Introduction to Game Theory

Bernhard Nebel, Robert Mattmüller, Stefan Wölfl, & Christian Becker-Asano

Sommersemester 2013

Definition: Let G be a Bayesian game

A pure strategy of player i is any mapping

$$s_i : \Theta_i \longrightarrow A_i .$$

A mixed strategy of player i is any mapping

$$s_i : \Theta_i \longrightarrow \Delta A_i$$

In the latter case define

$$s_i(a_i \mid \vartheta_i) := s_i(\vartheta_i)(a_i)$$

i.e., as the probability of player i that he plays action a_i under the condition that his type is ϑ_i.

In the context of Bayesian games we have different notions of expected utility:

* ex-ante: the player knows nothing about which player has which type (including himself)
* ex-interim: the player knows his own type only
* ex-post: the player knows the type of all players (incl. himself)

Assume in what follows a common prior p.

Ex-post:

$$U_i(s,\theta) := \sum_{a\in A} \left(\prod_{j\in N} s_j(a_j|\theta_j)\right)\cdot u_i(a,\theta)$$

Ex-interim:

$$U_i(s,\theta_i) := \sum_{\theta_{-i}\in\Theta_{-i}} p(\theta_{-i}|\theta_i)\cdot U_i(s,\underbrace{\theta_{-i},\theta_i}_{\theta})$$

Ex-ante:

$$U_i(s) := \sum_{\theta_i\in\Theta_i} p(\theta_i)\cdot U_i(s,\theta_i)$$

$$\overset{!}{=} \sum_{\theta\in\Theta} p(\theta)\cdot U_i(s,\theta)$$

Define $\quad B_i(s_{-i}) := \underset{s_i\in S_i}{\arg\max}\ U_i(s_{-i},s_i)$

Definition: A Bayes-NE is a mixed strategy profile $s\in S$ such that for each $i\in N$, $s_i\in B_i(s_{-i})$

A Bayesian game G can be represented as a strategic game in normal form G'. Define G' as follows:

$$N' := N$$

$$A'_i := \{\text{pure strategies of } i \text{ in } G\}$$

$$u'_i(\underline{s}) := U_i^G(s)$$

One can prove: the Bayes-NE of G are exactly the NE of the induced game G'.

Thus each finite Bayesian game has a Bayes-NE!

Example:

ϑ_{11}

	L	R
T	2, 0	0, 2
B	0, 2	2, 0

$p = 0.3$

	L	R
T	2, 2	0, 3
B	3, 0	1, 1

$p = 0.1$

ϑ_{12}

	L	R
T	2, 2	0, 0
B	0, 0	1, 1

$p = 0.2$

ϑ_{21}

	L	R
T	2, 1	0, 0
B	0, 0	1, 2

$p = 0.4$

ϑ_{22}

S_1

ϑ_{11}	ϑ_{12}
T	T
T	B
B	T
B	B

S_2

ϑ_{21}	ϑ_{22}
L	L
L	R
R	L
R	R

Now calculate for all strategy profiles s $U_i^G(s)$, ...

For example:

$$u_1'(TT, LR) = U_1^G(TT, LR) = \sum_{\vartheta \in \Theta} p(\vartheta) \cdot U_1(TT, LR, \vartheta)$$

$$= p(\vartheta_{11}, \vartheta_{21}) \cdot u_1(T, L, \vartheta_{11}, \vartheta_{21}) + p(\vartheta_{11}, \vartheta_{22}) \cdot u_1(T, R, \vartheta_{11}, \vartheta_{22})$$

$$+ p(\vartheta_{12}, \vartheta_{21}) \cdot u_1(T, L, \vartheta_{12}, \vartheta_{21}) + p(\vartheta_{12}, \vartheta_{22}) \cdot u_1(T, R, \vartheta_{12}, \vartheta_{22})$$

$$= 0.3 \cdot 2 + 0.1 \cdot 0 + 0.2 \cdot 2 + 0.4 \cdot 0$$

$$= \underline{1}$$

	LL	LR	RL	RR
TT	2, 1	1, 0.7	1, 1.2	0, 0.9
TB	..	..	..	..
BT	..	--	---	...
BB	...	...	..	...

Now you can start to use the methods learned before!

2.10 Solving Strategic Games

In this section we will how to find mixed NE by using methods known from linear optimization. We will consider 2 player games only.

Zerosum games: By the Nash theorem we know that each finite strategic game has a NE (in mixed strategies). By the Proposition about NE and MM (in section 2.6), each NE is a pair of maximinimizers. Thus, it is sufficient to search for pairs of maximinimizers.

Idea: Consider the mixed strategies of player 1. For each such strategy α find a response β_α of player 2 that minimizes the utility of player 1. By the Support Lemma it suffices to consider the pure strategies of player 2.

Given this setting, we try to maximize $U_1(\alpha, \beta_\alpha)$ → Optimization problem

Excursion: Linear Optimization Problems / Linear Programming

A linear optimization problem (LOP) over real-valued variables $x_1, \ldots, x_n$ is defined by a finite set of linear in-/equality constraints of the form:

$$\sum_{1 \leq i \leq n} a_{ji} x_i \circledast c_j$$

with $a_{ji}, c_j \in \mathbb{R}$ and $\circledast \in \{\leq, \geq, =, \cancel{<}, \cancel{>}\}$

and an objective function

$$F(x_1, \ldots, x_n) = \sum_{1 \leq i \leq n} b_i x_i$$

($b_i \in \mathbb{R}$) that is to be minimized.

Each LOP can be transformed into normal form:

$$a_{11} x_1 + a_{12} x_2 + \dots + a_{1n} x_n = c_1$$

$$\vdots$$

$$a_{m1} x_1 + a_{m2} x_2 + \dots + a_{mn} x_n = c_m$$

$$\left.\begin{array}{l} x_1 \geq 0 \\ \vdots \\ x_n \geq 0 \end{array}\right\} \text{Non-negativity constraints}$$

<u>Maximization problems</u>: To maximize an objective function $F(x_1, \dots, x_n)$, simply consider

$$\tilde{F}(x_1, \dots, x_n) := -F(x_1, \dots, x_n)$$

$\geq$ - Inequalities: Linear constraints of the form

$\sum_i a_{ji} x_i \geq c_j$ can simply be replaced by

$$\sum_i - a_{ji} x_i \leq - c_j$$

$\leq$ - Inequalities. Linear constraints of the form

$\sum_i a_{ji} x_i \leq c_j$ can be transformed into expressions

in normal form:

$$\left(\sum_i a_{ji} x_i \right) + \underline{\underline{s_j}} = c_j$$

$$\underline{\underline{s_j}} \geq 0 \qquad \text{slack variable}$$

Non-negativity constraints:

To ensure the non-negativity of the variables we can introduce for a variable x_i two variables x_i' and x_i'', add the constraints $x_i' \geq 0$, $x_i'' \geq 0$, and substitute x_i by $x_i' - x_i''$ in all constraints and the objective function.

References

Holler, Manfred J. & Gerhard Illing,
Einführung in die Spieltheorie,
Springer-Verlag, Berlin, 5. Aufl., 2002.

Nisan, Noam, Tim Roughgarden, Éva Tardos, & Vijay V. Vazirani (eds.),
Algorithmic Game Theory,
Cambridge University Press, 2007.

Osborne, Martin J. & Ariel Rubinstein,
A Course in Game Theory,
MIT Press, Cambridge, MA, 1994.