Economics, Game Theory and Computer Science

Krzysztof R. Apt

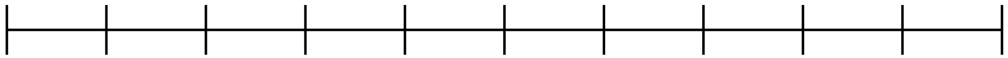
CWI and University of Amsterdam

Warning about Grand Challenges

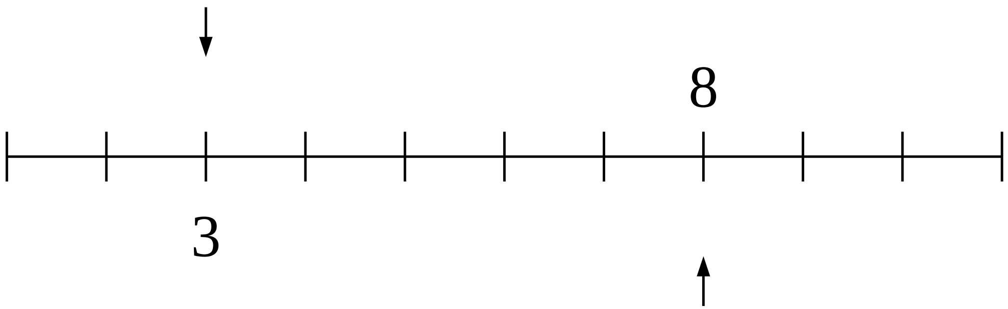
- AI.
- Fifth Generation Project.
- Iraq.
- . . .

Strategic Games: an Example

Location game (Hotelling '29)



Where should I place my bakery?
Example:



Then
baker$_1(3, 8) = 5$,
baker$_2(3, 8) = 6$.

Strategic Games

Given n players a strategic game is a sequence

$$(S_1, \ldots, S_n, p_1, \ldots, p_n),$$

where

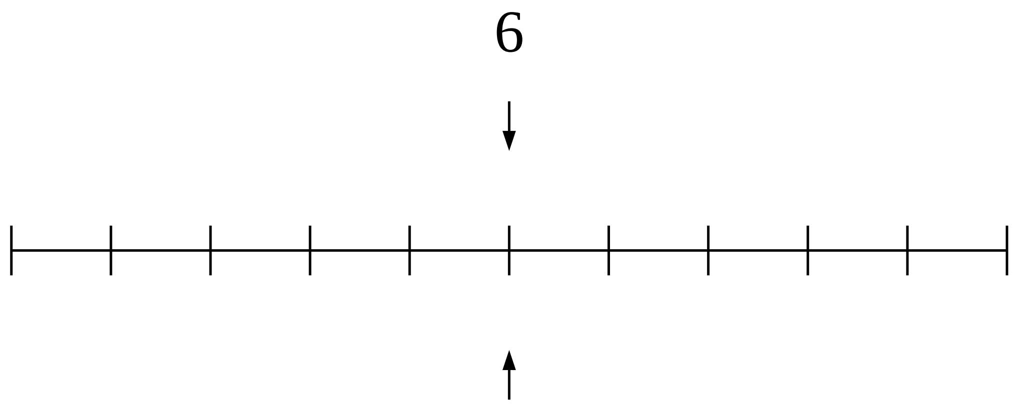
- each S_i is a non-empty set of strategies available to player i,
- p_i is the payoff function for the player i:

$$p_i : S_1 \times \ldots \times S_n \rightarrow \mathcal{R}.$$

Nash Equilibrium

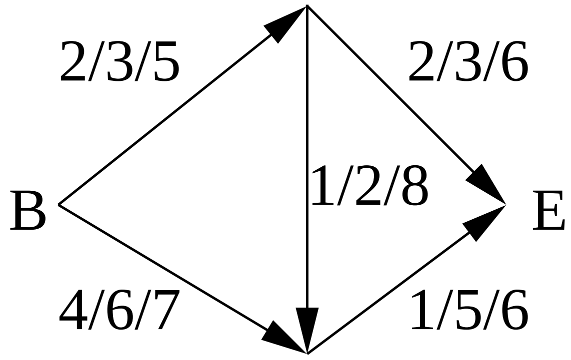
- Given s let $s_{-i} := (s_1, \ldots, s_{i-1}, s_{i+1}, \ldots, s_n)$.
- Strategy s_i of player i is a best response to s_{-i} iff s_i is a maximum of $p_i(\cdot, s_{-i})$.
- Joint strategy s is a Nash equilibrium if each s_i is a best response to s_{-i}.
- Intuition: no player has a posteriori regrets.

Example: Nash equilibrium in location game:



Example: Congestion Games

(Rosenthal '73)



- Best response dynamics.

Nash Theorem

Mixed extension of a finite game:

- strategies for player i: probability distributions over his set of strategies,
- payoff for player i defined as the aggregated payoff:
  $p_i(m_1, \ldots, m_n) := \sum_{s \in S_1 \times \ldots \times S_n} prob(s) \cdot p_i(s)$,
  where
  $prob(s_1, \ldots, s_n) := m_1(s_1) \cdot \ldots \cdot m_n(s_n)$.

Nash Theorem Every mixed extension of a finite game has a Nash equilibrium.

Complexity of Nash equilibria

Papadimitriou (SODA '01):

“But the most interesting aspect of the Nash equilibrium concept for our community is that *it is a most fundamental computational concept whose complexity is wide open.* “

Strategic Games: a CS View

Assumption: all players are directly connected.

```
[P_1 || ... || P_n]
```

where

```
P_i ::
choose(s_i);
rec_1  : = false; ...; rec_n  : = false;
sent_1 : = false; ...; sent_n : = false;
*[[](j <> i) not rec_j,  P_j? s_j -> rec_j  := true
  [](j <> i) not sent_j, P_j! s_i -> sent_j := true
 ];
payoff_i := p_i(s_1,...,s_n)
```

Underlying Assumptions

- Each player has full knowledge of all strategies and payoff functions (complete information).
- Each player wants to maximize his payoff (is rational).
- Each player believes all other players have complete information and are rational (common knowledge of complete information and of rationality).

Consequently

- `choose(s_i)` hides a non-trivial reasoning.
- This reasoning is based on epistemic analysis.

Example

Consider the following strategic game:

	L	M	R
T	$3, 2$	$2, 1$	$1, 1$
C	$2, 1$	$1, 1$	$4, 0$
B	$1, 1$	$1, 1$	$0, 4$

Which strategies should the players choose?

B is strictly dominated by T. Eliminating it we get

	L	M	R
T	$3, 2$	$2, 1$	$1, 1$
C	$2, 1$	$1, 1$	$4, 0$

Now R is strictly dominated by L.

Example, ctd

By eliminating it we get:

	L	M
T	$3,2$	$2,1$
C	$2,1$	$1,1$

Now C is strictly dominated by T, so we get:

	L	M
T	$3,2$	$2,1$

But now M is strictly dominated by L, so we get:

	L
T	$3,2$

Conclusion: the players should choose respectively T and L.
It is the unique Nash equilibrium.

Strategic Games: Game Theoretic View

It focuses on topics such as:

- Appropriate notions of equilibrium and game reductions.
- Incomplete information (Bayesian games).
- Epistemic foundations of players' behaviour.
- Design of games in which players ensure desired common outcome by being egoistic (rational). (Mechanism design, special case: auctions).
- How to prevent cheating (strategic behaviour).

Strategic Games: Game Theoretic Analysis

It ignores topics such as:

- How information is encoded, transmitted, retrieved, verified, . . .
- How decisions are taken in distributed environments.
- How consensus is reached in distributed environments.

A Small Type of a Grand Challenge

Find a theory of processes that combines the CS and Game Theory views and captures the concepts of:

- communication,
- consensus,
- competition,
- incomplete information,
- epistemic reasoning,
- strategic behaviour,
- ...

(Communicating rational processes).

Example

Suppose now

	L	M	R
T	$3, 2$	$2, 1$	$1, ?$
C	$2, 1$	$1, 1$	$4, 0$
B	$1, 1$	$1, 1$	$0, 4$

and that $p_2(T, R) = 5$.
Player 1 can only reduce the game to

	L	M	R
T	$3, 2$	$2, 1$	$1, ?$
C	$2, 1$	$1, 1$	$4, 0$

even if he learns that $p_2(T, R) = 5$.

Revised Decision Making

	L	M	R
T	$3, 2$	$2, 1$	$1, 5$
C	$2, 1$	$1, 1$	$4, 0$

```
P_1 ::
[       K_1 p_2(T,R) -> x := p_2(T,R)
[] not K_1 p_2(T,R) -> P_2? x
];
[ x <  2 -> s_1 := T
  x >= 2 -> s_1 := 1/2 * T + 1/2 * C
]
```

So if $p_2(T, R) = 5$, then both P_1 and P_2 get 2.5.

Is that so?

Revised Decision Making, ctd

If $p_2(T, R) = 5$ and P_2 knows that P_1 does not know it, then

- P_2 should cheat and report to P_1 that $p_2(T, R) = 1$.
- Then P_1 will reduce the game to

$$\begin{array}{cc} & L \\ T & \boxed{3, 2} \end{array}$$

  and choose T.
- Knowing this P_2 will choose R.
- P_1 will get 1 and P_2 will get 5, since

$$\begin{array}{cc} & R \\ T & \boxed{1, 5} \end{array}$$

Revised Decision Making, ctd

	L	M	R
T	$3,2$	$2,1$	$1,5$
C	$2,1$	$1,1$	$4,0$

```
P_2 ::
[  K_2 K_1 p_2(T,R)          ->
 [  p_2(T,R) < 2             -> s_2 := L
 [] 2 <= p_2(T,R) <= 3 -> s_2 := L
 [] 3 < p_2(T,R)             -> s_2 := R
 ]
[] K_2 not K_1 p_2(T,R) ->
 [  p_2(T,R) <  2 -> P_1! p_2(T,R); s_2 := L
 [] p_2(T,R) >= 2 -> P_1!1; s_2 := R
 ]
]
```

Conclusion

- Strategic behaviour can pay off in presence of common knowledge of rationality and absence of complete information.
- In a more complicated setup strategic behaviour is more difficult to realize:

	L	M	R
T	$3, ?$	$2, ?$	$1, ?$
C	$2, ?$	$1, ?$	$4, ?$
B	$1, ?$	$1, ?$	$0, ?$

	L	M	R
T	$?, 2$	$?, 1$	$?, 1$
C	$?, 1$	$?, 1$	$?, 0$
B	$?, 1$	$?, 1$	$?, 4$

An Interview with Robert Aumann (2005)

Aumann: [...] In computer science we have distributed computing, in which there are many different processors. The problem is to coordinate the work of these processors, which may number in the hundreds of thousands, each doing its own work.

Hart: That is, how processors that work in a decentralized way reach a coordinated goal.

An Interview with Robert Aumann, ctd

Aumann: Exactly. Another application is protecting computers against hackers who are trying to break down the computer. this is a very grim game, just like war is a grim game, and the stakes are high; but it is a game. That's another kind of interaction between computers and game theory.
Still another comes from computers that solve games, play games, and design games —like auctions— particularly on the Web. These are applications of computers to games.
[...]