基于免疫粒子群算法的广义 Nash 均衡问题求解*

贾文生[a,b]，向淑文[a,b†]，杨剑锋[b]，何基好[a]
（贵州大学 a. 理学院；b. 计算机科学学院，贵阳 550025）

摘　要：针对广义 Nash 均衡求解问题，提出了一种免疫粒子群算法。首先利用非线性互补问题，将广义 Nash 均衡问题转换为非线性方程组问题，然后把免疫算法中抗体的免疫记忆功能和抗体浓度抑制机制引入基本粒子群算法，设计了一种免疫粒子群算法。最后通过数值实验表明，该算法保持了粒子群种群多样性，增强了粒子群算法的全局寻优能力，加快了算法的收敛速度，具有较好的性能。

关键词：免疫算法；粒子群算法；广义 Nash 均衡；非线性互补问题
中图分类号：TP301；O225　　**文献标志码**：A　　**文章编号**：1001-3695(2013)09-2637-04
doi：10.3969/j.issn.1001-3695.2013.09.020

Solving generalized Nash equilibrium problem based on immune particle swarm algorithm

JIA Wen-sheng[a,b], XIANG Shu-wen[a,b†], YANG Jian-feng[b], HE Ji-hao[a]
(*a. College of Science, b. College of Computer Science, Guizhou University, Guiyang* 550025, *China*)

Abstract: This paper presented an immune particle swarm algorithm for solving generalized Nash equilibrium problem. First it reformulated the generalized Nash equilibrium problem as the nonlinear equations problem by the nonlinear complementarity problem. Then it designed an immune particle swarm algorithm by involving the immune memory function and the antibody concentration inhibition mechanism into the original swarm algorithm. Finally the computer simulation results demonstrate that the proposed algorithm is effective and it not only keeps the variety of the original swarm, but also improves the abilities of seeking the global optimization result and the evolution speed of convergence.

Key words: immune algorithm; particle swarm algorithm; generalized Nash equilibrium; nonlinear complementarity problem

0　引言

广义 Nash 均衡问题由诺贝尔奖得主 Debreu、Arrow 等人[1,2]提出，近年来受到了越来越多学者的关注，产生了大量研究成果，已广泛应用于经济均衡、电力市场、环境污染治理、计算机和交通网络等领域[3~7]。但遗憾的是 Arrow、Debreu 并没有给出一个求解广义 Nash 均衡的算法，广义 Nash 均衡算法的设计和求解至今仍没有一套相对成熟的理论和方法，仍是学者关注的热点。对广义 Nash 均衡的求解，传统的数学分析方法大体上有两种做法，一种思路是将广义 Nash 均衡问题转换为拟变分不等式问题设计求解算法[8]，另一思路是将广义 Nash 均衡问题利用 *N-I* 函数转换为带约束或无约束的最优化问题设计牛顿类型的求解算法[9]。但是传统的数学分析算法往往要求博弈局中人的支付函数和策略集具有较高的连续性和凸(凹)性，并且算法本身往往依赖于初始点的选取，在一定程度上限制了该类算法的应用。Roughgarden[10]指出 Nash 均衡的求解是一个 NP-hard 问题，随着博弈规模的不断增大，传统的数学分析算法面临着计算复杂度高和计算时间长等问题。随着现代智能算法的不断深入研究和发展，智能算法在解决 NP 难问题上体现了较强的优越性。因此，本文尝试借鉴生物进化理论和生物行为规律的免疫粒子群算法来计算和模拟广义 Nash 均衡的求解，是一种新的途径和方法。通过实例的计算和比较，表明本文提出的免疫粒子群智能算法具有较好的性能。

1　广义 Nash 均衡的描述

对于一个广义博弈问题，假设有 N 个局中人，第 $v(v=1,2,\cdots,N)$ 个局中人的决策变量为 $x^v \in R^{n_v}$，且将所有局中人的决策变量所组成的向量记为 $\boldsymbol{x}=(x^1,x^2,\cdots,x^N)^{\mathrm{T}} \in R^n$，其中 $n=\sum_{i=1}^{N} n_i$。为强调向量 $\boldsymbol{x}$ 中的第 v 个变量，记 $\boldsymbol{x}=(x^v,x^{-v})^{\mathrm{T}}$，其中 x^{-v} 表示除局中人 v 之外的所有局中人所构成的决策变量。记第 v 个局中人的策略集为 $X^v \subset R^{n_v}$，所有局中人所构成的策略集记为 $X=\prod_{j\in N} X^v \subset R^n$，而 $X^{-v}=\prod_{j\in N, j\neq v} X^j$ 表示除局中人 v 之外的所有博弈参与者所构成的策略集。博弈中每个局中人 v 都选定自己的策略 x^v 极小化其支付函数 θ_v，即

$$\min_{x^v \in R^{n_v}} \theta_v(x^v,x^{-v}),\ \forall x^v \in K_v(x^{-v}) \tag{1}$$

收稿日期：2013-01-03；**修回日期**：2013-02-22　**基金项目**：国家自然科学基金资助项目(11161008)；国家留学基金资助项目(201201470039)；国家教育部博士点基金资助项目(20115201110002)；贵州省自然科学基金资助项目(黔科合 J 字[2012]2139 号)

作者简介：贾文生(1981-)，男，河南南阳人，博士研究生，主要研究方向为算法博弈论(jws0505@163.com)；向淑文(1964-)，男(通信作者)，教授，博导，主要研究方向为博弈论、非线性分析；杨剑锋(1986-)，男，博士研究生，主要研究方向为软件可靠性分析；何基好(1976-)，男，博士研究生，主要研究方向为算法博弈论.

其中,映射 $K_v: X^{-v} \to 2^{X^v}$ 表示受所有 $j \neq v$ 的局中人的决策影响的可行策略集值映射(2^{X^v} 表示 X^v 的所有非空子集的全体),则对任意的 $x^{-v} \in X^{-v}$,都有 $K_v(x^{-v}) \subseteq X^v$。若记 $S_v(x^{-v})$ 表示极小化问题式(1)的解集,则广义博弈 Nash 均衡问题为找一个向量 $\boldsymbol{x}^*$ 使得对所有的 v 都有 $x_v^* \in S_v(x^{-v*})$,称 $\boldsymbol{x}^*$ 为广义 Nash 均衡解。需要特别指出的是,若对所有的 $v = 1,2,\cdots,N$,任意的 $x^{-v} \in X^{-v}$,都有 $K_v(x^{-v}) = X^v$,则广义 Nash 均衡就变为一般 N 人非合作博弈的 Nash 均衡,因此本文设计的算法对一般 N 人非合作博弈的 Nash 均衡也是适用的。

在实际应用中,广义博弈问题的决策者可行域通常是由有限个约束定义的,其中每个决策者 v 的可行集 $K_v(x^{-v})$ 通常用下式表示:$K_v(x^{-v}) = \{x^v \in R^{n_v}: h^v(x^v) \leqslant 0, s(x) \leqslant 0\}$。其中 $h^v: R^{n_v} \to R^{m_v}, v = 1,2,\cdots,N, s: R^n \to R^{m_0}$,即 h^v 是所有决策变量 x^v 的约束,$s(x)$ 是所有决策者的共享约束。带共享约束的广义博弈问题是一类重要问题,具有广泛的应用背景,它描述的情形是在博弈过程中两个或多个局中人共享某些资源,例如文献[2]中抽象经济模型、文献[4]中电信系统中的频率分配模型和文献[6]中的环境污染控制模型等。

2 广义 Nash 均衡、非线性互补问题和非线性方程组的转换

2.1 广义 Nash 均衡转换为非线性互补问题

下面首先利用最优化问题中著名的 KKT 条件,将广义 Nash 均衡问题转换为非线性互补问题。为了保证广义 Nash 均衡解的存在性,首先作如下假设:

a)对所有 $v = 1,2,\cdots,N$,函数 $\theta_v: R^{n_v} \to R, h^v: R^{n_v} \to R^{m_v}, s: R^n \to R^{m_0}$ 都二次可微,并且具有局部 Lipschitz 连续的二阶导数。

b)对所有 $v = 1,2,\cdots,N$,给定任意的 x^{-v},目标函数 $\theta(\cdot, x^{-v})$ 关于 x^v 是凸的,约束函数 $h^v: R^{n_v} \to R^{m_v}, s: R^n \to R^{m_0}$ 的每个分量函数都是凸的。

设 $\boldsymbol{x}^*$ 是广义博弈问题的一个 Nash 均衡解,若对局中人 v 适当的约束规范成立,则存在乘子向量 $\lambda^{*,v} \in R^{m_v}, \mu^{*,v} \in R^{m_0}$ 使得 $(x^{*,v}, \lambda^{*,v}, \mu^{*,v})$ 满足 KKT 条件:

$$\nabla_{x^v} L_v(x^v, x^{*,-v}, \lambda^v, \mu^v) = 0$$
$$0 \leqslant \lambda^v \perp -h^v(x^v) \geqslant 0$$
$$0 \leqslant \mu^v \perp -s(x^v, x^{*,-v}) \geqslant 0$$

其中:$L_v(\boldsymbol{x}, \lambda^v, \mu^v) = \theta_v(\boldsymbol{x}) + h^v(x^v)^{\mathrm{T}} \lambda^v + s(\boldsymbol{x})^{\mathrm{T}} \mu^v$ 为第 v 个局中人极小化问题的 Lagrange 函数。综合上述 N 个 KKT 条件,由文献[11]可得:若 $\boldsymbol{x}^*$ 是广义博弈问题的一个 Nash 均衡解,且对所有局中人一个适当的约束规范成立,则存在乘子向量 $\lambda^* \in R^m, \mu^* \in R^{N_{m0}}$ 使得 $(\boldsymbol{x}^*, \lambda^*, \mu^*)$ 满足如下系统:

$$L(\boldsymbol{x}, \lambda, \mu) = 0$$
$$0 \leqslant \lambda \perp -h(\boldsymbol{x}) \geqslant 0$$
$$0 \leqslant \mu \perp -S(\boldsymbol{x}) \geqslant 0 \tag{2}$$

其中
$$\lambda = \begin{pmatrix} \lambda^1 \\ \lambda^2 \\ \vdots \\ \lambda^N \end{pmatrix}, \mu = \begin{pmatrix} \mu^1 \\ \mu^2 \\ \vdots \\ \mu^N \end{pmatrix}, h(\boldsymbol{x}) = \begin{pmatrix} h^1(x^1) \\ h^2(x^2) \\ \vdots \\ h^N(x^N) \end{pmatrix}$$

$$S(\boldsymbol{x}) = \begin{pmatrix} s(\boldsymbol{x}) \\ s(\boldsymbol{x}) \\ \vdots \\ s(\boldsymbol{x}) \end{pmatrix}, L(\boldsymbol{x}, \lambda, \mu) = \begin{pmatrix} \nabla_{x^1} L_1(\boldsymbol{x}, \lambda^1, \mu^1) \\ \nabla_{x^2} L_2(\boldsymbol{x}, \lambda^2, \mu^2) \\ \vdots \\ \nabla_{x^N} L_N(\boldsymbol{x}, \lambda^N, \mu^N) \end{pmatrix}$$

式(2)可以看成是广义博弈问题的一阶必要条件。若广义博弈问题满足假设 a)和 b),则对任意给定的 x^{-v},局中人 v 的极小化问题就是凸优化问题,从而由极小值原理易知系统式(2)的解必是广义 Nash 均衡的解,于是式(2)也是广义 Nash 均衡问题的一阶充分性条件,故有如下结果:

定理 设广义博弈问题满足假设 a)和 b),若 $(\boldsymbol{x}^*, \lambda^*, \mu^*)$ 满足式(2),则 $\boldsymbol{x}^*$ 为广义博弈 Nash 均衡点。

下面定义 $F: R^n \to R^n, F(\boldsymbol{x}) = \mathrm{vec}\{\nabla_{x^v} \theta_v(\boldsymbol{x})\}_{v=1}^N$,则式(2)等价地转换为如下系统:

$$F(\boldsymbol{x}) + \sum_{v=1}^N \nabla_x h^v(\boldsymbol{x}) \lambda^v + \nabla_x s(\boldsymbol{x}) \mu = 0$$
$$0 \leqslant \lambda^v \perp -h^v(x^v) \geqslant 0$$
$$0 \leqslant \mu \perp -S(\boldsymbol{x}) \geqslant 0 \tag{3}$$

式(3)是一个非线性互补问题,从而把一个广义 Nash 均衡问题利用凸优化问题的 KKT 条件转换成为一个非线性互补问题。

2.2 互补问题转换为非线性方程组问题

下面用补函数方法将非线性互补问题转换为一个非线性方程组问题。

函数 $\phi: R^2 \to R$ 称为一个补函数,若有 $\phi(a,b) = 0 \Rightarrow a \geqslant 0, b \geqslant 0, ab = 0$。目前已有很多种补函数,比较常用的是 Fisher-Burmeister(FB)函数,即 $\phi_{FB}(a,b) = a + b - \sqrt{a^2 + b^2}$。令

$$\Phi_{FB}(\boldsymbol{x}, \lambda, \mu) = \begin{pmatrix} L(\boldsymbol{x}, \lambda, \mu) \\ \phi_{FB}(-h_1^1(\boldsymbol{x}), \lambda_1^1) \\ \vdots \\ \phi_{FB}(-h_i^v(\boldsymbol{x}), \lambda_i^v) \\ \vdots \\ \phi_{FB}(-h_{m_N}^N(\boldsymbol{x}), \lambda_{m_N}^N) \\ \phi_{FB}(-s_1(\boldsymbol{x}), \mu_1) \\ \vdots \\ \phi_{FB}(-s_{m_0}(\boldsymbol{x}), \mu_{m_0}) \end{pmatrix}$$

其中 ϕ_{FB} 是 FB 函数,则求解式(3)等价地转换为求解 $\Phi_{FB}(\boldsymbol{x}, \lambda, \mu) = 0$,从而求解广义 Nash 均衡问题通过互补问题转换为求解非线性方程组问题。针对广义 Nash 均衡问题转换为非线性方程组问题,设计免疫粒子群算法求解。

3 免疫粒子群算法的设计

3.1 基本粒子群算法

基本粒子群算法最早由 Kennedy 和 Eberhart[12] 于 1995 年通过对鸟群觅食过程的分析和模拟而提出,由于该算法具有简单易行、收敛速度快和易于并行等特点在许多领域得到了应用和发展[13~17]。在应用粒子群算法求解优化问题时,每一个优化问题的潜在解都可以看成 d 维搜索空间中的一个粒子。设第 i 个粒子 x_i 的位置表示为 $x_i = (x_{i1}, x_{i2}, \cdots, x_{id})$,它经历过的最好位置,即有最好的适应度值时的位置,记为 $p_{\mathrm{best}}(i)$,$p_{\mathrm{best}}(i) = (p_{i1}, p_{i2}, \cdots, p_{id})$。在群体中所有粒子经历过的最好位置用 g_{best} 表示。设第 i 个粒子的速度为 $v_i = (v_{i1}, v_{i2}, \cdots, v_{id})$。对每一次迭代,粒子的速度和位置变化根据如下方程进行:

$$v_i^{k+1} = wv_i^k + c_1 r_1 (p_{\mathrm{best}}{}^k(i) - x_i^k) + c_2 r_2 (g_{\mathrm{best}}{}^k - x_i^k) \tag{4}$$

$$x_i^{k+1} = x_i^k + v_i^{k+1} \tag{5}$$

其中:w 为粒子惯性权重;c_1、c_2 为粒子学习因子,分别调节全

局最好粒子和个体最好粒子飞行方向的最大步长,通常取 $c_1 = c_2 = 2$;r_1、r_2 为在(0,1)内的随机数,对惯性权重一般按照文献[13]中的公式计算。

$$\omega = \omega_{\max} - k\frac{\omega_{\max} - \omega_{\min}}{k_{\max}} \tag{6}$$

其中:$\omega_{\max}$ 为最大惯性权重,$\omega_{\min}$ 为最小惯性权重,$k_{\max}$ 为最大迭代次数,k 为当前迭代次数。

3.2　免疫粒子群算法的基本思想

免疫粒子群算法是在基本粒子群算法的基础上引入生命科学中的免疫原理,将所要求解的问题视为抗原,每一个抗体代表问题的一个解,同时每一个抗体也代表基本粒子群中的一个粒子。抗原与抗体的亲和度由粒子群算法中的适应度函数值来衡量。在粒子(抗体)群体迭代更新的过程中,总是希望适应度好的粒子(抗体)保留下来,但如果此类粒子(抗体)过于集中,则很难保证粒子(抗体)的多样性,从而易使算法陷入局部最优解。因此,本文考虑把免疫记忆功能与抗体浓度的自我调节机制引入基本粒子群算法,借助粒子(抗体)浓度的选择公式来保持各适应度层次的粒子维持在一定的浓度,以保持粒子群种群的多样性,便于寻求全局最优解。

算法中每一个粒子由所有局中人的策略表示,即 $\boldsymbol{x} = (x^1, x^2, \cdots, x^N)$,定义免疫粒子群算法的适应度函数如下:

$$f(\boldsymbol{x}) = \| \Phi_{FB}(\boldsymbol{x}, \lambda, \mu) \|^2$$

显然,$f(\boldsymbol{x}) = 0 \Leftrightarrow \Phi_{FB}(\boldsymbol{x}, \lambda, \mu) = 0$,所以求方程组 $\Phi_{FB}(\boldsymbol{x}, \lambda, \mu) = 0$,就相当于在约束区域内求 $\min\limits_{x \in K(\boldsymbol{x})} f(\boldsymbol{x})$ 使得 $f(\boldsymbol{x})$ 的最小值充分接近于 0。

定义第 i 个粒子(抗体)的浓度如下[13]:

$$D(x_i) = \frac{1}{\sum\limits_{j=1}^{M+Q} |f(x_i) - f(x_j)|} \quad i = 1, 2, \cdots, M+Q$$

定义基于上述粒子(抗体)浓度的概率选择公式如下:

$$P(x_i) = \frac{\dfrac{1}{D(x_i)}}{\sum\limits_{i=1}^{M+Q} \dfrac{1}{D(x_i)}} = \frac{\sum\limits_{j=1}^{M+Q} |f(x_i) - f(x_j)|}{\sum\limits_{i=1}^{M+Q} \sum\limits_{j=1}^{M+Q} |f(x_i) - f(x_j)|} \tag{7}$$

其中,x_i 和 $f(x_i)$,$i = 1, \cdots, M+Q$ 分别表示第 i 个粒子(抗体)和适应度函数值。

3.3　免疫粒子群算法的实现步骤

根据上述免疫粒子群算法的基本思想,设计算法的实现步骤如下:

a)确定免疫粒子群算法的基本参数值,包括粒子群群体规模 M,粒子最大惯性权重 $\omega_{\max}$,最小惯性权重 $\omega_{\min}$,学习因子 c_1、c_2,最大迭代次数 $k_{\max}$,精度 ε。

b)随机生成 M 个初始粒子 x_i 和初始粒子 x_i 的速度 v_i,形成初始化粒子群 p_0。

c)计算每个粒子适应度函数值,找出粒子的个体极值 $p_{\text{best}}(i)$,$i = 1, \cdots, M$ 和全体极值 g_{best}。

d)按照式(6)计算粒子惯性权重 ω。

e)按照式(4)和(5)更新粒子的速度和位置,并将 g_{best} 对应的粒子位置存入记忆库。

f)依次检验第 i 个粒子的位置 x_i^{k+1},保证所有粒子在每一次迭代过程中都在其可行策略集的空间内。

g)随机生成 Q 个粒子,方法同 b)。

h)根据式(7),从 $M+Q$ 个粒子中依浓度的概率大小选取 M 个粒子。

i)以记忆库中的粒子代替新选取的 M 个粒子中适应度最差的粒子,生成一个新的粒子群 p_1,准备进入下一次迭代。

j)根据最大迭代次数和精度判断是否结束迭代,并输出符合条件的最优粒子(即所求问题的近似解)和迭代次数,否则转 c)。

3.4　算法性能评价

免疫粒子群算法作为一种生物演化智能算法,与遗传算法有很多相似之处,因此可以借鉴 Dejong[18] 给出的分析遗传算法性能而提出的算法评价方法,即离线性能来测试算法的收敛特性。

定义[18]　设 $X_e^*(s)$ 为环境 e 下策略 s 的离线性能,$f_e^*(t)$ 为第 t 代相应于环境 e 的最佳适应度,则有 $X_e^*(s) = \frac{1}{T}\sum\limits_{t=1}^{T} f_e^*(t)$,即离线性能表示算法到第 t 代的最佳适应度的平均值。

4　数值例子

首先考虑文献[19]中给出的广义 Nash 均衡算例。

例 1　设有两人广义博弈,局中人的决策变量分别为 x_1、x_2,且满足 $0 \leqslant x_1 \leqslant 10$,$0 \leqslant x_2 \leqslant 10$,$x_1 + x_2 \leqslant 15$。局中人的支付函数分别为 θ_1、θ_2,其中:

$$\theta_1(x_1, x_2) = x_1^2 + \frac{8}{3}x_1x_2 - 34x_1$$

$$\theta_2(x_1, x_2) = x_2^2 + \frac{5}{4}x_1x_2 - 24.5x_2$$

用本文给出的免疫粒子群算法求解(算法的参数设置为:粒子群群体规模 $M = 20$,$Q = 10$,学习因子 $c_1 = c_2 = 2$,最大迭代次数为 300,适应度函数精度设置为 $\varepsilon = 10^{-2}$),五次计算结果和离线性能分别如表 1 和图 1 所示。

表 1　免疫粒子群算法的计算结果

计算次数	Nash 均衡	迭代次数	适应度函数值
1	$x_1 = 4.9419$, $x_2 = 9.0415$	112	0.0059
2	$x_1 = 5.0416$, $x_2 = 8.9744$	114	0.0049
3	$x_1 = 5.0102$, $x_2 = 8.9865$	138	0.0064
4	$x_1 = 4.9933$, $x_2 = 8.9988$	88	0.0099
5	$x_1 = 4.9378$, $x_2 = 9.0482$	102	0.0078

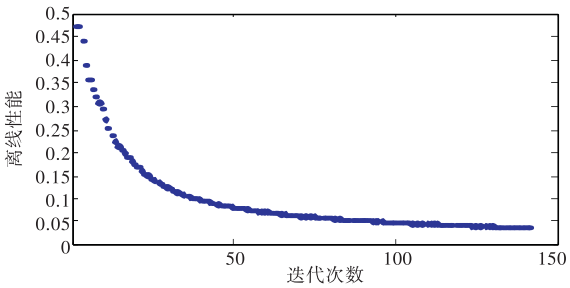

图1　免疫粒子群算法的离线性能

由计算结果可知,本文五次计算结果平均迭代 110 次即可给出例 1 的近似广义 Nash 均衡点(5,9)。与文献[19]用数学分析的投影算法相比,不但迭代次数大大减少,而且每次计算初始点是随机生成的,计算过程不依赖初始点的选取,而文献[19]中的算法当初始点选取为(10,0)时,计算不出 Nash 均衡点。另外,算法的离线性能也反映了该算法具有较好的收敛性。

例 2　考虑由 Kesselman 等人[20]提出的网络交换模型。

在网络交换中,每个用户单独产生流量,遵循先进先出原

则,并且缓冲器的容量是有限的,每个用户的效用取决于传输效率和网络拥塞程度。确切地说,假设有 N 个用户,缓冲器容量为 B,每个用户 v 决定了它在该缓冲器上的数据包数量,记做 $x^v \in [0, +\infty)$,并假设缓冲器遵循去尾原则,即若缓冲器饱和,则数据包释放并重新给出。用户 v 的效用函数如下:

$$\theta_v(\boldsymbol{x}) = \begin{cases} \dfrac{x^v}{\sum\limits_{v=1}^{N} x^v}\left(1 - \dfrac{\sum\limits_{v=1}^{N} x^v}{B}\right) & \sum\limits_{v=1}^{N} x^v > 0 \\ 0 & \sum\limits_{v=1}^{N} x^v = 0 \end{cases}$$

求效用函数的最大化,$\dfrac{x^v}{\sum\limits_{v=1}^{N} x^v}$表示用户 v 的传输率,用户的效用随着传输率的增大而增大,$\dfrac{\sum\limits_{v=1}^{N} x^v}{B}$表示缓冲器的拥塞水平。注意到如果缓冲器饱和拥塞则每个用户的效用为 0,由于遵循去尾原则,第 v 个用户的优化问题为

$$\min_{x^v} \{ -\theta_v(\boldsymbol{x}) \}$$

$$\text{s.t.} \quad x^v \geqslant 0, \sum_{v=1}^{N} x^v \leqslant B$$

则 N 个用户的极小化问题构成了一个广义博弈问题。

为了保证效用函数在整个可行域上有定义,本文采用修正后的约束形式,即 $x^v \geqslant 0.001, v = 1, \cdots, N$。采用本文的方法可以将上述广义博弈问题转换为如下非线性方程组问题:

$$\Phi_{FB}(\boldsymbol{x}, \lambda, \mu) = \begin{pmatrix} \dfrac{1}{B} - \dfrac{\sum\limits_{v=1}^{N} x^v - x^1}{(\sum\limits_{v=1}^{N} x^v)^2} - \lambda^1 + \mu \\ \dfrac{1}{B} - \dfrac{\sum\limits_{v=1}^{N} x^v - x^N}{(\sum\limits_{v=1}^{N} x^v)^2} - \lambda^N + \mu \\ \phi_{FB}(-(-0.001 - x^1), \lambda^1) \\ \vdots \\ \phi_{FB}(-(-0.001 - x^N), \lambda^N) \\ \phi_{FB}(-(\sum\limits_{v=1}^{N} x^v - B), \mu) \end{pmatrix} = 0$$

在数值实验中,本文分别取用户数 $N = 10, 40, 80, 100, 1000$ 进行计算(算法的参数设置为:粒子群群体规模 $M = 20$, $Q = 10$,学习因子 $c_1 = c_2 = 2$,最大迭代次数为 300,适应度函数精度设置为 $\varepsilon = 10^{-2}$,解的误差设置为 $\| x_k^v - x_{k-1}^v \| \leqslant 10^{-6}$),计算结果如表 2 所示。

表 2 免疫粒子群算法的计算结果

用户数 N	迭代次数	Nash 均衡	适应度函数值
10	2	$x^v = 0.982\ 772\ 9$	0.052 9
40	2	$x^v = 0.248\ 402\ 9$	0.012 0
80	2	$x^v = 0.124\ 139\ 1$	0.046 7
100	2	$x^v = 0.009\ 275\ 9$	0.097 0
1000	3	$x^v = 0.001\ 000\ 0$	0.002 1

针对文献[20]中给出的上述广义 Nash 均衡模型,文献[11]给出了一个很好的光滑牛顿算法并计算了用户数 $N = 10$, 40,80,100 的结果。在同样的精度下,本文给出的算法仅 2 次迭代就可以求得近似 Nash 均衡解,当用户数为 1000 时,迭代次数也仅需 3 次,进一步显示了免疫粒子群算法作为智能算法解决大规模广义 Nash 均衡问题的优越性。

5 结束语

总之,通过例 1 和 2 数值算例的比较和分析可以看出,本文提出的免疫粒子群算法求解广义 Nash 均衡问题是有效的。该算法把免疫记忆机制和基于抗体浓度的选择机制引入基本粒子群算法,从而保持了粒子群种群在进化迭代过程中的多样性,综合了免疫算法和基本粒子群算法的优点,不仅保持了粒子群算法简单、易于实现、收敛速度快的特点,而且增强了基本粒子群算法的全局寻优能力,加快了算法的收敛速度。实验表明该算法的实际计算效果优于已有的投影算法和光滑牛顿算法,更重要的是该算法不需要对适应度函数作光滑化处理和不依赖初始点的选择,拓宽了求解广义博弈 Nash 均衡算法的适用范围。

参考文献:

[1] DEBREU G. A social equilibrium existence theorem[J]. Proc of the National Academy of Sciences,1952,38(10):886-893.

[2] ARROW K J,DEBREU G. Existence of an equilibrium for a competitive economy[J]. Econometrical,1954,22(3):265-290.

[3] HARKER P T. A variational inequality approach for the determination of oligopolistic market equilibrium [J]. Mathematical Programming,1984,30(1):105 -111.

[4] URYASEV S,RUBINSTEIN R Y. On relaxation algorithms in computation of noncooperative equilibria[J]. IEEE Trans on Automatic Control,1994,39(6):1263-1267.

[5] FACCHINEI F,FISCHER A,PICCIALLI V. Generalized Nash equilibrium problems and Newton methods[J]. Mathematical Programming,2009,117(1-2):163-194.

[6] BRETON K J,ZACCOUR G,ZAHAF M. A game-theoretic formulation of joint implementation of environmental projects[J]. Eur J Oper Res,2005,168(1):221-239.

[7] 周晶,徐晏. 公共交通网络系统的经营模型[J]. 系统工程学报,2001,16(4):261-267.

[8] PANG J S,FUKUSHIMA M. Quasi-variational inequalities,generalized Nash equilibria and multi-leader-follower games[J]. Computational Management Science,2005,2(1):21-56.

[9] Von HEUSINGER A,KANZOW C. Optimization reformulations of the generalized Nash equilibrium problem using Nikaido-Isoda-type functions[J]. Comput Optim Appl,2009,43(3):353-377.

[10] ROUGHGARDEN T. Computing equilibria:a compution complexity perspective[J]. Economic Theory,2010,42(1):193-236.

[11] 李强. 求解广义纳什均衡问题的光滑牛顿法[D]. 大连:大连理工大学,2009.

[12] KENNEDY J,EBERHART R C. A new optimizer using particles swarm theory[C]//Proc of the 6th International Symposium on Micro Machine and Human Science. 1995:1942-1948.

[13] 黄席樾,向长城,殷礼胜. 现代智能算法理论及应用[M]. 2 版. 北京:科学出版社,2009.

[14] 纪震,廖惠连,吴青华. 粒子群算法及应用[M]. 北京:科学出版社,2010.

[15] 高鹰,谢胜利. 混沌粒子群优化算法[J]. 计算机科学,2004,31(8):13-15.

[16] 张兰,雷秀娟,马千知. 基于粒子群优化算法的多交叉口信号配时[J]. 计算机应用研究,2010,27(4):1252-1256.

[17] ANGELINE P J. Using selection to improve particle swarm optimization[C]//Proc of IEEE International Conference on Evolutionary Computation. 1998:84-89.

[18] DEJONG K A. Analysis of the behavior of a class of genetic adaptive system[D]. Ann Arbor: University of Michigan, 1975.

[19] ZHANG Jian-zhong, QU Biao, XIU Nai-hua. Some projection-like methods for the generalized Nash equilibria[J]. Comput Optim Appl,2010,45(10):89-109.

[20] KESSELMAN A,LEONARDI S,BONIFACI V. Game-theoretic analysis of Internet switching with selfish users[C]//Lecture Notes in Computer Science,vol 3828. 2005:236-245.