profiles exponential in number of players
- need succinct representation of game
- chicken game: symmetric (all players with same strategies)

  ⇒ number of players for each strategy sufficient

# Existence of Correlated Equilibria (CE)

**Theorem:** Every finite game has a CE.

## Existence of Correlated Equilibria (CE)

**Theorem:** Every finite game has a CE.

**Proof:**

- **Nash equilibria are correlated. [NASH 1954]**

# Existence of Correlated Equilibria (CE)

**Theorem:** Every finite game has a CE.

**Proof:**

- **Nash equilibria are correlated. [NASH 1954]**
- **Min-Max Theorem of linear programming. (HART/SCHMEIDLER 1989)**

# Existence of Correlated Equilibria (CE)

**Theorem:** **Every finite game has a CE.**

**Proof:**

- **Nash equilibria are correlated. [NASH 1954]**
- **Min-Max Theorem of linear programming. (HART/SCHMEIDLER 1989)**
- **Stationary distributions of finite Markov chains. (NAU/MCCARDLE 1990) $\rightarrow$ STOC 2005: used for polynomial-time construction of CE.**

# Linear Programming Duality

$$
(P) \qquad \begin{aligned} \max \sum x_s & \\ Ux &\geq 0 \\ x &\geq 0. \end{aligned}
$$

# Linear Programming Duality

$$\max \sum x_s$$

$$(P) \qquad \begin{aligned} Ux &\geq 0 \\ x &\geq 0. \end{aligned}$$

$$(D) \qquad \begin{aligned} U^T y &\leq -1 \\ y &\geq 0 \end{aligned}$$

# Linear Programming Duality

$$\max \sum x_s$$

$$(P) \qquad \begin{aligned} Ux &\geq 0 \\ x &\geq 0. \end{aligned}$$

$$(D) \qquad \begin{aligned} U^T y &\leq -1 \\ y &\geq 0 \end{aligned}$$

- **(D) with polynomially many variables**

## Linear Programming Duality

$$
(P) \qquad \begin{aligned} \max \sum x_s \\ Ux &\geq 0 \\ x &\geq 0. \end{aligned}
$$

$$
(D) \qquad \begin{aligned} U^T y &\leq -1 \\ y &\geq 0 \end{aligned}
$$

- **(D) with polynomially many variables**
- **and exponentially many constraints**

## Ellipsoid algorithm on (D)

- **in each iterate $K$: generate candidate solution $y^K$ (infeasible)**

## Ellipsoid algorithm on (D)

- in each iterate $K$: generate candidate solution $y^K$ (infeasible)
- from this: compute distribution $x^K$ with product distributions $x_s^K$

## Ellipsoid algorithm on (D)

- in each iterate $K$: generate candidate solution $y^K$ (infeasible)
- from this: compute distribution $x^K$ with product distributions $x_s^K$
- terminates after polynomially many steps (although exponentially many constraints)

# Ellipsoid algorithm on (D)

- in each iterate $K$: generate candidate solution $y^K$ (infeasible)
- from this: compute distribution $x^K$ with product distributions $x_s^K$
- terminates after polynomially many steps (although exponentially many constraints)
- but precision issues: need exact $y^K$

- **hence: compute $x^K$ directly with central-cut ellipsoid algorithm**

- **hence: compute $x^K$ directly with central-cut ellipsoid algorithm**
- **needs modified dual**

$$(D') \qquad \begin{aligned} U^T y &\leq -\frac{1}{2} \\ y &\geq 0 \end{aligned}$$

- **hence: compute $x^K$ directly with central-cut ellipsoid algorithm**

- **needs modified dual**

$$(D') \qquad \begin{aligned} U^T y &\leq -\frac{1}{2} \\ y &\geq 0 \end{aligned}$$

- **round $x^K$ with fixed precision**

- **end with polynomially many distributions** $x^K$

- **end with polynomially many distributions $x^K$**
- **recall: each $x := x^K$ has product distributions as components**

$$x_s = (x^1_{s^1}, \ldots, x^n_{s^n})$$

- **end with polynomially many distributions** $x^K$
- **recall: each** $x := x^K$ **has product distributions as components**

$$x_s = (x^1_{s^1}, \ldots, x^n_{s^n})$$

- $x^p_{s^p}$ **is distribution over strategy set** $S^p$

- **end with polynomially many distributions** $x^K$
- **recall: each** $x := x^K$ **has product distributions as components**

$$x_s = (x^1_{s^1}, \ldots, x^n_{s^n})$$

- $x^p_{s^p}$ **is distribution over strategy set** $S^p$
- **chicken game: symmetric product distributions**

# Exponential matrix product

- **collect $x^K$ in matrix $X$**

## Exponential matrix product

- **collect $x^K$ in matrix $X$**
- **consider matrix product $UX^T$:**

  **$U$ with exponentially many columns**

  **$X^T$ with exponentially many rows**

# Exponential matrix product

- collect $x^K$ in matrix $X$
- consider matrix product $UX^T$:

  $U$ with exponentially many columns

  $X^T$ with exponentially many rows

- computation takes exponential time

## Exponential matrix product

- collect $x^K$ in matrix $X$
- consider matrix product $UX^T$:

  $U$ with exponentially many columns

  $X^T$ with exponentially many rows

- computation takes exponential time
- but: $UX^T$ has only polynomially many entries

# Polynomial Expectation Property (1)

- **for each product distribution $x_s$ of a fixed $x := x^K$: concentrate probability of $p$ on strategy $i$**

## Polynomial Expectation Property (1)

- **for each product distribution $x_s$ of a fixed $x := x^K$: concentrate probability of $p$ on strategy $i$**
- **still a product distribution**

## Polynomial Expectation Property (1)

- for each product distribution $x_s$ of a fixed $x := x^K$: concentrate probability of $p$ on strategy $i$
- still a product distribution
- a game is said to have the polynomial expectation property [PAPADIMITRIOU 2005] if: exists poly-time algorithm $E$ for $p$'s utility on strategy $i$ under given distribution

# Polynomial Expectation Property (2)

**observe: two invocations of $E$ compute an entry of $UX^T$:**

$$\left[E(z,p,x^K(p \leftarrow i)) - E(z,p,x^K(p \leftarrow j))\right] x^K_{is^{-p}}$$

**is equivalent to**

$$\sum_{s^{-p} \in S^{-p}} \left[u^p(is^{-p}) - u^p(js^{-p})\right] \cdot x^K_{is^{-p}}$$

## Linear Programming: 2nd invocation

- **gives rise to new linear program**

$$(P') \qquad U\tilde{X}^T\alpha \geq 0$$
$$\alpha \geq 0.$$

# Linear Programming: 2nd invocation

- gives rise to new linear program

$$(P') \qquad\qquad \begin{aligned} U\tilde{X}^T\alpha &\geq 0 \\ \alpha &\geq 0. \end{aligned}$$

- recall infeasibility of (D') – also for $x^K$ (convex combination):

$$(D') \qquad\qquad \begin{aligned} \left[XU^T\right]y &\leq -\frac{1}{2} \\ y &\geq 0 \end{aligned}$$

## Linear Programming: 2nd invocation

- **gives rise to new linear program**

$$(P') \qquad \begin{aligned} U\tilde{X}^T\alpha &\geq 0 \\ \alpha &\geq 0. \end{aligned}$$

- **recall infeasibility of (D') – also for $x^K$ (convex combination):**

$$(D') \qquad \begin{aligned} \left[XU^T\right]y &\leq -\frac{1}{2} \\ y &\geq 0 \end{aligned}$$

- **FARKAS' Lemma: (P') has non-zero solution $\alpha$**

# Finish with Correlated Equilibrium

- scale $\alpha$ to distribution

# Finish with Correlated Equilibrium

- scale $\alpha$ to distribution
- so $x := X^T\alpha$ non-zero solution of (P):

$$(P) \qquad \begin{aligned} Ux &\geq 0 \\ x &\geq 0. \end{aligned}$$

# Finish with Correlated Equilibrium

- scale $\alpha$ to distribution
- so $x := X^T\alpha$ non-zero solution of (P):

$$(P) \qquad \begin{aligned} Ux &\geq 0 \\ x &\geq 0. \end{aligned}$$

- gives correlated equilibrium. $\square$

## Trusted Intermediary

- **randomized algorithm in polynomial-time**

## Trusted Intermediary

- **randomized algorithm in polynomial-time**
- **take as input the polynomially many** $x^K$ **and the non-zero** $\alpha$

## Trusted Intermediary

- randomized algorithm in polynomial-time
- take as input the polynomially many $x^K$ and the non-zero $\alpha$
- first choose one of the $x^K$

## Trusted Intermediary

- **randomized algorithm in polynomial-time**
- **take as input the polynomially many $x^K$ and the non-zero $\alpha$**
- **first choose one of the $x^K$**
- **then within this, choose a strategy for each player (according to distribution)**

## Trusted Intermediary

- randomized algorithm in polynomial-time
- take as input the polynomially many $x^K$ and the non-zero $\alpha$
- first choose one of the $x^K$
- then within this, choose a strategy for each player (according to distribution)
- this suggests one strategy for each player

## Extensions and open problems

- **framework applicable to most classes of strategic-form games**

## Extensions and open problems

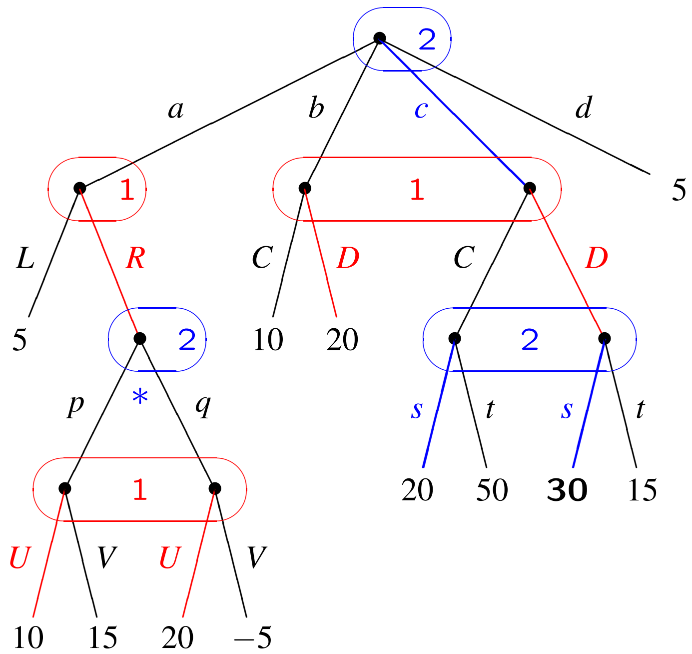
- framework applicable to most classes of strategic-form games
- requires specific algorithm for product distributions

## Extensions and open problems

- framework applicable to most classes of strategic-form games
- requires specific algorithm for product distributions
- also polynomial expectation property needs adaptation

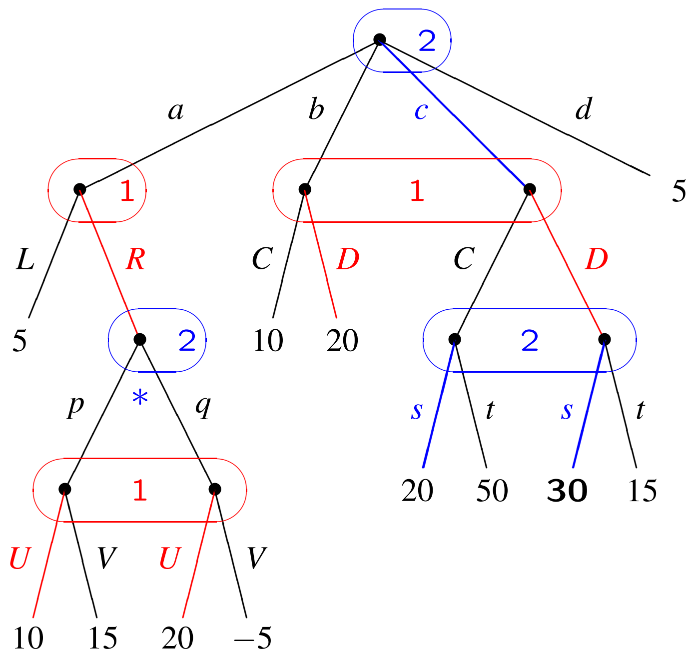
# Open for extensive-form games

- **Trees with imperfect information**



# Open for extensive-form games

- **Trees with imperfect information**




- **Number of pure strategies exponential in size of extensive game**

# References


1. Hart, Schmeidler (1989): Existence of Correlated Equilibria, Math. of Op. Res., 14
2. Nau, McCardle (1990): Coherent Behaviour in Noncooperative Games, J. of Econ. Theory, 50
3. Papadimitriou (2005): Computing Correlated Equilibria in Multi-Player Games, STOC Proc. 37