第三期矩阵半张量积暑期班

第3讲: 有限博弈的半张量积方法
3.3 演化博弈

李长喜
lichangxi@pku.edu.cn

北京大学, 工学院

2023年08月14日, 聊城大学

主要内容

1. 重复博弈的局势演化方程

☞ 重复博弈

设$G=(N,S,C)$ 为一个有限正规博弈. 如果这个博弈被重复多次(无穷次), 那么, 由理性的假定, 每个玩家都会根据已有信息, 设法最大化自己利益. 设$|N|=n$, $|s_i|=k_i$, $i=1,\cdots,n$. 用$x_i(t+1)$ 表示玩家i 在$t+1$ 次博弈时的策略, 那么, 有以下的演化方程

$$\begin{cases} x_1(t+1)=f_1(x(t),x(t-1),\cdots,x(0)) \\ x_2(t+1)=f_2(x(t),x(t-1),\cdots,x(0)) \\ \vdots \\ x_n(t+1)=f_n(x(t),x(t-1),\cdots,x(0)), \end{cases} \tag{1}$$

这里$x(\tau)\sim(x_1(\tau),\cdots,x_n(\tau))$ 表示τ 时刻新有策略变量.

☞ 局势演化方程

最常见的一种演化方程, 其下一时刻策略仅依赖于当下的策略(马氏型), 于是, 演化方程变为

$$\begin{cases} x_1(t+1) = f_1(x_1(t), x_2(t), \cdots, x_n(t)) \\ x_2(t+1) = f_2(x_1(t), x_2(t), \cdots, x_n(t)) \\ \vdots \\ x_n(t+1) = f_n(x_(t), x_2(t), \cdots, x_n(t)). \end{cases} \tag{2}$$

局势演化方程大致可以进一步分为两种:

- 确定型: 这时, $x_i(t) \in \mathcal{D}_{k_i}$, $i = 1, \cdots, n$. 在向量形式下, 则$x_i(t) \in \Delta_{k_i}$. 则在向量形式下有

$$x_i(t+1) = M_i x(t), \quad i = 1, \cdots, n, \tag{3}$$

这里, $x(t) = \ltimes_{i=1}^{n} x_i(t)$, $M_i \in \mathcal{L}_{k_i \times k}$ 为f_i 的结构矩阵, $i = 1, \cdots, n$. 各式相乘, 则得其代数状态空间方程:

$$x(t+1) = Mx(t), \tag{4}$$

这里,

$$M = M_1 * M_2 * \cdots * M_n \in \mathcal{L}_{k \times k}. \tag{5}$$

- 概率型: 这时, 状态变量$x(t) = (r_1, r_2, \cdots, r_k)^T \in \Upsilon_k$ 表示, $P(x(t) = \delta_k^j) = r_j, j = 1, \cdots, k.$ 这时, (3) 仍成立, 只是$Col_j(M_i) \in \Upsilon_{k_i}$, 它表示$x(t) = \delta_k^j$ 时$x_i(t+1)$ 的概率分布. 如果每个个体更新其策略的时候满足条件独立的假设, 则(4) 也仍然有效, 但M 的(i,j) 元, 记着$m_{i,j}$, 它表示, $x(t) = \delta_k^j$ 时$x(t+1) = \delta_k^i$ 的概率. 即

$$m_{ij} = P\left\{x(t+1) = \delta_k^i \mid x(t) = \delta_k^j\right\}, \quad i,j = 1, \cdots, k. \tag{6}$$

因此, M 是列概率转移矩阵. (5) 可写成

$$M = M_1 * M_2 * \cdots * M_n \in \Upsilon_{k \times k}. \tag{7}$$

[1] Changxi Li, Xiao Zhang, Jun-e Feng, and Daizhan Cheng. Transition analysis of stochastic logical control networks. IEEE Transactions on Automatic Control, 2023, DOI: 10.1109/TAC.2023.3281986.

2. 策略更新规则

局势演化方程, 或者说, 每个玩家的策略演化方程, 都是由他们所采用的策略更新规则(strategy updating rule) 来决定的. 目前, 在理论研究中常用的策略更新规则一般是由专家们设计出来的. 下面举出几种常用的.

- 短视最优响应 (myopic best response adjustment, MBRA).

站在玩家i 的立场上, 考察其他人在t 时刻的策略$s_{-i}(t)$, 选择对付他们的最佳策略, 记作

$$O_i(t) = \text{argmax}_{s_i \in S_i} c_i(s_i, s_{-i}(t)).$$

对于短视最优响应, 各玩家更新时间很重要. 我们对此做以下划分:

- 时间串联型(sequential MBRA): 一个时刻只有一个玩家更新策略. 它还可以细分为
  - 周期型串联(periodical MBRA): 玩家按顺序轮流更新:

$$\begin{cases} x_i(t+1) = f_i(x_1(t), \cdots, x_n(t)), \\ x_j(t+1) = x_j(t), j \neq i; t = kn + (i-1),\ k = 0, 1, 2, \cdots. \end{cases} \tag{8}$$

  - 随机型串联(stochastic MBRA): 每个玩家以相同的概率($p = \frac{1}{n}$) 被选上更新自己策略.

- 时间并联型(Parallel MBRA): 所有玩家同时更新他们的策略. 此时, 演化方程即为(4).
- 时间级联型(Cascading MBRA): 虽然所有玩家同时更新他们的策略, 但当玩家j 更新它的策略时, 它知道并使用玩家i, $i<j$, 的新策略. 即,

$$\begin{cases} x_1(t+1) = f_1(x_1(t), \cdots, x_n(t)) \\ x_2(t+1) = f_2(x_1(t+1), x_2(t), \cdots, x_n(t)) \\ \vdots \\ x_n(t+1) = f_n(x_1(t+1), \cdots, x_{n-1}(t+1), x_n(t)). \end{cases} \tag{9}$$

- 带参数$\tau > 0$ 的对数响应 (Logit Response (LR) with Parameter $\tau > 0$)

第i 个玩家在$t+1$ 时刻取策略$j \in S_i$ 的概率为

$$P_\tau^i\left(x_i(t+1) = j | x(t)\right) = \frac{\exp\left(\frac{1}{\tau} c_i\left(j, x^{-i}(t)\right)\right)}{\sum_{s_i \in S_i} \exp\left(\frac{1}{\tau} c_i\left(s_i, x^{-i}(t)\right)\right)}. \tag{10}$$

- 无条件模仿 (Unconditional Imitation (UI))
  玩家i 在所有玩家中选t 时刻收益最好的玩家, 取其策略为自己下一时刻的策略. 如果最优玩家不唯一,
  - 1-型UI: 取指标最小的. 即设

$$j^* = \min\{\mu | \mu \in \operatorname{argmax}_j \ c_j(x(t))\}. \tag{11}$$

    则

$$x_i(t+1) = x_{j^*}(t). \tag{12}$$

  - 2-型UI:以相同概率取其中任意一个. 即如果

$$\operatorname{argmax}_j \ c_j(x(t)) := \{j_1^*, \cdots, j_r^*\},$$

    则取

$$x_i(t+1) = x_{j_\mu^*}(t), \text{with probability } \ p_\mu^i = \frac{1}{r}, \mu = 1, \cdots, r. \tag{13}$$

- Fermi 规则 (Fermi's Rule (FM)). 以等概率任选一个玩家j. 然后比较j 与自己的上一次收益, 然后以如下方法决定下一次策略:

$$x_i(t+1) = \begin{cases} x_j(t), & \text{以概率 } p_t \\ x_i(t), & \text{以概率 } 1-p_t, \end{cases} \tag{14}$$

这里p_t 由以下Fermi 函数决定:

$$p_t = \frac{1}{1 + \exp(-\mu(c_j(t) - c_i(t)))}.$$

其中,参数$\mu > 0$ 可任选. 特别是, 当$\mu = \infty$ 时可得

$$x_i(t+1) = \begin{cases} x_i(t), & c_i(x(t)) \geq c_j(x(t)) \\ x_j(t), & c_i(x(t)) < c_j(x(t)). \end{cases} \tag{15}$$

3. 从更新策略到演化方程

本节讨论如何由策略更新规则得到策略及局势演化方程. 可以说, 局势演化方程是完全由策略更新规则所确定的. 在上一节, 我们对几种策略更新规则作了十分详尽的描述, 就是为了使它能唯一确定局势演化方程. 下面, 我们通过具体例子来说明怎样由策略更新规则确定局势演化方程.

Example 3.1

考察一个重复博弈$G \in \mathcal{G}_{[3;3,2,3]}$, 其支付矩阵为:

表 1: 例3.1 支付矩阵

$C\backslash P$	111	112	113	121	122	123	211	212	213
c_1	1	$\underline{2}$	-1	-2	0	1	-2	1	$\underline{1}$
c_2	2	$\underline{3}$	$\underline{4}$	$\underline{3}$	2	1	$\underline{3}$	2	$\underline{2}$
c_3	-2	-1	$\underline{0}$	-4	$\underline{\underline{-2}}$	-3	-3	-2	$\underline{0}$
$C\backslash P$	221	222	223	311	312	313	321	322	323
c_1	$\underline{1}$	0	$\underline{2}$	$\underline{3}$	$\underline{2}$	$\underline{1}$	-1	$\underline{2}$	-2
c_2	2	$\underline{3}$	1	3	2	$\underline{4}$	$\underline{5}$	$\underline{3}$	1
c_3	-1	-1	$\underline{0}$	$\underline{0}$	-3	-3	-2	$\underline{\underline{-1}}$	$\underline{\underline{-1}}$

我们讨论更新规则如何确定形势演化方程.

Example 3.1(cont'd)

考虑策略更新规则为短视最优响应(MBRA):
对玩家1, 比较$c_1(111)$, $c_1(211)$, 与$c_1(311)$, 因$c_1(111) = 1$, $c_1(211) = -2$, 与$c_1(311) = 3$, 故当玩家2 及玩家3 都取策略1 时, 玩家1 的最佳响应是取策略3. 也就是说: $f_1(111) = f_1(211) = f_1(311) = 3$. 再比较$c_1(112) = 2$, $c_1(212) = 1$, 与$c_1(312) = 2$, 这时, 如果考虑确定型更新(MBRA-D), 则有$f_1(112) = f_1(212) = f_1(312) = 1$. 如果考虑概率型更新(MBRA-P), 则有$f_1(112) = f_1(212) = f_1(312) = 1(\frac{1}{2}) + 3(\frac{1}{2})$. 后面这个记号表示, 取1 的概率为$\frac{1}{2}$, 取3 的概率为$\frac{1}{2}$. 类此, 就可以得到$f_1$ 的结构矩阵. 同样, f_2, f_3 的结构矩阵也可得到.

Example 3.1(cont'd)

记

$$\begin{cases} x_1(t+1) = f_1(x_1(t), x_2(t), x_3(t)) = M_1 \ltimes_{i=1}^{3} x_i(t) := M_1 x(t) \\ x_2(t+1) = f_2(x_1(t), x_2(t), x_3(t)) = M_2 \ltimes_{i=1}^{3} x_i(t) := M_2 x(t) \\ x_3(t+1) = f_3(x_1(t), x_2(t), x_3(t)) = M_1 \ltimes_{i=1}^{3} x_i(t) := M_3 x(t). \end{cases} \tag{16}$$

最后可得到局势演化方程

$$x(t+1) = (M_1 * M_2 * M_3) x(t) := M x(t). \tag{17}$$

Example 11.3.1(cont'd)

这里

- MBRA-D:

$$\begin{aligned} M_1 &= \delta_3[3,1,2,2,3,2,3,1,2,2,3,2,3,1,2,2,3,2] \\ M_2 &= \delta_2[2,1,1,2,1,1,1,2,1,1,2,1,2,2,1,2,2,1] \\ M_3 &= \delta_3[3,3,3,2,2,2,3,3,3,3,3,3,1,1,1,2,2,2]. \end{aligned} \tag{18}$$

$$M = \delta_{18}[18,3,9,11,14,8,15,6,9,9,18,9,16,4,7,11,17,8]. \tag{19}$$

- MBRA-P:

Example 11.3.1(cont'd)

$$
\begin{aligned}
M_1 &= \delta_3[3, 1(\tfrac{1}{2}) + 3(\tfrac{1}{2}), 2(\tfrac{1}{2}) + 3(\tfrac{1}{2}), \\
&\quad 2, 3, 2, 3, 1(\tfrac{1}{2}) + 3(\tfrac{1}{2}), 2(\tfrac{1}{2}) + 3(\tfrac{1}{2}), \\
&\quad 2, 3, 2, 3, 1(\tfrac{1}{2}) + 3(\tfrac{1}{2}), 2(\tfrac{1}{2}) + 3(\tfrac{1}{2}), 2, 3, 2] \\
M_2 &= \delta_2[2, 1, 1, 2, 1, 1, 1, 2, 1, 1, 2, 1, 2, 2, 1, 2, 2, 1] \\
M_3 &= \delta_3[3, 3, 3, 2, 2, 2, 3, 3, 3, 3, 3, 3, 1, 1, 1, 2(\tfrac{1}{2}) + 3(\tfrac{1}{2}), \\
&\quad 2(\tfrac{1}{2}) + 3(\tfrac{1}{2}), 1(\tfrac{1}{3}) + 2(\tfrac{1}{3}) + 3(\tfrac{1}{3})].
\end{aligned}
$$

$$
\begin{aligned}
M &= \delta_{18}[18, 3(\tfrac{1}{2}) + 15(\tfrac{1}{2}), 9(\tfrac{1}{2}) + 15(\tfrac{1}{2}), 11, 14, 8, 15, \\
&\quad 6(\tfrac{1}{2}) + 18(\tfrac{1}{2}), 9(\tfrac{1}{2}) + 15(\tfrac{1}{2}), 9, 18, 9, 16, \\
&\quad 4(\tfrac{1}{2}) + 10(\tfrac{1}{2}), 7(\tfrac{1}{2}) + 13(\tfrac{1}{2}), 11(\tfrac{1}{2}) + 12(\tfrac{1}{2}), \\
&\quad 17(\tfrac{1}{2}) + 18(\tfrac{1}{2}), 7(\tfrac{1}{3}) + 8(\tfrac{1}{3}) + 9(\tfrac{1}{3})].
\end{aligned}
$$

Example 11.3.1(cont'd)

下面考虑更新时间. 实际上, 由前面的操作过程不难看出, (16) 是时间并联型更新. 对于确定型(MBRA-D) 演化方程, 我们将其改造成时间级联型更新. 首先有

$$x_1(t+1) = M_1 x(t) := \tilde{M}_1 x(t),$$

这里, $\tilde{M}_1 = M_1$. 其次

$$\begin{aligned} x_2(t+1) &= M_2 x_1(t+1) x_2(t) x_3(t) \\ &= M_2 M_1 x_1(t) x_2(t) x_3(t) x_2(t) x_3(t) \\ &= M_2 M_1 x_1(t) O_6^R x_2(t) x_3(t) \\ &= M_2 M_1 \left(I_3 \otimes O_6^R\right) x(t) \\ &:= \tilde{M}_2 x(t), \end{aligned}$$

Example 11.3.1(cont'd)

于是有

$$\begin{aligned}\tilde{M}_2 &= M_2 M_1 (I_3 \otimes O_6^R) \\ &= \delta_2[2,1,1,1,2,1,2,1,1,1,2,1,2,1,1,1,2,1].\end{aligned}$$

同样地, 可得

$$\begin{aligned}x_3(t+1) &= M_3 x_1(t+1) x_2(t+1) x_3(t) \\ &:= \tilde{M}_3 x(t),\end{aligned}$$

Example 11.3.1(cont'd)

这里

$$\begin{aligned}\tilde{M}_3 &= M_3 M_1 \left(I_{18} \otimes \tilde{M}_2\right) O_{18}^R \left(I_6 \otimes O_3^R\right) \\ &= \delta_3[2,3,3,3,2,3,2,3,3,3,2,3,2,3,3,3,2,3].\end{aligned}$$

最后可得局势演化方程为

$$x(t+1) = Lx(t),$$

这里,

$$\begin{aligned}L &= \tilde{M}_1 * \tilde{M}_2 * \tilde{M}_3 \\ &= [17,3,9,9,17,9,17,3,9,9,17,9,17,3,9,9,17,9].\end{aligned}$$

有些策略更新规则一般只用于网络演化博弈. 此时, 有一个网络图. 例如, 无条件模仿. 如果将它用于一般演化博弈, 那么, 一次演化后大家就都一样了, 这种情况没什么意义. 但在网络中, 每一个玩家只在它邻域中选择模仿对象, 这样, 就会出现丰富的演化过程.

再从演化方程的形式看, (2) 是最常见的一种, 也是本书主要的研究对象. 虽然, 上一节介绍的几种策略更新规则都会导致这种形式的演化方程, 但并不是每一种策略更新规则都如此.

4. 策略的收敛性

与微分方程类似, 策略演化方程最强的稳定性是全局“渐近稳定”. 但由于有限博弈策略的有界(限)性, 只要全局收敛就足够了. 仿照微分方程理论, 我们可以通过构造李雅普诺夫函数的方法来验证收敛性.

Definition 11.4.1

给定一个演化博弈G, 设$|N| = n$, $|S_i| = k_i$, $i = 1, \cdots, n$, $k := \prod_{i=1}^{n} k_i$.

- 一个伪逻辑函数$\psi : \Delta_k \to \mathbb{R}$ 称为G 的一个李雅普诺夫函数, 如果

$$\psi(x(t+1)) - \psi(x(t)) \geq 0, \quad t \geq 0, \tag{20}$$

并且, 如果$\psi(x(t+1)) = \psi(x(t))$ 则$x(t+1) = x(t)$.

Definition 11.4.1 (cont'd)

- 当使用混合策略时, ψ 应当用其期望值代替, 即, $E\psi : \Upsilon_k \to \mathbb{R}$ 满足

$$E\psi(x(t+1)) - E\psi(x(t)) \geq 0, \quad t \geq 0, \tag{21}$$

并且, 如果$E\psi(x(t+1)) = E\psi(x(t))$ 则$Ex(t+1) = Ex(t)$.

由定义可推得如下结论.

Theorem 11.4.2

一个演化博弈, 如果存在一个李雅普诺夫函数, 则它一定会收敛到一个平衡点.

注意, 平衡点未必唯一. 因此, 收敛到那个平衡点依赖于初值.
设一个演化博弈如前. 其局势演化方程为

$$x(t+1) = Tx(t). \tag{22}$$

容易验证以下的引理:

Lemma I1.4.3

一个演化博弈G 具有李雅普诺夫函数, 当且仅当存在一个行向量$V_\psi \in \mathbb{R}^k$, 使得

$$V_\psi (T - I_k) \geq 0.$$

而且, 如果$V_\psi (T - I_k) = 0$ 则有j 使$Col_j(T) = \delta_k^j$.

利用这个引理可以得到

Theorem 11.4.4

设演化博弈G具有局势演化方程(22)，那里$T = \delta_k[i_1, i_2, \cdots, i_k]$. G具有李雅普诺夫函数$V_\psi = [a_1, a_2, \cdots, a_k]$，当且仅当

(i) 方程

$$a_{i_j} \geq a_j, \quad j = 1, \cdots, k, \tag{23}$$

有解$a_j, j = 1, \cdots, k$;

(ii) 如果$a_{i_j} = a_j$则有$i_j = j$.

Example 11.4.5

一个正规博弈$G = (N, S, C)$, 其中$N = \{1, 2\}$, $S_1 = \{1, 2\}$, $S_2 = \{1, 2, 3\}$, 其支付矩阵见表2.

表 2: 例11.4.5 支付矩阵

c\s	11	12	13	21	22	23
c_1	1.1	1.8	2.0	2.2	1.6	3.2
c_2	3.3	2.8	3.1	2.5	3.6	4.1

Example 11.4.5(cont'd)

使用MBRA, 我们有最佳响应函数如表3

表 3: 例11.4.5 最佳响应函数

f\s	11	12	13	21	22	23
f_1	2	1	2	2	1	2
f_2	1	1	1	3	3	3

Example 11.4.5(cont'd)

使用并联MBRA 则得

$$\begin{aligned} x_1(t+1) &= \delta_2[2,1,2,2,1,2]x(t) := M_1x(t) \\ x_2(t+1) &= \delta_3[1,1,1,3,3,3]x(t) := M_2x(t), \end{aligned}$$

于是可得局势演化方程:

$$\begin{aligned} x(t+1) &= Mx(t) = M_1 * M_2x(t) \\ &= \delta_6[4,1,4,6,3,6]x(t). \end{aligned} \tag{24}$$

Example 12.4.5(cont'd)

如果选

$$P(x) = \delta_6[2,1,4,5,3,6]x := V_p x. \tag{25}$$

则有

$$V := V_p(M - I_6) = [3,1,1,1,1,0] \geq 0.$$

最后, 可以验证$V_j = 0$ 则$j = 6$. 而当$j = 6$ 我们有$Col_j(M) = \delta_6^j$. 因此, $P(x)$ 是该系统的李雅普诺夫函数. 因平衡点唯一, 该演化博弈全局收敛.

作业

1. 两人玩石头 - 剪刀 - 布. 策略更新规则是: (i) 这次赢了, 下次就不动; (ii) 这次输了, 下次就取对方这次的策略; (iii) 这次平了, 下次就取能够胜这次策略的策略.试写出局势演化方程, 并分析其演化性质.

2. A, B, C 三人玩囚徒困境, 即三人同时每人各选一策略. 根据 $A-B$ 的局势判定 A 在与 B 的博弈中所得, 记为 $c_{A,B}$, 再根据 $A-C$ 的局势判定 A 在与 C 的博弈中所得, 记为 $c_{A,C}$, 最后, A 在本轮所得为 $c_A = c_{A,B} + c_{A,C}$. 类似地, 可以定义 c_B 和 c_C. 设支付双矩阵为表 4.
三人的策略更新规则如下:

- A: 无条件模仿.
- B, C: 短视最优响应.

(i) 试写出局势演化方程.

(ii) 讨论局势演化方程的收敛性.

表 4: 囚徒困境 (双矩阵)

$P_1 \backslash P_2$	1	2
1	$-1,\ -1$	$-10,\ 0$
2	$0,\ -10$	$-5,\ -5$

思考题: 考虑一个有限博弈$G = (N, S, C)$, 如果存在一个函数$P : S \to \mathbb{R}$, 满足对任意的个体$i \in N$，下面的式子成立

$$c_i(x_i, s_{-i}) - c_i(y_i, s_{-i}) = P(x_i, s_{-i}) - P(y_i, s_{-i}), \quad x_i, y_i \in S_i, s_{-i} \in S_{-i}, \tag{26}$$

那么G就被称为势博弈, P为该博弈的势函数. 如果势博弈按照确定性短视最优响应更新其策略, 讨论其势函数是否为该动态下的李雅普诺夫函数.

谢 谢 !

Q&A