RESEARCH ARTICLE

A Differential Evolution Algorithm Based on Nikaido-Isoda Function for Solving Nash Equilibrium in Nonlinear Continuous Games

Feng He¹, Wei Zhang¹, Guoqiang Zhang^{1,2*}

¹ College of Management and Economics, Tianjin University, Tianjin, China, ² Xuchang University, Xuchang, China

* zgq@tju.edu.cn

OPEN ACCESS

Citation: He F, Zhang W, Zhang G (2016) A Differential Evolution Algorithm Based on Nikaido-Isoda Function for Solving Nash Equilibrium in Nonlinear Continuous Games. PLoS ONE 11(9): e0161634. doi:10.1371/journal.pone.0161634

Editor: Wei-Xing Zhou, East China University of Science and Technology, CHINA

Received: April 18, 2016

Accepted: August 9, 2016

Published: September 2, 2016

Copyright: © 2016 He et al. This is an open access article distributed under the terms of the [Creative Commons Attribution License](https://creativecommons.org/licenses/by/4.0/), which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.

Data Availability Statement: Data are generated from the experiment, the comparable algorithm is described in the paper "Koh, A. An evolutionary algorithm based on Nash Dominance for Equilibrium Problems with Equilibrium Constraints[J]. Applied Soft Computing, 2012, 12:161-173." All the codes for our algorithm can be obtained at: https://figshare.com/articles/A_differential_evolution_algorithm_based_on_Nikaido-Isoda_function_for_solving_Nash_equilibrium_in_nonlinear_continuous_games/3574116.

Funding: This manuscript is supported by National Natural Science Foundation of China (71320107003,

Abstract

A differential evolution algorithm for solving Nash equilibrium in nonlinear continuous games is presented in this paper, called NIDE (Nikaido-Isoda differential evolution). At each generation, parent and child strategy profiles are compared one by one pairwise, adapting Nikaido-Isoda function as fitness function. In practice, the NE of nonlinear game model with cubic cost function and quadratic demand function is solved, and this method could also be applied to non-concave payoff functions. Moreover, the NIDE is compared with the existing Nash Domination Evolutionary Multiplayer Optimization (NDEMO), the result showed that NIDE was significantly better than NDEMO with less iterations and shorter running time. These numerical examples suggested that the NIDE method is potentially useful.

Introduction

Nash equilibrium (NE) is one of the most important concepts in game theory, since it reveals the intrinsic link between the game equilibrium and economic equilibrium. NE is widely used in practical applications, therefore, it is a key issue to find NE. However, infinite strategies and complex payoff functions (such as non-concave, non-differentiable, multi-modal, etc.) make it difficult to solve NE. In fact, the difficulty of solving the NE has greatly limited its application in economics, management and sociology.

Economists mainly studied continuous games, in which there are infinite strategies and the payoff functions are also continuous. As we all know, the purpose of game analysis is to predicate a result of a game, which is NE. At present, the commonly used methods for locating NE of a continuous game are the following.

(1) Analytical method is the most straightforward approach, which sets the partial derivatives of every player to zero[1], thus we can obtain the optimum strategy for one player if the other players do not change their strategies. This is also known as player's best-response function. Solving all the best-response function can obtain the optimal strategy of each player, that is NE[2,3]. The analytical method is useful for providing insights into the behavior of players,

71532009). The funders had no role in study design, data collection and analysis, decision to publish, or preparation of the manuscript.

Competing Interests: The authors have declared that no competing interests exist.

however, it is difficult to apply this method to deal with multi-player or non-differential payoff function [4,5].

(2) Mathematical programming is the most widely used method in detecting NE, which is convenient to add certain constraint to the model, and easy to extend to a big system. The mathematical programming method can be divided into the following categories: sensitivity information based [6]; non-interior point algorithm[7]; nonlinear complementarity approach [8], etc. Furthermore, a variational inequality approach [9] has also been applied in Cournot model. However, the mathematical programming method is heavily dependent on the choice of the initial point. If we set an improper initial point, the solution may only be a local optimal solution.

(3) Matrix game is a method which disperses the strategy space, therefore, each player's strategy space is composed of several typical strategies [10]. In addition, the matrix game is also used in the mixed NE[11].

(4) Evolutionary algorithm is a stochastic optimization method, which is not strictly limited to the objective function and can also handling non-differentiable, nonlinear and other complex payoff functions. Son[12] proposed an improved co-evolutionary algorithm—hybrid co-evolutionary programming to find a global optimal NE, which overcame the local optimal shortcoming and improved the convergence of the co-evolutionary algorithm. There are also other optimization algorithms such as invasive weed optimization[13]; cooperative co-evolutionary invasive weed optimization[14]; PSO[15], etc.

In many economic models, there are some special functions to convert NE into a optimization problem, such as Nikaido-Isoda function[16] and Lyapunov function[17]. Moreover, equilibrium also can be described as solution of polynomial equations [18].

When building a game model, economists generally assume that the demand curves are monotonically decreasing functions, and the cost functions are constant or represented by a quadratic functions. The advantage of this assumption is to guarantee a unique NE, which is convenient to find NE. In real economic activity, however, the nonlinear demand functions and cost functions made the model more consistent with the real market competition, this will inevitably lead to the payoff functions become extremely complex, thus making the NE difficult to be solved.

A differential evolution algorithm(DE) for solving Nash equilibrium in nonlinear continuous games is presented in this paper, called NIDE. In each generation, parent and child strategy profiles are compared one by one pairwise with Nikaido-Isoda as fitness function. In applied research, the NE of nonlinear game model with cubic cost function and quadratic demand function(even some other non-concave payoff functions) are solved. Compared with the existing Nash Domination Evolutionary Multiplayer Optimization (NDEMO), the results suggested that the NIDE was significantly better than NDEMO with less iterations and shorter runtime for solving the NE problems. Meanwhile, it illustrated that the NIDE has high efficiency and wide applicability.

The organization of this paper is as following. Section 2 explains the basic concepts of game theory and provides a searching algorithm based on DE. Section 3 presents numerical examples and compared with NDEMO, and finally conclusions are drawn afterwards

Materials and Methods

Prerequisites

The normal-form game usually consists of three parts:

$N = \{1, \dots, n\}$ denotes the set of players, n is the number of players;

$S_i = \{S_{i1}, S_{i2}, \dots, S_{im_i}\}$ denotes the strategies set of player i ;

$u_i: S \rightarrow R$ denotes the payoff function for each player $i \in N$, where $S = S_1 \times S_2 \times \cdots \times S_n$ is a Cartesian product and represents all possible situation of the game, and R is set of real number.

Given a particular topology for a set S_i , there is corresponding direct product topology for the set $S = S_1 \times S_2 \times \cdots \times S_n$. For example, if S_i is a compact metric space, then S is also a compact metric space. As players' strategies can be quantified, so the pure strategy of each player S_i can be regarded as a bounded and closed convex subset in Euclidean space. If the number of strategies S_i for player i is infinite, it is called "Infinite game".

Definition 1: The infinite game $G = \{s_1, \cdots, s_n; \mu_1, \cdots, \mu_n\}$ is called continuous game, if given a certain topology for a set S , and the payoff function $u_i: S \rightarrow R$ is a continuous function for player $i \in N$.

In an N -player game, when player i chooses his pure strategy $s_i \in S_i$ given a pure strategy set of others $s_{-i} = (s_1, \cdots, s_{i-1}, s_{i+1}, \cdots, s_n) \in S_{-i}$, he receives a profit of $u_i(s_i, s_{-i})$. In order to study the equilibrium of continuous game, we need introduce the concept of NE.

Definition 2: In the normal-form game $G = \{s_1, \cdots, s_n; \mu_1, \cdots, \mu_n\}$, a combined strategy profile $s^* = (s_1^*, \cdots, s_i^*, \cdots, s_n^*)$ is an NE if

$$u_i(s_i^*, s_{-i}^*) \geq u_i(s_i, s_{-i}^*), \forall s_i \in S_i, \forall i \in N, \quad (1)$$

NE is a point where no player can obtain a higher profit by unilateral movement. That is, a profit-maximizing player will not deviate from NE. In fact, if a strategy profile $s' = (s'_1, \cdots, s'_i, \cdots, s'_n)$ is not NE, that means there must exist a player i who can obtain a higher profit by choosing another strategy s'_i when others choose s_{-i}^* , that is $u_i(s'_i, s_{-i}^*) > u_i(s_i^*, s_{-i}^*)$.

Definition 3: In an N -player game, a strategy s_i^* is called "dominant strategy" for player i , if

$$u_i(s_i^*, s_{-i}) > u_i(s'_i, s_{-i}), \forall s_{-i} \in S_{-i}, \forall s'_i \in S_i, s'_i \neq s_i^* \quad (2)$$

Accordingly, the strategy s'_i is called dominated strategy. From the definition 2, if $s_i^* \in S$ is a dominant strategy for player i , the outcome resulting from the strategy s_i^* is better than the outcome resulting from any strategy $s'_i \in S_i (s'_i \neq s_i^*)$, regardless of what the other players are doing.

Sometimes, a strategy to be a dominant strategy is only relative to the specific strategy, thus we have the following definition.

Definition 4: In a game, a player may have two strategies s'_i and s''_i , then strategy s''_i is said to relatively dominant strategy s'_i , than is

$$u_i(s'_i, s_{-i}) < u_i(s''_i, s_{-i}), \forall s_{-i} \quad (3)$$

It means that there may exist another better strategy than s'_i .

Next, we will introduce the concept of Nikaido-Isoda function[19], which plays an important role in the study of NE.

Definition 5: Let u_i be the payoff function of player i , then the Nikaido-Isoda function $\psi: (S_1 \times S_2 \times \cdots \times S_n) \times (S_1 \times S_2 \times \cdots \times S_n) \rightarrow R$ is

$$\psi(x, y) = \sum_{i=1}^n [u_i(y_i | x) - u_i(x)]. \quad (4)$$

Each term of the series in the Nikaido-Isoda function represents the change of one player's gains from strategy x_i to y_i , while the strategies of the others are held fixed. Nikaido-Isoda function represents the sum of the change in gains or losses of all players.

Based on the definition 4 and definition 5, given two strategy profiles

$$\{x, y\} \in S, (x \equiv (x_1, \cdots, x_n), y \equiv (y_1, \cdots, y_n)), y_i \neq x_i,$$

if $u_i(y_i|x) > u_i(x)$ always hold for each player $i \in N$, we say the strategy profile y is dominant to the strategy profile x . at this point, we have the following proposition hold.

Proposition 1: If the strategy profile y is dominant to the strategy profile x , then

$$\psi(y, x) < \psi(x, y). \quad (5)$$

Prove: If the strategy profile y is dominant to the strategy profile x , then

$$u_i(x_i, s_{-i}) < u_i(y_i, s_{-i}), \forall s_{-i}$$

substitute it into Eq (4), we have

$$\psi(x, y) = \sum_{i=1}^n [u_i(y_i|x) - u_i(x)] > 0$$

Similarly,

$$\psi(y, x) = \sum_{i=1}^n [u_i(x_i|y) - u_i(y)] < 0$$

Therefore, $\psi(y, x) < \psi(x, y)$ holds.

If x^* is NE, for any $y \in S$, and $y \neq x^*$, then $\psi(x^*, y) < \psi(y, x^*)$. Therefore the Nikaido-Isoda function plays the role of a “merit function” measuring the proximity of a strategy to NE.

NIDE algorithm

In 1995, Storn and Price[20–22] proposed a advanced evolutionary algorithm, called differential evolutionary(DE), which was a relatively new stochastic optimization method, and has been shown to be a simple and effective algorithm for the problem of continuous global optimization[21]. The DE requires few control variables, is robust, easy to use and lends itself very well to parallel computation.

There are three important variables in DE: Mutation Factor F , Probability of Crossover CR and Population size NP . Mutation Factor F is positive real number which controls the amplification of the differential variation. Experience has shown that, it is best to choose F between 0.4 and 1.0. In addition, a crossover operation is performed, using the parameter CR , the value of CR is smaller, the robustness of DE will be better. Generally, $CR \in [0, 1]$. If the population size NP is larger, the search ability of DE will be intensified, and the calculated amount of DE will be increased. According to literature [21], NP usually to be chosen between $5n-10n$, where n is the number of variables.

Based on Proposition 1, by redefinition of selection criteria used in evolutionary methods with the Nikaido-Isoda function, we propose a differential evolution(NIDE) to find the NE. At each generation, parent and child strategy profiles are compared one by one pairwise, using Nikaido-Isoda function to play for the part of fitness function. The whole process can be summarized into the following steps(Algorithm1).

Algorithm1: A differential evolution based on Nikaido-Isoda function (NIDE)

```

1: Input: the maximum number of iterations  $Max_{it}$ , the population size  $NP$ ,
Mutation Factor  $F$ , Probability of Crossover  $CR$ , the convergence tolerance  $\varepsilon$ 
2:  $it \leftarrow 1$ 
3: Randomly initialize populations of  $NP$  parent strategy profiles  $P$ 
4: Evaluate fitness to players with  $P$ 
5: while  $it < Max_{it}$  do
6:   for  $i = 1$  to  $NP$  do
7:      $x \leftarrow P_i^{it}$ 
8:     Use Algorithm 2 to create child strategy profiles  $y$ 
```

```

9:   if  $\psi(y, x) < \psi(x, y)$  then
10:      $P_i^{it+1} \leftarrow y$ 
11:   end if
12: end for
13: Randomly choose a chromosome  $s$  from  $P$ 
14: Compute Euclidean norm between  $s$  and every member in  $P$ 
15: if maximum of norm  $\leq \epsilon$  then
16:   Terminate
17: else
18:    $it \leftarrow it + 1$ 
19: end if
20: end while
21: Output: Nash Solutions

```

Child strategy profiles y are created by applying the DE operators via the stochastic combination of randomly chosen parents as discussed in Storn(2003) [23]. Through tested on five different forms of DE operators(DE/rand/1/bin, DE/best/1/bin, DE/rand/2/bin, DE/best/2/bin, DE/rand-to-best/1/bin), we found the operator “DE/rand/1/bin” possessed faster convergence. In order to show the performance of NIDE, we use the same algorithm in literature Koh(2012) [24] to create child strategy profiles, just describe by the following(Algorithm 2).

Algorithm 2: Creating a child vector via DE

```

1: Input: Current Population  $P$ 
2: Input: Mutation Factor  $F$ , Probability of Crossover  $CR$ 
3: Input: Lower Bounds  $LB_d$  and Upper Bounds  $UB_d$  in each dimension  $d$ 
4: Randomly choose 3 integers:  $r_1, r_2, r_3$  between 1 and  $NP$ 
5:  $r_1 \neq j, r_2 \neq j$  and  $r_3 \neq r_2 \neq r_1 \neq j$ 
6:  $x \leftarrow P_j$ 
7:  $a \leftarrow P_{r_1}$ 
8:  $b \leftarrow P_{r_2}$ 
9:  $c \leftarrow P_{r_3}$ 
10: Mutation: Produce a mutant vector  $z$  via a stochastic combination of donor vectors
11:  $z \leftarrow a + F(b - c)$ 
12: for  $d = 1$  to  $D$  do
13:   Crossover
14:    $y_d \leftarrow \begin{cases} z_d & \text{if } \text{rand}(0, 1) < CR \vee d = \text{intr}(1, D) \\ x_d & \text{otherwise} \end{cases}$ 
15:   Enforce Bound Constraints
16:    $y_d \leftarrow \begin{cases} (x_d + LB_d)/2 & \text{if } y_d < LB_d \\ (x_d + UB_d)/2 & \text{if } y_d > UB_d \\ y_d & \text{otherwise} \end{cases}$ 
17: end for
18: Output: child vector  $y$ 

```

In order to check convergence to the NE, randomly select a individual s and compute the Euclidean norm between s and every member of P (lines 13 to 14 of Algorithm1). If the maximum distance is less than ϵ , the population is judged to have converged to a NE and the algorithm can terminated. Otherwise the process is repeated.

Results and Discussion

In order to test the efficiency of the NIDE, we make a comparison between NIDE and NDEMO[24]. Table 1 gives the parameters used for the numerical experiments just the same in Koh(2012)[24]. All numerical experiments were conducted using Matlab R2013a running on a 32 bit Windows 7 machine with 2 GB of RAM.

Table 1. Parameters used in the NIDE for all numerical experiments.

Parameter name	Parameter values
Maximum number of iterations	400
Population size	50
Probability of crossover	0.35
Mutation factor	0.45
Termination tolerance	0.0001

doi:10.1371/journal.pone.0161634.t001

Oligopolistic game model with cubic cost function

When building a game model, Economists generally assume that the demand curves are monotonically decreasing functions, and the cost functions are constant or represented by a quadratic functions. The advantage of this assumption is there exists a unique NE, and it is convenient to finding the NE. In real economic activity, however, the nonlinear demand functions and cost functions make the model more consistent with the real market competition, this will inevitably lead to the payoff functions become extremely complex, thus making the NE difficult to be solved.

In microeconomics, however, the total cost function is always non-negative in its definitional domain and decreases to zero on the left side of the inflection point, then it gradually increases while on the right side of the inflection point[25]. The literature [25] pointed out the total cost curve is the analogy of cubic function, so we employ

$$f(x) = a + bx + cx^2 + dx^3 (d \neq 0) \quad (6)$$

There are four forms of cubic function. $d > 0$ and $\Delta \leq 0$; $d < 0$ and $\Delta \leq 0$; if $d < 0$ and $\Delta > 0$; $d < 0$ and $\Delta > 0$ and $\Delta = 4c^2 - 12bd$.

The total cost function is a cubic function as $a > 0$, $d > 0$, $\Delta \leq 0$, and the inflection point $(-\frac{c}{3d}, f(-\frac{c}{3d}))$ falls in the first quadrant, that is,

$$\left\{ \begin{array}{l} a > 0 \\ d > 0 \\ \Delta = 4c^2 - 12bd \leq 0 \\ -\frac{c}{3d} > 0 \\ f(-\frac{c}{3d}) = 2c^3 - 9bcd + 27ad^2 > 0 \end{array} \right. \quad (7)$$

Case 1: Example 1 with two players. Here we first consider the situation with two players $i, i = 1, 2$. Suppose that the output of player i is q_i , each player maximizes individual profits from the sale of the homogeneous good. The total outputs are

$$Q = q_1 + q_2,$$

and the inverse demand function[26] is

$$P = P(Q) = m - nQ^2.$$

The total cost curve is

$$C_i(q_i) = a_i + b_i q_i + c_i q_i^2 + d_i q_i^3, i = 1, 2$$

Hence the profit function of player i is given by

$$\pi_i(q_1, q_2) = q_i[m - nQ^2] - C_i(q_i), \quad (8)$$

We calculate the first order of the profit, i.e.

$$\begin{cases} \frac{\partial \pi_1}{\partial q_1} = m - nQ^2 - 2nq_1Q - b_1 - 2c_1q_1 - 3d_1q_1^2 = 0 \\ \frac{\partial \pi_2}{\partial q_2} = m - nQ^2 - 2nq_2Q - b_2 - 2c_2q_2 - 3d_2q_2^2 = 0 \end{cases} \quad (9)$$

Based on Eq (9), we can find out the player's response function,

$$\begin{cases} q_1^* = \frac{1}{3n + 3d_1}(-2nq_2 - c_1 + \sqrt{M}) \\ q_2^* = \frac{1}{3n + 3d_2}(-2nq_1 - c_2 + \sqrt{N}) \end{cases} \quad (10)$$

Where

$$M = (n^2 - 3nd_1)q_2^2 + 4nc_1q_2 + c_1^2 + 3(mn - nb_1 + md_1 - b_1d_1)$$

$$N = (n^2 - 3nd_2)q_1^2 + 4nc_2q_1 + c_2^2 + 3(mn - nb_2 + md_2 - b_2d_2).$$

It is difficult to obtain the analytical solutions in Eq (10), so we assign a value to each parameter, then use NIDE to detect the NE.

Set

$$m = 5, n = 1,$$

$$a_1 = 0.01, b_1 = 0.4, c_1 = -0.03, d_1 = 0.005,$$

$$a_2 = 0.01, b_2 = 0.35, c_2 = -0.025, d_2 = 0.006,$$

where a_i, b_i, c_i, d_i submit to the conditions (7).

The NIDE identified the NE(0.7563, 0.7697) successfully. In order to compare the iterations and runtime, the NIDE and NDEMO were both runned 10 times, the results were showed in (Fig 1 and Fig 2). The compare results produced by the NIDE and NDEMO were given in Table 2.

Case 2: Example 1 with three players. For a more complex case, we consider the three player model. We use the same forms of cost and demand function as in Case 1. Three players maximize their profit by adjusting their strategic variable q_1, q_2 and q_3 . The profit functions of each player are analogous to those in (8).

$$\text{Set } m = 5, n = 1; a_1 = 0.01, b_1 = 0.4, c_1 = -0.03, d_1 = 0.005; a_2 = 0.01, b_2 = 0.35, c_2 = -0.025, d_2 = 0.006; a_3 = 0.01, b_3 = 0.3, c_3 = -0.02, d_3 = 0.007.$$

where a_i, b_i, c_i, d_i submit to the conditions (7).

The NIDE could successfully identify the NE(0.5450, 0.5581, 0.5712)(Fig 3 and Fig 4) as well as Table 3 also show the comparative results of iterations and runtime.

Next, in order to show the high efficiency and wide applicability of NIDE, we introduce two more complex models.

Fig 1. Comparison of iterations between NIDE and NDEMO of Case 1 with two players.

doi:10.1371/journal.pone.0161634.g001

Fig 2. Comparison of runtime of between NIDE and NDEMO Case 1 with two players.

doi:10.1371/journal.pone.0161634.g002

Table 2. Comparison of the results of NIDE with NDEMO.

Algorithm	NIDE	NDEMO
The average number of iterations	68	206
The average number of runtime(s)	0.60345	1.85430
NE	0.7563, 0.7697	

doi:10.1371/journal.pone.0161634.t002

A game model with local NE

This example is taken from [12]. Suppose that there are two players in a market, and the non-concave profit functions for players are defined by

$$\begin{aligned}\pi_1(x_1, x_2) &= 21 + x_1 \sin(\pi \times x_1) + x_1 x_2 \sin(\pi \times x_2) \\ \pi_2(x_1, x_2) &= 21 + x_2 \sin(\pi \times x_2) + x_1 x_2 \sin(\pi \times x_1)\end{aligned}\quad (11)$$

where $\pi = 3.1415$, and the strategic choice of the two players between 0 and 7.5.

We calculate the first order of the profit, i.e.

$$\begin{cases} \frac{\partial \pi_1}{\partial x_1} = \sin(\pi \times x_1) + \pi \times x_1 \cos(\pi \times x_1) + x_2 \sin(\pi \times x_2) = 0 \\ \frac{\partial \pi_2}{\partial x_2} = \sin(\pi \times x_2) + \pi \times x_2 \cos(\pi \times x_2) + x_1 \sin(\pi \times x_1) = 0 \end{cases}\quad (12)$$

Through (Fig 5), we know that the Eq (12) have many solutions(the points of intersection of Fig 5) in the domain $[0, 7.5]$, and it is difficult to differentiate real NE and “local NE traps” use the conventional NE search algorithms.

Fig 3. Comparison of iterations between NIDE and NDEMO of Case 2 with two players.

doi:10.1371/journal.pone.0161634.g003

Fig 4. Comparison of runtime of between NIDE and NDEMO of Case 2 with two players.

doi:10.1371/journal.pone.0161634.g004

The NIDE overcomes the local optima and detects the NE which agrees with previous results. The comparative results of iterations and runtime are shown in (Fig 6 and Fig 7) and Table 4.

A game model with 5 players

This example is taken from Koh(2012)[24]. In this example, there are 5 firms, each firm maximizes individual profits from the sale of the homogeneous good, and inverse demand function is

$$P(q_1, q_2, q_3, q_4, q_5) = 5000^{1/1.1} (\sum_{i=1}^5 q_i)^{-(1/1.1)},$$

the total cost curve is

$$C_i(q_i) = \omega_i q_i + \left(\frac{\theta_i}{\theta_i + 1} \right) \lambda_i^{-1/\theta_i} q_i^{(\theta_i+1)/\theta_i}.$$

Table 3. Comparison of the results of NIDE with NDEMO.

Algorithm	NIDE	NDEMO
The average number of iterations	107	297
The average number of runtime(s)	1.05629	3.05636
NE	0.5450, 0.5581, 0.5712	

doi:10.1371/journal.pone.0161634.t003

Fig 5. NE in a game model with local NE.

doi:10.1371/journal.pone.0161634.g005

Then, the non-concave profit functions for player $i \in \{1, 2, 3, 4, 5\}$ is given by

$$\begin{aligned} U_i(q_1, q_2, q_3, q_4, q_5) &= P(q_1, q_2, q_3, q_4, q_5)q_i - C_i(q_i) \\ &= \left(5000^{1/1.1} (\sum_{i=1}^5 q_i)^{-(1/1.1)}\right) q_i - \omega_i q_i - \left(\frac{\theta_i}{\theta_i + 1}\right) \lambda_i^{-1/\theta_i} q_i^{(\theta_i+1)/\theta_i} \end{aligned} \quad (13)$$

The player dependent parameters shown in Table 5.

(Fig 8 and Fig 9) and Table 6 reports results obtained from applying NIDE and NDEMO to this game.

Electricity market with two players. Electricity market is a classical linear Cournot model [12]. For sample, we assume there are only two players, and they submit bids to generate quantities s_1 and s_2 , respectively.

In order to analyze the strategy of each player, we define each player's profit by

$$\begin{aligned} \pi_1(s_1, s_2) &= \lambda s_1 - C1(s_1) \\ \pi_2(s_1, s_2) &= \lambda s_2 - C2(s_2) \end{aligned} \quad (14)$$

where $\lambda = [2750 - 4(s_1 + s_2)]/75$, cost functions $C1 = 0.01s_1^2 + 10s_1$, and $C2 = 0.01s_2^2 + 10s_2$.

Fig 6. Comparison of iterations between NIDE and NDEMO of local NE game.

doi:10.1371/journal.pone.0161634.g006

Fig 7. Comparison of runtime of between NIDE and NDEMO of local NE game.

doi:10.1371/journal.pone.0161634.g007

Table 4. Comparison of the results of NIDE with NDEMO.

Algorithm	NIDE	NDEMO
The average number of iterations	52	105
The average number of runtime(s)	0.48559	1.09200
NE	6.6118	6.6118

doi:10.1371/journal.pone.0161634.t004

Table 5. The parameter specification for players.

Firm i	ω_i	λ_i	θ_i
1	10	5	1.2
2	8	5	1.1
3	6	5	1.0
4	4	5	0.9
5	2	5	0.8

doi:10.1371/journal.pone.0161634.t005

Fig 10 and Fig 11 and Table 7 reports results obtained from applying NIDE and NDEMO to this game.

From Table 7, NIDE's average running time is 0.2s with only 66 generations of evolution, while NDEMO took 0.5s with 122 generations. It proved the efficiency of NIDE algorithm is still superior to NDEMO in classical liner demand function model.

Fig 8. Comparison of iterations between NIDE and NDEMO of a game model with 5 players.

doi:10.1371/journal.pone.0161634.g008

Fig 9. Comparison of runtime of between NIDE and NDEMO of a game model with 5 players.

doi:10.1371/journal.pone.0161634.g009

Conclusions

In this paper, we proposed an evolutionary algorithm for solving NE in nonlinear continuous games. The algorithms of NIDE enabled us to handle the NE problems of nonlinear game model with the cubic cost function and the quadratic demand function, and even other non-concave payoff functions.

In order to assess the performance of NIDE, we compared the NIDE with the existing differential evolution algorithm(NDEMO), comparison of the results showed that the NIDE was significantly better than NDEMO with less iterations and runtime for solving the NE problems. It is more efficient in locating NE in practice and will not result in local optimal.

Meanwhile, through these numerical examples, suggesting that the NIDE could be a potentially useful method.

Table 6. Comparison of the results of NIDE with NDEMO.

Algorithm	NIDE	NDEMO
The average number of iterations	136	277
The average number of runtime(s)	2.92571	6.85258
NE	36.9325, 41.8182, 43.7066, 42.6592, 39.1789	

doi:10.1371/journal.pone.0161634.t006

Fig 10. Comparison of iterations between NIDE and NDEMO of Electricity market with two players.

doi:10.1371/journal.pone.0161634.g010

Fig 11. Comparison of runtime of between NIDE and NDEMO of Electricity market with two players.

doi:10.1371/journal.pone.0161634.g011

Table 7. Comparison of the results of NIDE with NDEMO.

Algorithm	NIDE	NDEMO
The average number of iterations	66	122
The average number of runtime(s)	0.20571	0.45258
NE	148.148, 148.148	

doi:10.1371/journal.pone.0161634.t007

Acknowledgments

The authors would like to acknowledge Prof. Minqiang Li, Prof. Dominique Guegan in supervising the PhD research; anonymous reviewers of providing helpful feedback and suggestions; National Natural Science Foundation of China (71320107003, 71532009) to provide financial support of this research.

Author Contributions

Conceptualization: FH GQZ WZ.

Formal analysis: FH GQZ.

Investigation: GQZ.

Methodology: GQZ.

Project administration: WZ.

Resources: WZ.

Software: FH GQZ.

Supervision: WZ.

Validation: FH GQZ.

Visualization: FH.

Writing – original draft: GQZ.

Writing – review & editing: FH WZ.

References

1. Song Y, Ni Y, Wen F, Hou Z, Wu FF. Conjectural variation based bidding strategy in spot markets: fundamentals and comparison with classical game theoretical bidding strategies[J]. Electric Power Systems Research, 2003, 67(1): 45–51.
2. Park J B, Kim J H, Kim J H, Jung M H, Park JK. A continuous strategy game for power transactions analysis in competitive electricity markets[J]. Power Systems, IEEE Transactions on, 2001, 16(4): 847–855.
3. Wen FS, David AK. PES News[J]. Power Engineering Review, IEEE, 2001, 21(4): 24–28.
4. Baldick R, Grant R, Kahn EP. Linear supply function equilibrium: generalizations, application, and limitations [M]. Berkeley, CA: University of California Energy Institute, 2000.
5. Baldick R, Hogan WW. Capacity constrained supply function equilibrium models of electricity markets: stability, non-decreasing constraints, and function space iterations [M]. University of California Energy Institute, 2001.
6. Weber JD, Overbye TJ. A two-level optimization problem for analysis of market bidding strategies. Proceeding of the IEEE Power Engineering Society Summer Meeting, Edmondson, Alberta, July 1999, pp: 682–687.
7. De LLM, Granville S. The stackelberg equilibrium applied to AC power systems—a non-interior point algorithm. IEEE Transactions on Power Systems, 2003, 18(2): 611–618.

8. Wang X, Li YZ, Zhang SH. Oligopolistic equilibrium analysis for electricity markets- a nonlinear complementarity approach. *IEEE Transaction on Power Systems*, 2004, 19(3): 1348–1355.
9. Wei JY, Smeers Y. Spatial oligopolistic electricity models with Cournot generators and regulated transmission prices. *Operations Research*, 1999, 47(1): 102–112.
10. Moitre D. Nash equilibria in competitive electric energy markets[J]. *Electric power systems Research*, 2002, 60(3): 153–160.
11. Lee KH, Baldick R. Tuning of discretization in bimatrix game approach to power system market analysis[J]. *Power Systems, IEEE Transactions on*, 2003, 18(2): 830–836.
12. Son YS, Baldick R. Hybrid coevolutionary programming for Nash equilibrium search in games with local optima[J]. *Evolutionary Computation, IEEE Transactions on*, 2004, 8(4): 305–315.
13. Sahraei-Ardakani M, Roshanaei M, Rahimi-Kian A, Lucas C. A study of electricity market dynamics using invasive weed colonization optimization [C]// *Computational Intelligence and Games*, 2008. CIG'08. *IEEE Symposium On. IEEE*, 2008: 276–282.
14. Hajimirsadeghi H, Ghazanfari A, Rahimi-Kian A, Lucas C. Cooperative coevolutionary invasive weed optimization and its application to Nash equilibrium search in electricity markets[C]// *Nature & Biologically Inspired Computing*, 2009. NaBIC 2009. *World Congress on. IEEE*, 2009: 1532–1535.
15. [Peydayesh M. A PSO based algorithm for finding global Nash equilibrium[C]// *Power and Energy Society General Meeting*, 2010 *IEEE. IEEE*, 2010: 1–6.
16. Nikaido H, Isoda K. Note on noncooperative convex games[J]. *Pacific Journal of Mathematics*, 1955, 5: 807–815.
17. McKelvey R. A Lyapunov Function for Nash Equilibria [R]. *Technical Report*, California Institute of Technology, 1992.
18. Eaves BC, Schmedders K. General equilibrium models and homotopy methods[J]. *Journal of Economic Dynamics and Control*, 1999, 23(9): 1249–1279.
19. Nikaido H, Isoda K. Note on noncooperative convex games[J]. *Pacific Journal of Mathematics*, 1955, 5: 807–815.
20. Storn R, Price K. Differential evolution-a simple and efficient adaptive scheme for global optimization over continuous spaces[M]. *Berkeley: ICSI*, 1995.
21. Storn R, Price K. Differential evolution—a simple and efficient heuristic for global optimization over continuous spaces[J]. *Journal of global optimization*, 1997, 11(4): 341–359.
22. Storn R, Price K. Minimizing the Real Functions of the ICEC'96 Contest by Differential Evolution[C]// *International Conference on Evolutionary Computation*. 1996: 842–844.
23. Storn R, Price K V, Lampinen J. Differential evolution-a practical approach to global optimization[J]. 2005.
24. Koh A. An evolutionary algorithm based on Nash Dominance for Equilibrium Problems with Equilibrium Constraints[J]. *Applied Soft Computing*, 2012, 12:161–173.
25. Pu X, Ma J. Complex dynamics and chaos control in nonlinear four-oligopolist game with different expectations[J]. *International Journal of Bifurcation and Chaos*, 2013, 23(03).
26. Zhang YB., Luo XJ, Xue JY. Adaptive dynamic Cournot model of optimizing generating units power output under nonlinear market demand, *Proc. CSEE (China)2003*, 23, 80–84.