DOI: 10.11992/tis.201706064
网络出版地址: http://kns.cnki.net/kcms/detail/23.1538.tp.20180423.1635.018.html

基于半张量积的企业创新网络演化博弈

武利琴[1],徐勇[1],王金环[1],李杰[2]
(1. 河北工业大学 理学院,天津 300401; 2. 河北工业大学 经济管理学院,天津 300401)

摘 要: 当代经济环境下,创新已经成为企业生存发展的必要条件。将所有企业按规模分为大小两种企业,建立企业创新双层耦合网络,并研究了企业间的博弈过程。首先,运用矩阵半张量积方法,以"智猪博弈"为基本博弈,得到每一时刻各企业的策略,而非企业总体创新的比例;其次,根据收益函数得到整个企业创新网络的最优稳定纳什均衡点;最后,增加政府调控,改变博弈基本支付矩阵,从而达到最优稳定纳什均衡状态,即所有企业全部创新。

关键词: 企业创新;半张量积;创新网络;演化博弈;纳什均衡;政府调控;智猪博弈;策略局势
中图分类号: F270; TP18 **文献标志码:** A **文章编号:** 1673−4785(2018)05−0776−07

中文引用格式: 武利琴,徐勇,王金环,等. 基于半张量积的企业创新网络演化博弈[J]. 智能系统学报, 2018, 13(5): 776–782.
英文引用格式: WU Liqin, XU Yong, WANG Jinhuan, et al. Evolutionary enterprise innovation networked game based on the semi-tensor product of matrices[J]. CAAI transactions on intelligent systems, 2018, 13(5): 776–782.

Evolutionary enterprise innovation networked game based on the semi-tensor product of matrices

WU Liqin[1], XU Yong[1], WANG Jinhuan[1], LI Jie[2]
(1. School of Sciences, Hebei University of Technology, Tianjin 300401, China; 2. School of Economics and Management, Hebei University of Technology, Tianjin 300401, China)

Abstract: In the contemporary economic environment, innovation has become the inevitable condition for the survival and development of enterprises. In this paper, all the enterprises were classified into two categories according to the scale, establishing a double-layer coupling network of enterprise innovation, and the process of the game between enterprises was studied. Firstly, taking "Boxed Pig Game" as the basic game, the semi-tensor product method was used to get the strategy of each enterprise at every moment, not the proportion of the enterprise overall innovation. Secondly, the optimal stability for the Nash equilibrium of the whole enterprise innovation network was reached according to the payoff function. Finally, the government regulation was introduced to change the basic matrix of the game and therefore reach a stable state of the optimal Nash equilibrium; that is, all enterprises realized innovation all round.
Keywords: enterprise innovation; semi-tensor product; innovation networks; evolutionary game; Nash equilibrium; government regulation; Boxed Pig game; strategy profile

网络上的演化博弈称为网络演化博弈,能够很好地刻画生物系统、物流系统、社会系统、多智能体系统的演化规律,引起了众多学者的广泛研究,如文献[1-4]就将网络演化博弈与各个系统紧密地联系在了一起。在一个网络演化博弈中,节点和边分别代表玩家和玩家之间的相互关系。在某些特定的策略调整规则下,玩家根据邻居上一时刻的策略和收益,不断更新自己下一时刻的策略。随着时间变化,整个网络演化局势和博弈动态都在不断变化。网络演化博弈理论有助于理解合作的涌现及演化规律,经常被用于研究经济管理方面的问题[5-6]。

收稿日期: 2017−06−19. **网络出版日期:** 2018−04−24.
基金项目: 国家社会科学基金项目 (16FGL014); 国家自然科学基金项目 (61203142); 河北省自然科学基金项目 (F2014202206).
通信作者: 徐勇. E-mail: xuyong@hebut.edu.cn.

当下经济发展背景下, 企业唯有不断创新才能长久生存发展下去。由于创新成本高额, 部分企业不愿意改变原有模式, 每个企业都想追求自己的收益最大, 然而对于政府或者整个社会来说, 所有企业全部创新, 才是最好的局势。当前已有一些对企业间创新博弈的研究方法, 如文献[5-8], 通过构建微分方程组, 利用给定的初始条件, 得到方程组的解。该解描述了企业的总体构成 P(创新比例) 随时间的演化趋势, 并通过数据仿真方法将博弈动态趋势以图的形态呈现[9], 然而并不能展示每一次博弈后整个企业网络的博弈动态。

本文通过构建精确的理论框架来分析和控制企业创新网络演化的博弈动态, 获得每次博弈后各企业的收益、策略等性质, 主要运用了矩阵半张量积这种新的方法。半张量最先由程代展教授提出, 成功应用于逻辑网络的分析与控制[10-13], 包括布尔网络稳定性、可控性、可观测性及最优控制[14-17]。半张量积在网络演化博弈动态行为及策略最优控制方面的研究也取得了显著成果[18-20], 并在智能电网、经济破产机制等实际问题中得到了广泛应用[21-22]。同时, 也可用于企业创新方面的研究。

在企业创新网络中, 本文将所有企业按规模分为大小两种企业, 建立企业创新双层耦合网络。网络中各节点不同时刻采取的策略不同, 会影响整个网络的演化趋势。因此可通过政府调控使得整个网络都达到创新的稳定局势, 并能永久保持下去。

1　预备知识

本节给出关于半张量积的常用符号、定义和基本性质。

1) $\boldsymbol{M}_{m\times n}$表示$m\times n$实矩阵的集合。

2) $\mathrm{Col}_i(\boldsymbol{M})$表示矩阵$\boldsymbol{M}$的第$i$列, $\mathrm{Col}(\boldsymbol{M})$表示矩阵$\boldsymbol{M}$的列集合。

3) $\mathcal{D}_k := \{1,2,\cdots,k\}$。

4) $\varDelta_n := \{\delta_n^i | i=1,2,\cdots,n\}$, 其中$\delta_n^i$为单位矩阵$\boldsymbol{I}_n$的第$i$列。

5) 矩阵$\boldsymbol{L} = [\delta_n^{i_1}\ \delta_n^{i_2}\ \cdots\ \delta_n^{i_t}]$为$n\times t$逻辑矩阵, 简写为$\boldsymbol{L} = \delta_n[i_1\ i_2\ \cdots\ i_t]$, 通常用$\mathcal{L}_{n\times t}$表示$n\times t$逻辑矩阵的集合。

6) $\boldsymbol{V}_r(\boldsymbol{A}) = [a_{1,1}\ a_{1,2}\cdots a_{1,n}\cdots a_{m,1}\ a_{m,2}\cdots a_{m,n}]^{\mathrm{T}}$表示矩阵 $\boldsymbol{A}$ 中行的展开。

定义 1[10]　设$\boldsymbol{A}\in\boldsymbol{M}_{m\times n}$, $\boldsymbol{B}\in\boldsymbol{M}_{p\times q}$, $l=\mathrm{lcm}\{n,p\}$为n与p的最小公倍数, 那么$\boldsymbol{A}$与$\boldsymbol{B}$的半张量积定义为

$$\boldsymbol{A}\ltimes\boldsymbol{B}\triangleq(\boldsymbol{A}\otimes\boldsymbol{I}_{l/n})(\boldsymbol{B}\otimes\boldsymbol{I}_{l/p})$$

半张量积是普通矩阵乘积的一般化, 因此通常省略半张量符号“$\ltimes$”。

定义 2[10]　设$\boldsymbol{A}\in\boldsymbol{M}_{p\times n}$, $\boldsymbol{B}\in\boldsymbol{M}_{q\times n}$。它们的 Khatri-Rao 积, 记$\boldsymbol{A}*\boldsymbol{B}$, 定义为

$$\boldsymbol{A}*\boldsymbol{B}=$$
$$[\mathrm{Col}_1(\boldsymbol{A})\otimes\mathrm{Col}_1(\boldsymbol{B})\ \mathrm{Col}_2(\boldsymbol{A})\otimes\mathrm{Col}_2(\boldsymbol{B})\ \cdots\ \mathrm{Col}_n(\boldsymbol{A})\otimes\mathrm{Col}_n(\boldsymbol{B})]\in\mathcal{M}_{pq\times n}$$

命题 1[11]　1) 设$\boldsymbol{X}\in\mathbf{R}^m$及$\boldsymbol{Y}\in\mathbf{R}^n$为两列向量, 则$\boldsymbol{W}_{[m,n]}\boldsymbol{XY}=\boldsymbol{YX}$, $\boldsymbol{W}_{[n,m]}\boldsymbol{YX}=\boldsymbol{XY}$, 其中$mn\times mn$维矩阵$\boldsymbol{W}_{[m,n]}$被称为换位矩阵, 且$\boldsymbol{W}_{[n,n]}:=W_{[n]}$;

2) 设$\boldsymbol{A}\in\boldsymbol{M}_{m\times n}$, 那么

$$\boldsymbol{W}_{[m,n]}\boldsymbol{V}_r(\boldsymbol{A})=\boldsymbol{V}_c(\boldsymbol{A}),\quad \boldsymbol{W}_{[n,m]}\boldsymbol{V}_c(\boldsymbol{A})=\boldsymbol{V}_r(\boldsymbol{A})$$

3) 设$\boldsymbol{X}\in\mathbf{R}^t$和$\boldsymbol{A}\in\mathbf{R}^{m\times n}$, 则$\boldsymbol{XA}=(\boldsymbol{I}_t\otimes\boldsymbol{A})\boldsymbol{X}$, 这称为伪交换性质。

引理 1[18]　假设$\boldsymbol{x}_i\in\boldsymbol{\varDelta}_k$, $i=1,2,\cdots,n$和$\boldsymbol{x}=\ltimes_{i=1}^n\boldsymbol{x}_i$, 则有$\boldsymbol{x}_i=\boldsymbol{\pi}_i^n\boldsymbol{x}$, 其中$\boldsymbol{\pi}_i^n=\boldsymbol{l}_{k^{i-1}}\otimes I_k\otimes\boldsymbol{l}_{k^{n-i}}$, $i=1,2,\cdots,n$, $\boldsymbol{\Pi}^n=[(\boldsymbol{\pi}_1^{n\mathrm{T}})\ (\boldsymbol{\pi}_2^n)^{\mathrm{T}}\cdots(\boldsymbol{\pi}_n^n)^{\mathrm{T}}]^{\mathrm{T}}$。

引理 2[11]　设$f:\mathcal{D}_k^n\to\mathbf{R}$是一伪逻辑函数, 则存在一个唯一的矩阵$\boldsymbol{M}_f\in\mathbf{R}^{1\times k^n}$, 称为$f$的结构矩阵, 满足

$$f(\boldsymbol{x}_1,\ \boldsymbol{x}_2,\ \cdots,\ \boldsymbol{x}_n)=\boldsymbol{M}_f\ltimes_{i=1}^n\boldsymbol{x}_i\tag{1}$$

式中: $\boldsymbol{x}_i\in\boldsymbol{\varDelta}_k$, $i=1,2,\cdots,n$,且$\boldsymbol{M}_f$列举了$\boldsymbol{x}=\ltimes_{i=1}^n\boldsymbol{x}_i=\boldsymbol{\delta}_{k^n}^1$到$\boldsymbol{x}=\boldsymbol{\delta}_{k^n}^{k^n}$过程中$f(\boldsymbol{x})$所有可能的值。引理 2 显示了怎样将一个逻辑函数表示成它的代数形式。

引理 3[11]　考虑一个k值逻辑动态网络:

$$\boldsymbol{x}(t+1)=\boldsymbol{Lx}(t)\tag{2}$$

式中: $\boldsymbol{x}(t)=\ltimes_{i=1}^n\boldsymbol{x}_i(t)$, $\boldsymbol{L}\in\mathcal{L}_{k^n\times k^n}$, 则

1) δ_k^i是结构矩阵 $\boldsymbol{L}$ 的稳定点, 当且仅当$\boldsymbol{L}$主对角线上的元素ℓ_{ii}等于 1。可得到式 (2) 中均衡点的数量, 用N_e表示, 有

$$N_e=\mathrm{Trace}(\boldsymbol{L})\tag{3}$$

2) 长度为s环的数量用C_s表示

$$\begin{cases} C_1=N_e \\ C_s=\dfrac{\mathrm{Trace}(L^s)-\displaystyle\sum_{k\in\rho(s)}kC_k}{s},\quad 2\leqslant s\leqslant k^n \end{cases}\tag{4}$$

式中: $\rho(s)$代表s真因子的集合, s的真因子是正整数$k<s$, 满足$\dfrac{s}{k}\in\mathbb{Z}_+$。

2　企业创新网络演化博弈

本文主要考虑企业群体之间的创新问题。企业创新成本投入和风险承担是必需的, 由于其规模不同, 导致资金周转与风险抵御能力的差距巨大, 因此将“智猪博弈”作为基本博弈是非常适合的。将参与博弈的企业群体按照规模分为两类: 大企业和小企业。将每个企业作为网络中参与博弈的玩家, 从而构建企业创新双层耦合网络, 上层表示大企业群体, 下层表示小企业群体。假设

双方创新成本为 2, 收益均为 10。博弈分为 4 种情况: 若大小企业都创新, 双方获得创新收益比为 7:3; 若仅大企业创新, 小企业可剽窃大企业的创新成果, 并抢先占领市场, 两者创新收益比为 6∶4; 若仅小企业进行创新, 大企业可凭借其规模效应获得更大的利润, 两者收益比为 9∶1; 若大小企业都不创新, 双方收益均为零。对应基本净收益矩阵:

$$\begin{bmatrix} x/y & s_1 & s_2 \\ s_1 & (5,1) & (4,4) \\ s_2 & (9,-1) & (0,0) \end{bmatrix} \tag{5}$$

式中: x表示大企业群体; y表示小企业群体; s_1表示创新策略; s_2表示不创新策略。企业拥有独立选择策略的权利, 因此可建立博弈模型来分析企业创新网络随时间演化的博弈动态过程。

2.1　企业创新网络博弈模型

对应上述双层网络演化博弈的过程构建模型。

1) 企业创新双层耦合网络:上层网络$G_1=(V_v,E_v)$, 其中$V_v=\{v_1,v_2,\cdots,v_{n_1}\}$是大企业玩家集, $E_v\subset V_v\times V_v$是大企业内部相互联系的边集。下层网络 $G_2=(V_w,E_w)$, 其中$V_w=\{w_1,w_2,\cdots,w_{n_2}\}$是小企业玩家集, $E_w\subset V_w\times V_w$是小企业内部相互联系的边集。$E_{vw}\subset V_v\times V_w$是网络$G_1$和$G_2$间相互联系的边集。故双层耦合网络由无向图$G=(V,E)$表示, 满足 $V=V_v\cup V_w=\{v_1,v_2,\cdots,v_{n_1},w_1,w_2,\cdots,w_{n_2}\}=\{v_1,v_2,\cdots,v_n\}$, $n=n_1+n_2$, $E=E_v\cup E_w\cup E_{vw}$。

2) 基本网络博弈: 由两个连通玩家 (不同层) 形成的基本博弈$\mathcal{G}$, 且策略集$S_1=S_2=\{s_1,s_2,\cdots,s_k\}$, 对应收益双矩阵:

$$\boldsymbol{M}=\begin{bmatrix} (c_1,\overline{c_1}) & (c_1,\overline{c_2}) & \cdots & (c_1,\overline{c_k}) \\ (c_2,\overline{c_1}) & (c_2,\overline{c_2}) & \cdots & (c_2,\overline{c_k}) \\ \vdots & \vdots & & \vdots \\ (c_k,\overline{c_1}) & (c_k,\overline{c_2}) & \cdots & (c_k,\overline{c_k}) \end{bmatrix} \tag{6}$$

令$\boldsymbol{M}_1$表示$\boldsymbol{M}$中每个数组的第一个元素所组成的矩阵, $\boldsymbol{M}_2$表示第二个元素所组成的矩阵。

3) 策略更新规则: 采用确定性无条件模仿策略更新规则。玩家i在$t+1$时刻的策略模仿它同层邻居$j\in\mathrm{SN}_i$在t时刻最优收益对应的策略, 设

$$\begin{aligned} j_v^* &= \mathrm{argmax}_{j\in\mathrm{SN}_i} p_j(x(t)) \\ j_w^* &= \mathrm{argmax}_{j\in\mathrm{SN}_i} p_j(y(t)) \end{aligned} \tag{7}$$

则

$$x_i(t+1)=x_{j_v^*}(t),\quad y_i(t+1)=y_{j_w^*}(t) \tag{8}$$

那么整个网络的策略更新表示为

$$(x_i(t+1),y_i(t+1))=f(x(t),y(t)) \tag{9}$$

式中: $x_i(t)$和$y_i(t)$是t时刻玩家v_i和w_i的策略, $x(t)=(x_1(t),x_2(t),\cdots,x_{n_1}(t))$, $y(t)=(y_1(t),y_2(t),\cdots,y_{n_2}(t))$。

4) 策略模仿优先权: 若 3) 内策略更新规则中被模仿的邻居玩家不唯一, 令

$$\begin{aligned} \mathrm{argmax}_{j\in\mathrm{SN}_i} p_j(x(t)) &= \{j_{v1}^*,j_{v2}^*,\cdots,j_{vr_1}^*\} \\ \mathrm{argmax}_{j\in\mathrm{SN}_i} p_j(y(t)) &= \{j_{w1}^*,j_{w2}^*,\cdots,j_{wr_2}^*\} \end{aligned}$$

则优先权如式 (10):

$$\begin{cases} j_v^*=\min\{\lambda|\lambda\in\mathrm{argmax}_{j\in\mathrm{SN}_i} p_j(x(t))\} \\ j_w^*=\min\{\mu|\mu\in\mathrm{argmax}_{j\in\mathrm{SN}_i} p_j(y(t))\} \end{cases} \tag{10}$$

$\mathcal{N}_i=\{v_j|(v_i,v_j)\in E\}=\mathrm{SN}_i\cup\mathrm{DN}_i$表示玩家$i$的所有邻居。$\mathrm{SN}_i=\{v_j|(v_i,v_j)\in E_v\cup E_w\}$表示与$i$同组的邻居, $\mathrm{DN}_i=\{v_j|(v_i,v_j)\in E_{vw}\}$表示与$i$不同组的邻居。由于大小企业之间存在合作竞争关系, 本文考虑博弈发生在不同组邻居DN_i中, 策略更新发生在同组邻居SN_i中。

2.2　博弈动态演化分析

网络博弈随着时刻的变化不断更新策略局势, 对应收益函数也发生变化, 从而形成动态网络演化博弈。根据网络的拓扑结构, 可得到企业创新双层耦合网络的邻接矩阵:

$$\boldsymbol{A}=\begin{bmatrix} (\boldsymbol{A}_v)_{n_1\times n_1} & (\boldsymbol{A}_{vw})_{n_1\times n_2} \\ (\boldsymbol{A}_{wv})_{n_2\times n_1} & (\boldsymbol{A}_w)_{n_2\times n_2} \end{bmatrix}$$

式中: $\boldsymbol{A}_v=\boldsymbol{A}_v^{\mathrm{T}}$, $\boldsymbol{A}_w=\boldsymbol{A}_w^{\mathrm{T}}$, $\boldsymbol{A}_{wv}=\boldsymbol{A}_{vw}^{\mathrm{T}}$。为了研究方便, 可将博弈过程进行代数公式化。不妨定义: 玩家v_i在时刻t的状态为$x_i(t)(1<i<n_1)$; 玩家w_i在时刻t的状态为$y_i(t)(1<i<n_2)$; 第j个策略用s_j表示, 且$s_j\sim\delta_k^j$。

玩家$v_i(1<i<n_1)$的收益函数表达式为

$$\begin{gathered} p_{v_i}(t)=\boldsymbol{V}_r^{\mathrm{T}}(\boldsymbol{M}_1)x_i(t)\sum_{j\in\mathrm{DN}_i}y_j(t)=\boldsymbol{V}_r^{\mathrm{T}}(\boldsymbol{M}_1)x_i(t)\mathrm{Row}_i(A_{vw})\cdot \\ \Pi^{n_2}y(t)=\boldsymbol{V}_r^{\mathrm{T}}(\boldsymbol{M}_1)(\boldsymbol{I}_k\otimes\mathrm{Row}_i(A_{vw})\pi^{n_2})\pi_i^{n_1}x(t)y(t):= \\ \boldsymbol{M}_{v_i}x(t)y(t) \end{gathered} \tag{11}$$

玩家$w_i(1<i<n_2)$的收益函数表达式为

$$\begin{gathered} p_{w_i}(t)=\boldsymbol{V}_r^{\mathrm{T}}(\boldsymbol{M}_2{}^{\mathrm{T}})y_i(t)\sum_{j\in\mathrm{DN}_i}x_j(t)=\boldsymbol{V}_r^{\mathrm{T}}(\boldsymbol{M}_2{}^{\mathrm{T}})\boldsymbol{W}_{[k,k]}\mathrm{Row}_i(A_{wv})\cdot \\ \Pi^{n_1}x(t)\pi_i^{n_2}y(t)=\boldsymbol{V}_r^{\mathrm{T}}(\boldsymbol{M}_2{}^{\mathrm{T}})\boldsymbol{W}_{[k,k]}\mathrm{Row}_i(A_{wv})\cdot \\ \Pi^{n_1}(\boldsymbol{I}_{k^{n_1}}\otimes\pi_i^{n_2})x(t)y(t):=\boldsymbol{M}_{w_i}x(t)y(t) \end{gathered} \tag{12}$$

式中: $x(t)=\ltimes_{i=1}^{n_1}x_i(t)$, $y(t)=\ltimes_{i=1}^{n_2}y_i(t)$。

基于上述双层耦合网络演化博弈过程, 考虑网络纯策略纳什均衡 (本文策略选择是确定型的, 因此在后文中全部简称为纳什均衡) 的存在性, 本文采用确定型策略更新规则。

定义 3[18]　对于一个博弈$\mathcal{G}$, 一个策略局势$x^*=(x_1^*,x_2^*,\cdots,x_n^*)\in S_1\times S_2\times\cdots\times S_n$是一个纳什均衡, 如果满足$p_i(x_i^*,x_{-i}^*)\geqslant p_i(x_i,x_{-i}^*)$对所有的$i\in N$, $x_i\in S_i$均成立, 其中$N=\{1,2,\cdots,n\}$是玩家集, S_i是第i个玩家的策略集, 且$x_{-i}^*=(x_1^*,x_2^*,\cdots,x_{i-1}^*,x_{i+1}^*,\cdots,x_n^*)$。

命题 2[18]　对于任给的$x,y\in\varDelta_k, x\neq y$, 必然存在一个整数$1\leqslant r\leqslant k-1$满足$x=\boldsymbol{M}_{o,k}^r y$, 其中$\boldsymbol{M}_{o,k}=\delta_k[2\ 3\ \cdots\ k\ 1]$是 k值逻辑算子$\varTheta_k$的结构矩阵, 且满足

$$\Theta_k\left(\frac{i}{k-1}\right)=\begin{cases}\dfrac{i-1}{k-1}, & i>0\\ 1, & i=0\end{cases}$$

引理 4[18]　对于双层网络演化博弈 $G=(\mathcal{G},S_1,S_2)$, 支付函数满足式 (11)、(12), 则G存在一个纳什均衡, 当且仅当存在一个整数$1\leqslant j\leqslant 2^n$, 满足$\mathrm{Col}(\boldsymbol{M}_p)\geqslant 0$, 对应$\{\delta_{2^n}^j|\mathrm{Col}_j(\boldsymbol{M}_p)\geqslant 0,1\leqslant j\leqslant 2^n\}$是所有纳什均衡的集合, 其中$\boldsymbol{M}_p=[\boldsymbol{M}_{v_1}^{\mathrm{T}}\ \boldsymbol{M}_{v_2}^{\mathrm{T}}\cdots\boldsymbol{M}_{v_{n_1}}^{\mathrm{T}}\ \boldsymbol{M}_{w_1}^{\mathrm{T}}\cdots\boldsymbol{M}_{w_{n_2}}^{\mathrm{T}}]^{\mathrm{T}}$。

对于$1\leqslant r\leqslant k-1$,

$$\boldsymbol{M}_{v_i,r}=\boldsymbol{M}_{v_i}[I_{k^{i-1}}\otimes(\boldsymbol{I}_k-\boldsymbol{M}_{o,k}^r)]$$
$$\boldsymbol{M}_{v_i}^{\mathrm{T}}=[\boldsymbol{M}_{v_i,1}^{\mathrm{T}}\ \boldsymbol{M}_{v_i,2}^{\mathrm{T}}\cdots\boldsymbol{M}_{v_i,k-1}^{\mathrm{T}}]$$

对于$1\leqslant\gamma\leqslant k-1$,

$$\boldsymbol{M}_{w_i,\gamma}=\boldsymbol{M}_{w_i}[\boldsymbol{I}_{k^{n_1+i-1}}\otimes(\boldsymbol{I}_k-\boldsymbol{M}_{o,k}^{\gamma})]$$
$$\boldsymbol{M}_{w_i}^{\mathrm{T}}=[\boldsymbol{M}_{w_i,1}^{\mathrm{T}}\ \boldsymbol{M}_{w_i,2}^{\mathrm{T}}\cdots\boldsymbol{M}_{w_i,k-1}^{\mathrm{T}}]$$

引理 4 只考虑了某一时刻该网络博弈的纳什均衡, 而无法判断随着演化某些不稳定的局势逐渐演化为稳定的纳什均衡局势。因此, 计算整个网络演化的转移矩阵是非常有必要的。

由引理 2 和支付函数表达式 (11)、(12),可得到整个网络演化博弈的转移矩阵, 步骤如下:

1) 重写式 (11)、(12) 为

$$p_{v_i}(x_i(t),y(t))=\boldsymbol{M}'_{v_i}y(t)x_i(t)$$
$$p_{w_i}(x(t),y_i(t))=\boldsymbol{M}'_{w_i}y(t)x_i(t)$$

2) 对于任意的策略x、y, 大企业玩家的最优反应策略为$\mathrm{BR}_{v_i}=\boldsymbol{L}'_{v_i}y,\ 1\leqslant i\leqslant n_1$,

$$l_{j,v_i}=\min\{l|\mathrm{Col}_l(\mathrm{Blk}_j(\boldsymbol{M}'_{v_i})),\ 1\leqslant l\leqslant k\},$$
$$\mathrm{Col}_j(\boldsymbol{L}'_{v_i})=\delta_k^{l_{j,v_i}}$$

小企业玩家的最优反应策略为

$$\mathrm{BR}_{w_i}=\boldsymbol{L}'_{w_i}y,\quad 1\leqslant i\leqslant n_2$$
$$l_{j,w_i}=\min\{l|\mathrm{Col}_l(\mathrm{Blk}_j(\boldsymbol{M}'_{w_i})),1\leqslant l\leqslant k\}$$
$$\mathrm{Col}_j(\boldsymbol{L}'_{w_i})=\delta_k^{l_{j,w_i}}$$

3) 根据优先权式 (10) 模仿同组邻居收益最大玩家的策略, 有如下形式:

$$x_i(t+1)=\boldsymbol{L}_{v_i}y(t)$$
$$\overline{\boldsymbol{M}}_{v_i}=\mathrm{Row}_i(\boldsymbol{A}_v+\boldsymbol{I}_{n_1})\left[\boldsymbol{L}'_{v_1}\ \boldsymbol{L}'_{v_2}\ \cdots\ \boldsymbol{L}'_{v_{n_1}}\right]^{\mathrm{T}}$$
$$\alpha_{s,v_i}=\min\{\alpha|\mathrm{Row}_\alpha(\overline{M}_{v_i})_{\alpha s}\},\ \mathrm{Col}_s(\boldsymbol{L}_{v_i})=\delta_k^{\alpha_{s,v_i}}。$$
$$y_i(t+1)=\boldsymbol{L}_{w_i}x(t)$$
$$\overline{\boldsymbol{M}}_{w_i}=\mathrm{Row}_i(\boldsymbol{A}_w+\boldsymbol{I}_{n_2})\left[\boldsymbol{L}'_{w_1}\ \boldsymbol{L}'_{w_2}\ \cdots\ \boldsymbol{L}'_{w_{n_2}}\right]^{\mathrm{T}}$$
$$\beta_{s,w_i}=\min\{\beta|\mathrm{Row}_\beta(\overline{\boldsymbol{M}}_{w_i})_{\beta s}\},\ \mathrm{Col}_s(\boldsymbol{L}_{w_i})=\delta_k^{\beta_{s,w_i}}。$$

于是 $x(t+1)=\boldsymbol{L}_vy(t),\ y(t+1)=\boldsymbol{L}_wx(t)$, 其中 $\boldsymbol{L}_v=\boldsymbol{L}_{v_1}*\boldsymbol{L}_{v_2}*\cdots*\boldsymbol{L}_{v_{n_1}},\ \boldsymbol{L}_w=\boldsymbol{L}_{w_1}*\boldsymbol{L}_{w_2}*\cdots*\boldsymbol{L}_{w_{n_2}}$。

4) 将 3) 中不同层的策略相乘, 有

$$x(t+1)y(t+1)=\boldsymbol{L}_vy(t)\boldsymbol{L}_wx(t)=\boldsymbol{L}_v(\boldsymbol{I}_{k^{n_2}}\otimes\boldsymbol{L}_w)y(t)x(t)=\boldsymbol{L}_v(\boldsymbol{I}_{k^{n_2}}\otimes\boldsymbol{L}_w)\boldsymbol{W}_{[k^{n_1},k^{n_2}]}x(t)y(t):=Lx(t)y(t)\tag{13}$$

式中$\boldsymbol{L}=\boldsymbol{L}_v(\boldsymbol{I}_{k^{n_2}}\otimes\boldsymbol{L}_w)\boldsymbol{W}_{[k^{n_1},k^{n_2}]}$为双层耦合网络演化的结构矩阵, 可由此来分析博弈的演化动态。

通过半张量积方法, 由博弈动态方程式 (13) 可得到博弈结构矩阵$\boldsymbol{L}$。分析结构矩阵可得到博弈的稳定点、极限环, 因此有定理 1。

定理 1　若存在整数$1\leqslant j\leqslant 2^n$, 同时满足以下两个条件:

1) $\mathrm{Col}_j(\boldsymbol{M}_p)\geqslant 0$;

2) $j\in N_e=\mathrm{Trace}(\boldsymbol{L}),\ \boldsymbol{L}=\boldsymbol{L}_v(\boldsymbol{I}_{k^{n_2}}\otimes\boldsymbol{L}_w)\boldsymbol{W}_{[k^{n_1},k^{n_2}]}$;

则称$\delta_{2^n}^j$为全局稳定纳什均衡局势。

证明　根据引理 4 可知, 满足定理 1 中条件 1) 的所有$\delta_{2^n}^j$都是纳什均衡点, 但纳什均衡点不一定唯一, 也不一定是稳定点。根据引理 3 可知, 如果j满足定理 1 中条件 2), 则$\delta_{2^n}^j$对应策略局势为稳定点。因此如果j同时满足定理 1 中条件 1)、2),那么结论成立, $\delta_{2^n}^j$为全局稳定纯策略纳什均衡局势。证毕。

推论 1　设J为所有满足定理 1 的$j\ (1\leqslant j\leqslant 2^n)$的集合, 令$h\in J$满足:

$$\begin{cases}\mathrm{Col}_h(\boldsymbol{M}_{v_i})\geqslant\mathrm{Col}_j(\boldsymbol{M}_{v_i})\\ \mathrm{Col}_h(\boldsymbol{M}_{w_i})\geqslant\mathrm{Col}_j(\boldsymbol{M}_{w_i})\end{cases},\quad h,j\in J\tag{14}$$

则$\delta_{2^n}^h$称为为全局稳定最优纳什均衡局势。

证明　由定理 1 可知, 所有J中元素对应的$\delta_{2^n}^j$都是稳定的纳什均衡点, 如果$h\in J$满足式 (14), 可知$\delta_{2^n}^h$是使各玩家收益总和最大的稳定纳什均衡点, 则称为最优稳定纳什均衡点。证毕。

2.3　政府调控下的演化博弈

本部分通过政府调控改变基本博弈矩阵, 从而使整个网络的演化达到理想的局势状态, 即大小企业全部创新, 且为稳定或者最优稳定纳什均衡状态。大小企业选择创新, 由于投资成本过高, 且收益相对减少, 短时不能获取到创新带来的收益, 因此政府通过调控诱发企业选择创新策略是非常必要的。控制设计如下。

政府的对企业创新的直接补贴为a, 对只想搭便车的不创新企业进行惩罚, 惩罚力度设为b, 对应双收益矩阵为

$$\begin{bmatrix} x/y & s_1 & s_2\\ s_1 & (5+a,1+a) & (4+a,4-b)\\ s_2 & (9-b,-1+a) & (-b,-b)\end{bmatrix}\tag{15}$$

上述控制表示政府对企业的补贴与惩罚是同时进行的。设计控制的宗旨是政府对企业创新只起调控作用, 尽可能少地进行投资或者获利。因此可假设补贴力度与惩罚力度相同, 即$a=b\ (a>0,b>0)$。

假设没有$a=b$这一条件, 则$a=0$, 表示企业只对不创新企业进行惩罚; 若$b=0$, 表示企业只对创新企业进行补贴。

通过仿真寻找恰当的a、b值，使得全局稳定点刚好为最优纯策略纳什均衡点，且大小企业全部创新，政府获利与投资之差尽可能小。当然，补贴或者惩罚力度不是盲目的，为了实现大小企业全部创新的目标，补贴力度$a=a_1+a_2$,惩罚力度$b=b_1+b_2$，需要同时满足方程组式 (16):

$$\begin{cases}(5+a_1)+(1+a_1)\geqslant(4+a_1)+(4-b_1)\\(5+a_2)+(1+a_2)\geqslant(9-b_2)+(-1+a_2)\\a_1=b_1\\a_2=b_2\end{cases}\tag{16}$$

于是，可得$a=b=2$，即补贴和惩罚力度均为 2。

博弈矩阵改变后，重新计算式 (13)，可得博弈的转移矩阵$\boldsymbol{L}$，通过定理 1 和推论 1 分析博弈的演化性质。若达不到所有企业全部创新的理想状态，则调整控制力度，直到大小企业玩家全部选择创新策略s_1。

上述调控借鉴了“智猪博弈”中防止小猪搭便车，促使大猪小猪都行动的思想，即政府通过同时对创新企业进行补贴，对不创新企业进行惩罚，从而干预企业的创新决策，促使大小企业全部创新。

3　举例分析

本节主要通过举例，研究增加政府调控后整个企业创新网络的演化动态。

为了计算简便，以 3 个大企业和 2 个小企业组成的企业创新网络为例，分析演化博弈过程。网络拓扑结构图如图 1 所示，其中第一层为大企业，$n_1=\{1,2,3\}$；第二层为小企业，$n_2=\{4,5\}$。

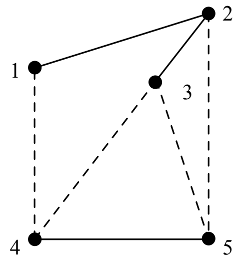

图 1　不同规模企业博弈结构图

Fig. 1　The game structure graph of different scale enterprises

根据网络拓扑结构图，我们可得到网络的邻接矩阵:

$$\boldsymbol{A}_v=\begin{bmatrix}0&1&0\\1&0&1\\0&1&0\end{bmatrix}_{3\times3},\boldsymbol{A}_w=\begin{bmatrix}0&1\\1&0\end{bmatrix}_{2\times2},\boldsymbol{A}_{vw}=\begin{bmatrix}1&0\\0&1\\1&1\end{bmatrix}_{3\times2}$$

情形 1　政府不进行干预 ($a=b=c=d=0$)，企业根据市场调节不断演化。收益矩阵为

$$\boldsymbol{M}=\begin{bmatrix}(5,1)&(4,4)\\(9,-1)&(0,0)\end{bmatrix},\boldsymbol{M}_1=\begin{bmatrix}5&4\\9&0\end{bmatrix},\boldsymbol{M}_2=\begin{bmatrix}1&4\\-1&0\end{bmatrix}$$

首先，根据引理 4 中的计算，有

$$\boldsymbol{M}_p=\begin{bmatrix}-4&-4&4&4&-4&-4&4&4&-4&-4&4&4&-4&-4&4&4&4&4&-4&-4&4&4&-4&-4&4&4&-4&-4&4&4&-4&-4\\-4&4&-4&4&-4&4&-4&4&4&-4&4&-4&4&-4&4&-4&-4&4&-4&4&-4&4&-4&4&4&-4&4&-4&4&-4&4&-4\\-8&0&0&8&8&0&0&-8&-8&0&0&8&8&0&0&-8&-8&0&0&8&8&0&0&-8&-8&0&0&8&8&0&0&-8\\-6&-6&6&6&-4&-4&4&4&-6&-6&6&6&-4&-4&4&4&-4&-4&4&4&-2&-2&2&2&-4&-4&4&4&-2&-2&2&2\\-6&6&-6&6&-4&4&-4&4&-4&4&-4&4&-2&2&-2&2&-6&6&-6&6&-4&4&-4&4&-4&4&-4&4&-2&2&-2&2\end{bmatrix}_{5\times32}$$

此时$\mathrm{Col}_4(\boldsymbol{M}_p)>0$，可得策略局势$\delta_{32}^4=\delta_2^1\delta_2^1\delta_2^1\delta_2^2\delta_2^2$为整个网络博弈的纳什均衡状态。此时 5 个企业中，大企业全部创新，小企业全部不创新。

其次，由博弈动态方程

$$x(t+1)y(t+1)=\boldsymbol{L}x(t)y(t)$$

得到$\boldsymbol{L}$为

$$\boldsymbol{L}=\delta_{32}[4\ 4\ 4\ 4\ 4\ 4\ 4\ 4\ 4\ 4\ 4\ 4\ 4\ 4\ 4\ 4\ 4\ 4\ 4\ 4\ 4\ 4\ 4\ 4\ 4\ 4\ 4\ 4\ 4\ 4\ 4\ 4]$$

由引理 3 知，博弈的最终状态是稳定点，即策略局势$\delta_{32}^4=\delta_2^1\delta_2^1\delta_2^1\delta_2^2\delta_2^2$，没有出现极限环。根据定理 1 和推论 1 可知，局势δ_{32}^4为全局最优稳定纯策略纳什均衡点，但只有大企业创新。为了让所有的玩家都实现创新，且达到稳定状态，我们考虑以下情形。

情形 2　政府对创新企业补贴为a，对不创新企业惩罚为b。由式 (15)、(16) 可得临界值$a=b=2$。通过 MATLAB 仿真，当$a=b=2$时，通过计算$\boldsymbol{M}_p$可得$J=\{1,5,9,13,17,21,25,29\}$，$\boldsymbol{L}=\delta_{32}[1\ 1\ 1\ 1\cdots1\ 1\ 1\ 1]$。由定理 1 和推论 1可知，全局稳定纳什均衡点为$\delta_{32}^1=\delta_2^1\delta_2^1\delta_2^1\delta_2^1\delta_2^1$，即大小企业全部选择创新，且为最优稳定纳什均衡；若$a=b>2$，不妨设$a=b=2.25$，通过计算$\boldsymbol{M}_p$可得$J=\{1\}$，且$\boldsymbol{L}=\delta_{32}[1\ 1\ 1\ 1\cdots1\ 1\ 1\ 1]$，由推论 1 可得整个网络博弈的全局最优稳定纳什均衡点为$\delta_{32}^1=\delta_2^1\delta_2^1\delta_2^1\delta_2^1\delta_2^1$，满足控制大小企业全部创新的目标；当$a=b<2$时，由于政府补贴与惩罚力度不够，整个演化局势依然保持无政府干预状态。

比较上述两种情形可知，政府对企业的调控，有效遏制了小企业“搭便车”的行为，从而促使大小企业实现全部创新。

4　结束语

本文将所有企业按规模分为大小两种企业，

建立企业创新双层耦合网络。运用矩阵半张量积方法, 以“智猪博弈”为基本博弈, 得到每一时刻各企业的策略, 整个网络局势演化随时间改变; 根据收益函数得到整个网络的最优稳定纳什均衡点; 最后, 通过政府调控, 改变博弈的基本矩阵, 从而达到最优稳定纳什均衡状态, 即所有企业全部创新。

参考文献:

[1] WU Qingchu, LOU Yijun, ZHU Wenfang. Epidemic outbreak for an SIS model in multiplex networks with immunization[J]. Mathematical biosciences, 2016, 277: 38–46.

[2] WANG Danzhu, LANG M X, SUN Yan. Evolutionary game analysis of co-opetition relationship between regional logistics nodes[J]. Journal of applied research and technology, 2014, 12(2): 251–260.

[3] ETESAMI S R, BASAR T. Complexity of equilibrium in diffusion games on social networks[C]//Proceedings of 2014 American Control Conference. Portland, USA, 2014: 2065–2070.

[4] YAN Yongyi, CHEN Zengqiang, YUE Jumei. STP approach to controlliability of finite state machines[J]. Ifac-PapersonLine, 2015, 48(28): 138–143.

[5] 于斌斌, 余雷. 基于演化博弈的集群企业创新模式选择研究[J]. 科研管理, 2015, 36(4): 30–38.
YU Binbin, YU Lei. A study on cluster enterprise technology innovation selection based on the evolutionary game[J]. Science research management, 2015, 36(4): 30–38.

[6] ZHOU Qing, FANG Gang, WANG Dongpeng, et al. Research on the robust optimization of the enterprise’s decision on the investment to the collaborative innovation: under the risk constraints[J]. Chaos, solitons and fractals, 2016, 89: 284–289.

[7] 盛光华, 张志远. 补贴方式对创新模式选择影响的演化博弈研究[J]. 管理科学学报, 2015, 18(9): 34–45.
SHENG Guanghua, ZHANG Zhiyuan. Allowance method’s influence on the innovation model choice in evolutionary game[J]. Journal of management sciences in China, 2015, 18(9): 34–45.

[8] 王健, 赵凯. “智猪博弈”下的合作创新研究——基于非对称演化博弈的分析[J]. 科技与经济, 2016, 29(2): 21–25.
WANG Jian, ZHAO Kai. Cooperative innovation of enterprises under “Boxed pig Game”—a research based on asymmetric evolutionary game[J]. Science and technology and economy, 2016, 29(2): 21–25.

[9] LIU Mengmeng, MA Yinghong, LIU Zhiyuan, et al. An IUR evolutionary game model on the patent cooperate of Shandong China[J]. Physic A: statistical mechanics and its applications, 2017, 475: 11–23.

[10] CHENG Daizhan, XU Tingting, QI Hongsheng. Evolutionary stability strategy of networked evolutionary games[J]. IEEE transactions on neural networks and learning systems, 2014, 25(7): 1335–1345.

[11] CHENG Daizhan, HE Fenghua, QI Hongsheng, et al. Modeling, analysis and control of networked evolutionary games[J]. IEEE transactions on automatic control, 2015, 60(9): 2402–2415.

[12] CHENG Daizhan, QI Hongsheng, HE F, et al. Semi-tensor product approach to networked evolutionary games[J]. Control theory and technology, 2014, 12(2): 198–214.

[13] CHENG Daizhan, XU Tingting, HE Fenghua, et al. On dynamics and Nash equilibriums of networked games[J]. IEEE/CAA journal of automatica sinica, 2014, 1(1): 10–18.

[14] ZHAO Yin, GHOSH B K, CHENG Daizhan. Control of large-scale Boolean networks via network aggregation[J]. IEEE transactions on neural networks and learning systems, 2016, 27(7): 1527–1536.

[15] LI Haitao, WANG Yuzhen, XIE Lihua. Output tracking control of Boolean control networks via state feedback: constant reference signal case[J]. Automatica, 2015, 59: 54–59.

[16] LI Haitao, XIE Lihua, WANG Yuzhen. On robust control invariance of Boolean control networks[J]. Automatica, 2016, 68: 392–396.

[17] CHENG Daizhan, ZHAO Yin, XU Tingting. Receding horizon based feedback optimization for mix-valued logical networks[J]. IEEE transactions on automatic control, 2015, 60(12): 3362–3366.

[18] GUO Peilian, WANG Yuzhen, JIANG Ping. Nash equilibrium, dynamics and control of evolutionary networked games with multi-group[C]//Proceedings of the 35th Chinese Control Conference. Chengdu, China, 2016: 585–590.

[19] GUO Peilian, WANG Yuzhen, LI Haitao. Algebraic formulation and strategy optimization for a class of evolutionary networked games via semi-tensor product method[J]. Automatic, 2013, 49(11): 3384–3389.

[20] ZHAO Guodong, WANG Yuzhen. Formulation and optimization control of a class of networked evolutionary games with switched topologies[J]. Nonlinear analysis: hybrid systems, 2016, 22: 98–107.

[21] ZHU Bing, XIA Xiaohua, WU Zhou. Evolutionary game theoretic demand-side management and control for a class

of networked smart grid[J]. Automatic, 2016, 70: 94–100.

[22] FU Shihua, WANG Yuzhen, ZHAO Guodong. A matrix approach to the analysis and control of networked evolutionary games with bankruptcy mechanism[J]. Asian journal of control, 2017, 19(2): 717–727.

作者简介：

武利琴，女，1992 年生，硕士研究生，主要研究方向为网络演化博弈及应用。

徐勇，男，1971 年生，教授，博士，主要研究方向为非线性系统控制、图论、复杂网络控制与优化。参加或主持省部级科研项目 10 余项，发表学术论文 30 余篇，被 EI 检索 10 余篇。

王金环，女，1980 年生，副教授，博士，主要研究方向为多智能体系统协同控制、网络演化博弈。主持国家级、省部级科研项目 4 项，发表学术论文 30 余篇，被 SCI 和 EI 收录 30 余篇。

2018 年人工智能与云计算大会（AICCC 2018）
2018 Artificial Intelligence and Cloud Computing Conference (AICCC 2018)

2018 Artificial Intelligence and Cloud Computing Conference (AICCC 2018) will be held during 21–23 December, 2018 in Tokyo, Japan. This conference is meant for researchers from academia, industries and research & development organizations all over the globe interested in the areas of Artificial Intelligence and Cloud Computing. It will put special emphasis on the participations of PhD students, Postdoctoral fellows and other young researchers from all over the world. It would be beneficial to bring together a group of experts from diverse fields to discuss recent progress and to share ideas on open questions. The conference will feature world-class keynote speakers in the main areas.

While artificial intelligence (A.I.) has struggled to gain footholds in other niches, it is finding its place in the world of cloud computing, a sort of revolution within the revolution that could rapidly change the face of businesses using cloud computing solutions over the next few years. What Is A.I.? First contemplated and theorized in the 1950s, A.I. generally refers to the ability of machines to perform intellectual tasks. It has branched into several different sub-levels since then, including machine learning, which, beginning around 1980, enabled computers to learn and build models, deep learning, which uses neural networks, and cognitive computing, a product of the last decade or so, which can allow machines to interact naturally with us. Machine learning (ML) and deep learning (DL) are the branches of A.I. most commonly linked to cloud computing. They can be harnessed for unsupervised work in analytics, predictions, and data mining-three absolutely huge parts of thousands of applications used by businesses in the cloud every single day. Deep learning takes machine learning a step forward. Instead of using a singular algorithm to crunch data, it uses a legion of them, all related, to create new, deep networks without human oversight. Deep learning has been successful in image recognition, facial recognition, even prediction of diseases based on a patient’s electronic health records. In three areas of cloud computing, A.I. is taking long strides. Those areas are ML algorithms, Big Data and parallel processing.