International Journal of Industrial Engineering Computations 15 (2024) 189–208

Contents lists available at GrowingScience

International Journal of Industrial Engineering Computations

homepage: www.GrowingScience.com/ijiec

A dynamic incentive mechanism for data sharing in manufacturing industry

Ruihan Liu[a], Yang Yu[b] and Min Huang[c*]

[a]College of Information Science and Engineering; Northeastern University, Shenyang, Liaoning, 110819, P.R. China
[b]School of Economics and Management, Dalian University of Technology, Dalian, Liaoning, 116024, P.R. China
[c]College of Information Science and Engineering; Northeastern University, Shenyang, Liaoning, 110819, P.R. China

C H R O N I C L E

Article history:
Received August 5 2023
Received in Revised Format September 6 2023
Accepted October 23 2023
Available online October 23 2023

Keywords:
Data sharing
Dynamic incentive mechanism
Evolutionary game
Networked evolutionary game
Q-Learning

A B S T R A C T

Data sharing is a critical component in a blockchain traceability platform. Therefore, creating a reasonable incentive mechanism to ensure that all enterprises participate in data sharing is vital for blockchain platforms. Currently, many researchers employ evolutionary game theory to analyze problems related to data sharing. However, evolutionary game theory typically assumes that the population composed of enterprises is mixed uniformly. Enterprises in the manufacturing industry are not uniformly mixed, as they tend to have specific connections with each other due to the size of enterprises and volume of business. Therefore, a networked evolutionary game is introduced to solve this problem. Firstly, an incentive model for enterprises sharing data is established. Then, a scale-free network is employed to simulate the connections between enterprises. To comprehensively consider the individual and group benefits of enterprises in the game, this study designs a strategy update rule for networked evolutionary game based on Discrete Particle Swarm Optimization and Variable Neighborhood Descent algorithm. To tackle the challenge of determining reasonable incentive values in networked evolutionary games, this study proposes a dynamic incentive mechanism based on the Q-Learning algorithm. Finally, the experiments indicate that this method can successfully facilitate the stable involvement of enterprises in data sharing.

1. Introduction

The manufacturing industry chain typically encompasses supply, processing (Feng & Yu, 2023), transportation, and sales (Wang & Du, 2007), each of which generates a substantial amount of production data. However, the multi-source, heterogeneous, and massive volume of data present significant challenges in achieving product information traceability. Traditional manufacturing traceability systems are often centralized, which poses a risk of information security in the event of a database breach (Feng et al., 2020). Additionally, the manufacturing industry chain is complex, and fragmented information undermines the value of data (Chen et al., 2019). The creation of a product must go through various stages, including raw material supply, processing, transportation, and sales, and data generated in each stage is sourced from different enterprises. Integrating data from different stages and participants on a traceability platform can result in high costs. Differences between traceability systems across different production stages may make it difficult to achieve traceability goals (Yang et al., 2021). Blockchain is a distributed ledger structure that is cooperatively maintained by several parties and uses cryptographic principles to assure transmission and access security (Lu et al., 2022; Wang et al., 2018; Chowdhury et al., 2018). It achieves consistent storage of data and makes information difficult to tamper with, thus solving the issue of information traceability in the process of collection, circulation, and sharing (Nofer et al., 2017). By combining blockchain technology with industrial product traceability, transparency and security can be greatly improved (Wen et al., 2019). Uploading traceability data from various links of the industrial chain to the blockchain can effectively solve the problems that

* Corresponding author
E-mail: mhuang@mail.neu.edu.cn (M. Huang)
ISSN 1923-2934 (Online) - ISSN 1923-2926 (Print)
2024 Growing Science Ltd.
doi: 10.5267/j.ijiec.2023.10.004

exist in traditional traceability systems. Data sharing in blockchain can be a great solution to the problem of traceability. However, there are still issues with the real data sharing process such as insufficient willingness of enterprises to share data (Guo et al., 2023), difficulty in guaranteeing data security (Cui et al., 2021), and imperfect blockchain platform for sharing incentive mechanism. Therefore, it is crucial to create a reasonable incentive mechanism which can ensure the stable participation of enterprises sharing data and establish a blockchain platform. In the study, we first develop an incentive model for enterprises participating in data sharing on the blockchain. Subsequently, we study the strategy evolution process of enterprises participation in data sharing under the given incentive using a network evolutionary game approach. Within the network evolutionary game, the study designs a strategy update rule based on both the Discrete Particle Swarm Optimization (DPSO) and Variable Neighborhood Descent (VND) algorithm. Finally, we propose a dynamic incentive mechanism based on Q-Learning algorithm (DIMQL) to solve the problem of difficulty in finding reasonable incentive values under network evolution games.

2. Related Work

Our research covers the fields of data sharing, evolutionary game theory, network evolutionary game, and Q-Learning. In this part, we review the literature in the fields. In the age of big data, the continuous growth of data has created challenges in data sharing. By its characteristics of decentralization and information sharing, blockchain technology is believed to become a new platform for data sharing (Yang et al., 2020, 2021). The incentives are designed to combine reputation and payment, and use "credit coins" as the carrier of incentives to encourage users to share data (Chen et al., 2022b). In the study on the motivations for data sharing, Guo et al. (2018) established the idea of data competitiveness. Based on considering data competition, Guo et al. (2019) adopted differential privacy to secure privacy and proposed a contract theory method to develop incentive mechanisms.

Game theory is applied to biological evolution in evolutionary game theory, which offers a strong framework to study the process of decision-making in the population of rational agents (Smith & Price, 1973). When the evolutionary game reaches equilibrium, the strategies of all enterprises reach a dynamic stable state. Even if a small number of enterprises are disturbed and deviate from the steady state, the evolutionary game can be corrected within a certain period so that the evolution reaches the stable state again. We want enterprises to share data stably, which is like the equilibrium of game. Therefore, evolutionary games are employed to study the issue of enterprise data sharing. The following is the current state of research on evolutionary games to solve data sharing problems: Gao (2019) utilized evolutionary games to examine the role of government subsidies on the sterilization of manufacturing enterprises. Xuan et al. (2020) analyzed the evolution of user participation in data sharing using evolutionary game theory and designed an adaptive smart contract mechanism to continuously incentivize user share data. Wang et al. (2021) investigated the reciprocity-based incentive mechanism in P2P systems using evolutionary games. Esmaeili et al. (2022) devised an incentive mechanism and applied an evolutionary game to study the cooperation of participants in a dilemma. Liu et al. (2022) studied the game between logistics platforms and suppliers using evolutionary game theory. Then, he obtained specific paths for transforming other evolutionary stable states (ESS) into ideal ESS through sensitivity analysis. Yu et al. (2021) employed evolutionary game to examine enduring impacts of information sharing on the adoption of organic farming by producers. Guo and Wei (2021) studied the symmetric information sharing issue of agricultural goods using evolutionary games. Wang et al. (2019) observed the configuration trends of different manufacturing services based on evolutionary game theory. Han et al. (2022) combined evolutionary game with a Steinberg game and designed a dynamic hierarchical game. Du (2023) used a two-stage evolutionary game model to confirm the role of cross-network effects in the cooperation between e-merchants and sellers. Shi et al. (2022) combined evolutionary dynamics with Q-Learning and verified that continuous evolutionary dynamics with a filtering mechanism can well predict the learning process of algorithms such as discrete Q-Learning.

Analyzing the incentive problem of data sharing employing evolutionary game theory requires treating all enterprises as a uniformly mixed population. Uniform mixing implies that any two enterprises have an opportunity to engage in the game, and each enterprise engages in a game with other enterprises with equal probability. However, in the manufacturing industry, it is often the case that not all pairs of enterprises can engage in the game; instead, each enterprise can only engage in the game with certain specific enterprises. This type of population is considered a non-uniformly mixed population. Considering that the population of enterprises is a non-uniform mixed population, we introduce a network (Barzinpour & Ahmadi, 2013) to simulate the connections between enterprises and employ networked evolutionary games to study incentive problems of enterprises' participation in data sharing. Networked Evolutionary Game is a novel interdisciplinary research field that combines complex networks and evolutionary game theory. It explores the strategy evolution behavior of social, biological and other groups through the formation and evolution of strategies on network groups. The following is the current state of research on networked evolutionary games to solve the data sharing problem: Zhang and Wu (2021) combined evolutionary games with complex networks to analyze the behavior of cooperation and betrayal in blockchain. Jiang et al. (2017) used homogeneous models based on complex evolutionary game models and heterogeneous models based on reinforcement learning to study the credibility of user information sharing strategies and utility related information. Gui et al. (2022) he investigated Nash equilibrium and consensus on strategies in an evolutionary networked game using heterogeneous updating rules. Cheng et al. (2013) compared stable strategies in infinite popular evolutionary games and networked games and proposed a new definition in the stable strategy for network evolutionary games. Xia et al. (2020) employed an evolutionary game to examine the cooperation problem of intelligent attackers in the network.

Q-learning is currently used in industry (Fu et al., 2023). Q-Learning is also applied in the setting of incentives. In the incentive mechanism for sensing data sharing in IoT, Liu et al. (2020) used Q-Learning methods to get the best perceptual contribution of participants. Chen et al. (2022a) addressed the issues of privacy protection and user engagement among Internet of Things (IoT) users and designed an incentive method based on Q-Learning, which aims to reduce the flow offset cost and enhance user enthusiasm. Yu et al. (2015) created an incentive contracts menu to address the issue of electricity pricing and utilized Q-Learning to evaluate the response of power generation companies to the incentive menu.

In previous studies, few scholars have considered the game relationship between enterprises in data sharing, and the incentive mechanisms given are relatively fixed. However, the networked evolutionary game is more suitable for the game relationship between enterprises in the real situation. Therefore, this study combines the incentive mechanism with networked evolutionary games and designs DIMQL to address the incentive issue of data sharing.

The primary contributions of this study are outlined blew:

1. An incentive model for enterprise data sharing is established and an evolutionary game is employed to study the evolution of strategies for enterprises participating in data sharing under an incentive mechanism. The experimental results suggest that the incentive value that can make enterprises stably share data is the reasonable incentive value.
2. Considering the specific relationships among enterprises, networked evolutionary games are employed to examine the incentive problem of data sharing. To balance the individual benefits and group benefits among enterprises, a strategy update rule based on DPSO and VND algorithm is proposed.
3. To address the challenge of determining rational incentive values in network evolutionary games, this study proposes the DIMQL. The experiments demonstrate that the method effectively promotes enterprise participation in data sharing.

The remaining sections of this paper are structured as follows. Section 3 introduces an incentive model for enterprises' participation in data sharing and studies the data sharing incentive issue using evolutionary game theory. Section 4 introduces a scale-free network to simulate the connections between enterprises. Then, a strategy update rule based on DPSO and VND is designed, and the incentive problem of data sharing is analyzed using networked evolutionary games. Section 5 proposes a DIMQL to address the challenge of determining reasonable incentive values in networked evolutionary games. Section 6 performs numerical experiments to validate the equilibrium of the evolutionary game and networked evolutionary game under the data sharing incentive model. Additionally, the feasibility of the proposed DIMQL is verified through experiments. Section 7 concludes the study and suggests further research.

3. Evolutionary Game for Enterprises Sharing Data

In this section, we will explore the evolution process of firms' participation in data sharing under incentives based on evolutionary games. We will build a data sharing incentive model for enterprises. In each game, enterprises will continuously adjust their strategies based on the associated benefits, eventually leading to a dynamic equilibrium for all enterprises' strategies. Among all possible evolutionary equilibria, our objective is to identify an equilibrium where all enterprises engage in data sharing, while ensuring the incentive value remains reasonable. We assume that all enterprises are limited rational and that they form a uniformly mixed population. This implies that game behavior may occur between any two enterprises, and the probability of its occurrence is equal.

3.1 Model Building

The data sharing incentive model mainly consists of four elements, which can be represented as $G=(x,N,K,U)$. The meaning of the relevant symbols in the model is shown in Table 1.

Table 1
Meaning of symbols in the data-sharing incentive model

Symbol	Meaning
x	The proportion of enterprises sharing data
N	Number of enterprises
K	Strategy Space
U	Two-player and two-strategy payment matrix

3.1.1 Basic Hypothesis

Some hypotheses are formulated to create the data sharing incentive model.

Hypothesis 1: In a two-player and two-strategy game, each enterprise has two alternative strategies. K = {K_1, K_2} = {share data, not share data} represents the strategies space.

Hypothesis 2: The data itself has value. Regardless of whether the enterprise participates in data sharing or not, it will derive benefits. The enterprise owns a quantity of shared data, denoted as C. The value of data is expressed by $\ln(1+C)$.

Hypothesis 3: Uploading data to the blockchain and maintaining the blockchain system afterward will incur time and technical costs for enterprises. This part of cost is considered as data sharing cost. In this paper, m represents cost coefficient, and mC represents the cost of sharing data.

Hypothesis 4: Data sharing among enterprises facilitates the improvement of products and services, as well as effective data exploitation. If both firms engage in share, they mutually open their data, resulting in shared benefits for both parties. k represents the benefit coefficient of data sharing, and $k\ln(1+C)$ represents the benefit of data sharing.

Hypothesis 5: To incentivize enterprises to voluntarily share data, the blockchain platform rewards enterprises that share data and penalizes those that do not share data. Both rewards and penalties are given to enterprises in the form of points. In this study, $\alpha\left(0<\alpha<1\right)$ represents the reward incentive coefficient for data sharing, and αC represents the reward for participating in data sharing; $\beta\left(0<\beta<1\right)$ represents the punishment coefficient for not sharing data, and βC represents the penalty for not participating in data sharing.

3.1.2 Payoff Matrix

The evolutionary game of data sharing incentive problem is a symmetric game. According to the above hypotheses, in each round of the game, the possible benefits obtained by the enterprises can be expressed as:

Table 2
Payoff matrix of the evolutionary game

		P_2	
		K_1	K_2
P_2	K_1	$\ln(1+C)+k\ln(1+C)+\alpha C-mC$	$\ln(1+C)+\alpha C-mC$
		$\ln(1+C)+k\ln(1+C)+\alpha C-mC$	$\ln(1+C)-\beta C$
	K_2	$\ln(1+C)-\beta C$	$\ln(1+C)$
		$\ln(1+C)+\alpha C-mC$	$\ln(1+C)$

Following analysis is conducted for each case in the payoff matrix:

a. Both players choose to share data (K_1). In this case, companies not only benefit from data sharing but also receive rewards from the blockchain platform. In addition, each enterprise must pay amount of sharing cost when actively sharing data. Therefore, when both players choose K_1, they will receive a benefit of $\ln(1+C)+k\ln(1+C)+\alpha C-mC$.

b. One player chooses to share their data (K_1), while the other chooses not to share data (K_2). In this case, the enterprise participating in data sharing shares its data but does not receive the desired data from the non-participating enterprise, thus missing out on the benefits of data sharing. The enterprise that does not upload data also cannot get valuable data from the blockchain platform. Therefore, the benefit for the participating enterprise is $\ln(1+C)+\alpha C-mC$, while the benefit for the non-participating enterprise is $\ln(1+C)-\beta C$.

c. Both players choose not to share data (K_2). In this case, none of the enterprises on the blockchain platform are willing to share data, and each enterprise's benefit solely depends on its own data value, which is represented by $\ln(1+C)$.

3.2 ESSs of Enterprises Sharing Data

The ideal state of blockchain data sharing is to attract all enterprises to participate. In the evolutionary game, enterprises keep learning and imitate other enterprises' strategies in the evolution process to finally reach an optimal equilibrium and maximize the overall benefit. We divide time into 1, 2, …, t, …, and suppose that the possibility of enterprises sharing data at the stage t is $x(t)$. The probability $x(t)$ is equal to the ratio of firms involved in data sharing. Similarly, the probability that a firm does not participate in data sharing at stage t is $1-x(t)$. The probability $1-x(t)$ also represents the ratio of firms not involved in data sharing. The expected benefit of enterprises sharing data can be calculated using Eq. (1):

$$U_t^1=x(t)\left[\ln(1+C)+k\ln(1+C)+\alpha C-mC\right]+\left[1-x(t)\right]\left[\ln(1+C)+\alpha C-mC\right] \tag{1}$$

The expected benefit for firms that decide not to share data is Eq. (2):

$$U_t^2=x(t)\left[\ln(1+C)-\beta C\right]+\left[1-x(t)\right]\ln(1+C) \tag{2}$$

Average expected benefit of firms sharing data can be expresses as Eq. (3):

$$\bar{U}_t = x(t)U_t^1 + \left[1 - x(t)\right]U_t^2 \tag{3}$$

The replication dynamic equation (Fudenberg et al., 1998) in the evolutionary game is Eq. (4):

$$F\left(x(t)\right) = \frac{dx(t)}{dt} = x(t)\left(U_t^1 - \bar{U}_t\right) \tag{4}$$

Define $F\left(x(t)\right)$ as 0, three stable states can be derived as:

$$x_a^* = 0, x_b^* = 1, x_c^* = \frac{C(m-\alpha)}{k\ln(1+C)+\beta C}$$

If x^* is the stable state, it must exist $F'\left(x^*\right) < 0$ (Friedman, 1991). Three ESSs can be identified.

Case 1: If $m \le \alpha$, $x_b^* = 1$ is the ESS in which enterprises choose to share data, and $x_a^* = 0$ is not an ESS. $x_c^* = \frac{C(m-\alpha)}{k\ln(1+C)+\beta C}$ doesn't exist.

Case 1 suggests that regardless of which strategy an enterprise initially chooses, they will eventually engage in data sharing.

Case 2: If $m > \alpha$ and $C(m-\alpha) - k\ln(1+C) - \beta C < 0$, then $x_a^* = 0$ and $x_b^* = 1$ are the ESSs for enterprises to share data, while $x_c^* = \frac{C(m-\alpha)}{k\ln(1+C)+\beta C}$ is not an ESS. When $x \in \left(0, \frac{C(m-\alpha)}{k\ln(1+C)+\beta C}\right)$ exists, no enterprises will be willing to share data. When $x \in \left(\frac{C(m-\alpha)}{k\ln(1+C)+\beta C}, 1\right)$ exists, firms will select to share data ultimately.

Case 2 indicates that the final strategy choice of enterprises is influenced by the initial share of enterprises taking part in data sharing, and the benefits that the enterprises obtain from the blockchain system force the enterprise's strategy to eventually move to a different stable state.

Case 3: If $m > \alpha$ and $C(m-\alpha) - k\ln(1+C) - \beta C \ge 0$, then $x_a^* = 0$ is an ESS, while $x_b^* = 1$ is not an ESS. $x_c^* = \frac{C(m-\alpha)}{k\ln(1+C)+\beta C}$ does not exist.

Case 3 suggests that all enterprises will decide not to upload data regardless of the initial sharing strategy group.

Based on the above inferences, it is evident that ESS is not unique for enterprises sharing data. This depends on the initial proportion of enterprises uploading data and the benefits for enterprises. The reasonable incentive value is the one that can ensure the stable participation of all enterprises in data sharing. The reasonable incentive values can be derived through evolutionary game.

4. Networked Evolutionary Game for Enterprises Sharing Data

In the above research, we utilized evolutionary games to investigate the process of strategies evolution for enterprise sharing data within a uniformly mixed population. However, the composition of subordinate enterprises in manufacturing industries is complex. In the real market, enterprises engage in direct or indirect interactions and competition with each other through technology, market information, and capital flows, leading to the occurrence of strategic games. Therefore, the game in the manufacturing industry takes place within a non-uniformly mixed population where each enterprise can engage in the game with other enterprises that share cooperative or competitive relationships.

Taking into account this characteristic, we introduce a scale-free network to simulate the connections between enterprises and design a networked evolutionary game that combines DPSO with VND to analyze the issue of incentivizing data sharing among enterprises.

The networked evolutionary game model for blockchain data sharing in the manufacturing industry comprises three components: the game model between enterprises, the network structure, and the evolution update rules. The income calculation for the data sharing incentive model is provided in Table 2. In the subsequent sections, we will elucidate the network structure and evolution update rules.

4.1 Network Modeling of Enterprise Relationships

In this paper, we utilize a topology structure $G = (N, E)$ to describe the interconnections among enterprises. Here,

$N = (n_1, n_2, ..., n_n)$ represents a set of nodes, with each node representing an individual enterprise. Additionally, there is a collection of relationships denoted by *E*, where each line *e* represents a direct competition or cooperation between enterprises.

$$E = \begin{bmatrix} e_{11} & e_{12} & \cdots & e_{1n} \\ e_{21} & e_{22} & \cdots & e_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ e_{n1} & e_{n2} & \cdots & e_{nn} \end{bmatrix}$$

Here, e_{ij} represents the connection between n_i and n_j. When $e_{ij} = 1$ exists, it indicates a direct relationship of competition or cooperation between n_i and n_j; When $e_{ij} = 0$ exists, it indicates no direct relationship of competition or cooperation between n_i and n_j. The network structure is a simple undirected graph, and there is no scenario where an enterprise is connected to itself or has two connections on the same end.

Random, small-world, and scale-free network topologies are commonly used to portray complex systems such as supply chains (Bellamy & Basole, 2013). In the manufacturing industry, there are four main chains: supply chain, manufacturing chain, transportation chain, and sales chain. Each industry chain comprises different types of enterprises. Within an industry chain, although the enterprises share similar types, their scales may differ. Large-scale enterprises have a broader scope of operations, possess strong driving, and resource integration capabilities, and lead the growth of small and medium-sized firms within the same industry chain. They may also engage in business cooperation with enterprises from other industry chains. As a result, large-scale enterprises establish connections with a greater number of enterprises, while small-scale enterprises have limited business volume and production resources, resulting in fewer connections with other enterprises. Most common nodes in a scale-free network have just a few connections, while a minority of nodes possess many connections. This characteristic bears a striking resemblance to the connections among enterprises in a supply chain. Therefore, we choose to utilize a scale-free network to simulate the connections between enterprises.

In this paper, we employ a step-by-step growth model to generate a scale-free network. In this model, the network is generated through the following steps:

1. Initial network: At the initial stage, we generate a network containing *n* nodes. The initial network is a fully connected network.

2. Node addition:
a. Generate a new node each time.
b. Connect the newly added node to the already existing nodes. The target node for connection is selected based on the degrees of the existing nodes. The probability of selecting a node increases with its degree. This probability d_i can be expressed as $\Pi(d_i) = \dfrac{d_i}{\sum_i d_i}$.

4.2 Strategy Update Rules for Data Sharing Based on Networked Evolutionary Game

Particle Swarm Optimization algorithm (PSO) uses the information shared by individuals to drive the population to gradually evolve in the problem solution space and eventually obtain the optimal solution to the problem. PSO has characteristics, including agents' propensity to remember their own beliefs while taking into account the beliefs of their peers, and cognitive and social behaviors that seek consistency in cognition. They will adjust if they realize that the beliefs of their peers are more superior. Regarding the issue of data sharing incentives, at the end of each round of the game, enterprises compare their own revenue with that of their own neighbors and imitate the strategy of the highest revenue earner among their neighbors. This behavioral characteristic is very similar to that of PSO, so we choose PSO to analyze the networked evolutionary game. Solution of a Networked Evolutionary Game based on DPSO. In the paper, we map the elements of the game to PSO:

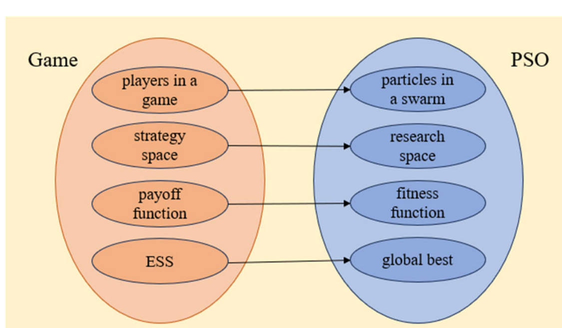

Fig. 1. The Mapping Relationship between Game and PSO

In the strategies space, enterprises have only two alternative strategies: share data or not share data. Therefore, the problem can be seen as a discrete type of particle swarm optimization (DPSO).

Suppose there are N enterprises in total, and the strategy selection for enterprise i is represented by $X_i(X_i = 0 \ or \ 1)$. The set of strategy choices for all enterprises is $X = (X_1, X_2, ..., X_i, ..., X_N), 1 \le i \le N$.

The updated equations for the velocity and position of a standard particle are given by Eq. (5):

$$
\begin{aligned}
&v_i^{t+1} = \omega v_i^t + c_1 r_1 \left(pbest_i^t - X_i^t\right) + c_2 r_2 \left(gbest_i^t - X_i^t\right) \\
&s(v_i^{t+1}) = \frac{1}{1 + e^{-v_{id}}} \\
&X(v_{id})^{t+1} = \begin{cases} 1, & rand() < s(v_i^{t+1}) \\ 0, & otherwise \end{cases}
\end{aligned}
\tag{5}
$$

The PSO continuously adjusts the velocity and position of the particle during its flight, aiming to bring the particle closer and closer to the optimal position while gradually converging its velocity to 0. When $v = 0$ exists, it is likely that $pbest_i = X_i$, $gbest_i = X_i$, indicating that the current position is the historical optimal position and does not require further changes. However, in traditional DPSO, when $s = 0.5$, $v = 0$ as well, which implies that as the particle approaches the optimal position, its likelihood of variation increases, resulting in weak local search capability. Therefore, we opt for an updated DPSO formula:

$$
\begin{aligned}
&v_i^{t+1} = \omega v_i^t + c_1 r_1 \left(pbest_i^t - X_i^t\right) + c_2 r_2 \left(gbest_i^t - X_i^t\right) \\
&s(v_i^{t+1}) = \begin{cases} 1 - \dfrac{2}{1 + e^{-v_{id}}}, v_i \le 0 \\ \dfrac{2}{1 + e^{-v_{id}}} - 1, v_i > 0 \end{cases} \\
&when \quad v < 0 \\
&\qquad X_i = \begin{cases} 0, rand() \le s(v_i^{t+1}) \\ X_i, otherwise \end{cases} \\
&when \quad v > 0 \\
&\qquad X_i = \begin{cases} 1, rand() \le s(v_i^{t+1}) \\ X_i, otherwise \end{cases}
\end{aligned}
\tag{6}
$$

In Eq. (6), if the velocity is close to 0, the position may not require immediate changes, and thus the probability of position change is low. Conversely, if the velocity is far from 0, the position may need to change, and therefore the probability of position change is higher.

In the networked evolutionary game, the chosen strategy update rule in this paper is that the player imitates the optimal strategy of its neighbors. In each round of the game, the enterprise selects several enterprises from its neighbors to participate in the game, and the payoff for the player in this round is the average revenue of these games. The enterprise imitates the strategy of its neighbors with the highest revenue and considers it as the individual optimal strategy $pbest_{\mathrm{i}}$. After one round of the game for all enterprises, if the total revenue is higher than that of the previous round, the current set of strategies becomes the group optimal strategy $gbest_{\mathrm{i}}$; Otherwise, the previous set of strategies remains as the group optimal strategy.

We then update the strategy using DPSO. In Eq. (5) and Eq. (6), the flight velocity of each particle v_i is determined by $pbest_{\mathrm{i}}$ and $gbest_{\mathrm{i}}$, and these velocities have an impact on the equilibrium of the game. $pbest_{\mathrm{i}}$ and $gbest_{\mathrm{i}}$ are determined by the enterprises' payoffs. This implies that the equilibrium of the game in the current environment is not purely about individual payoff maximization or group payoff maximization. Instead, it represents a balance between individual enterprise payoff and group payoff. This balance allows enterprises to optimize their individual interests while also considering the maximum interests of the entire group.

The algorithmic flow of the game based on PSO is as follows:

Algorithm 1: Networked Evolutionary Game Based on DPSO

Input: x_0, C, k, α, β, m, $\theta = 0.4$, ω, c_1, c_2,
Output: X

```
Generate a network G = (N, E)
Generate the initial solution according to x_0
while (iteration<Max iteration) do
        The player determines pbest_i and gbest_i by playing games with its neighbors
    If (iteration< θ *Max iteration) do
        for (i<N) do
            Update v[i], s[i], X[i] using Equation (5)
        end for
    end if
  else do
      for (iteration< θ *Max iteration) do
          for (i<N) do
                Update v[i], s[i], X[i] using Equation (6)
            end for
        end for
    end else
end while
```

4.3 Improvement of Networked Evolutionary Game Based on DPSO

The randomness of intelligent algorithms results in fluctuations in the evolutionary trajectory. In this section, we have incorporated VND into the networked evolutionary game based on PSO to alleviate the strategy fluctuations of enterprises in the early stages of evolution.

We have designed three variable neighborhood operators that manipulate the strategy set of enterprises. The operators are as follows:

(1) The operator selects a segment within the solution set and reverses the strategies, transforming 0 into 1 and 1 into 0.
(2) This operator randomly selects two segments of solutions from the solution set of enterprises and swaps them. Each segment occupies one-fifth of the total length of the solution.
(3) The operator selects a segment within the solution set and performs a reversal of the solution from the beginning to the end

Then, we add the designed operators to the networked evolutionary game, and the algorithm process is as follows:

Algorithm 2: Networked Evolutionary Game Algorithm Based on DPSO with VND

Input: x_0, C, k, α, β, m, θ, $\theta = 0.4$, ω, c_1, c_2
Output: X

```
Generate a network G = (N, E)
Generate the initial solution according to x_0
while (iteration<Max iteration) do
    The player determines pbest_i and gbest_i by playing games with its neighbors
    If (iteration< θ *Max iteration) do
        for (i<N) do
            Update v[i], s[i], X[i] using Equation (5)
        end for
         Perform VND and update the solution if there is a solution that maximizes the revenue of population. Otherwise, keep the original solution.
    end If
    else do
        for (iteration< θ *Max iteration) do
            for (i<N) do
                Update v[i], s[i], X[i] using Equation (6)
            end for
        end for
    end else
end while
```

5. A Dynamic Incentive Mechanism for Promoting Data Sharing in Networked Evolutionary Game

Based on the experiment, we can observe that in certain situations, the incentive values we have set may not encourage data sharing among enterprises. However, to meet the requirements of the blockchain traceability system, all enterprises must share their relevant production, processing, transportation, and sales data. Therefore, this paper proposes a dynamic incentive method based on Q-Learning. In cases where the given incentive values fail to promote data sharing among enterprises, the method dynamically adjusts the incentive values and the sharing cost of enterprises, ultimately ensuring that all enterprises share data.

In the data sharing incentive model, the profits of enterprises are influenced by the amount of data shared, the sharing benefit coefficient, the reward incentive coefficient, the punishment incentive coefficient, and the sharing cost coefficient. Among these factors, the amount of data uploaded by enterprises is difficult to modify due to traceability system requirements. The sharing benefit of enterprises relies on cooperation with other enterprises to explore the additional value of the data, making it challenging for human intervention. The sharing cost of enterprises is associated with the blockchain platform, while the rewards and penalties received by enterprises are directly set by the blockchain platform. When incentivizing enterprises, we can simultaneously regulate the sharing costs and the rewards and penalties obtained by enterprises. The Q-Learning algorithm is a reinforcement learning algorithm known for its high flexibility and adaptability. When adjusting the model parameters, it is challenging to achieve the ideal scenario of all enterprises sharing data through a few adjustments. Therefore, selecting the Q-Learning algorithm to transform parameter combinations allows for dynamic adjustments to the enterprise's earnings. This approach enables the enterprise to gradually find acceptable incentive values and shared costs during the adjustment process, transforming strategies from non-participation to participation in data sharing.

In the following, we will provide detailed explanations of the adjustment process for these three parameters. The elements used in the Q-Learning algorithm include states, actions, agents, rewards, and penalties. In this paper, we set several parameter combinations (denoted as $s=(\alpha,\beta,m)$) for reward incentive coefficients $\alpha=\{\alpha_0,...,\alpha_i,...,\alpha_f\}$, penalty incentive coefficients $\beta=\{\beta_0,...,\beta_i,...,\beta_f\}$, and shared cost coefficients $m=\{m_0,...,m_i,...,m_f\}$, which represent the states of the Q-table. The actions in the Q-table correspond to changes in states, with the constraint that each action modifies only one parameter value at a time and the size of this parameter value change is $step$. Assuming that the current action is to reduce parameter β of state (α_3,β_2,m_5) , then the next state will be (α_3,β_1,m_5) , where $\beta_1=\beta_2-step$. The agent refers to the enterprise. We define the rewards and penalties R in Q-Learning as the difference between the earnings after each state change and the earnings before the change. A positive difference represents the reward obtained by the enterprise in Q-Learning, while a negative difference represents the penalty obtained by the enterprise.

Given that the Q-Learning algorithm adapts the values of three parameters: the reward incentive coefficient, the punishment incentive coefficient, and the shared cost coefficient, the Q-table in this context is thus three-dimensional. All Q-values of Q-table are initialized as 0, and they are updated according to the following formula:

$$Q_{[\alpha][\beta][m][i]}=(1-l)Q_{[\alpha][\beta][m][i]}+l(R_i+\gamma Q_{\max}) \tag{7}$$

Here, l represents the learning rate, and γ denotes the decay factor for rewards.

Moreover, in order to avoid situations where some actions are not explored when searching for the optimal state, we have incorporated a greedy strategy into the Q-table updates.

The process of the algorithm is illustrated in the table below:

Algorithm 3: DIMQL

Input: s_0 , f , $step$, greedy
Output: X

```
Initiate all Q-values to 0
Determine the current state s0
while (iteration<Max iteration)
    if (first||rand () <greedy) do
        Select a random action a within the available in the current state
        Perform action a, get reward or punishment R
        Update the Q-table using Equation (7)
        Move to the new state s'
        Execute Algorithm2(new state)
    end if
    else do
        According to the maximum Q-value in the subsequent state, choose action a from those readily available in the current state
        Perform action a, get reward or punishment R
        Update the Q-table using Equation (7)
        Move to the new state s'
        Execute Algorithm2(new state)
    end else
end while
```

6. Experimental results

In this section, we design numerical experiments which are described as follows.

6.1 Experiments on Data Sharing among Enterprises Based on Evolutionary Game

Considering the possibility of a fixed limit on the quantity of data that businesses can upload to the blockchain platform in reality, the experiments in this section assume that the blockchain has a fixed amount of data with the same data value C. Additionally, we consider the shared benefits of enterprises to be fixed, that is, k is fixed. In this part, the impact of various parameters on ESSs for a specific incentive mechanism is verified by setting different parameters x, α, β and m. There are 100 enterprises involved in the game. The parameter settings and evolution process are as follows:

Table 3
Verification parameters of the evolutionary game

Test items	x_0	C	k	α	β	m
Test 1	0.5	10	0.1	0.2	0.15	0.15
Test 2	0.25	10	0.1	0.2	0.15	0.25
Test 3	0.45	10	0.1	0.2	0.15	0.25
Test 4	0.8	10	0.1	0.15	0.07	0.25

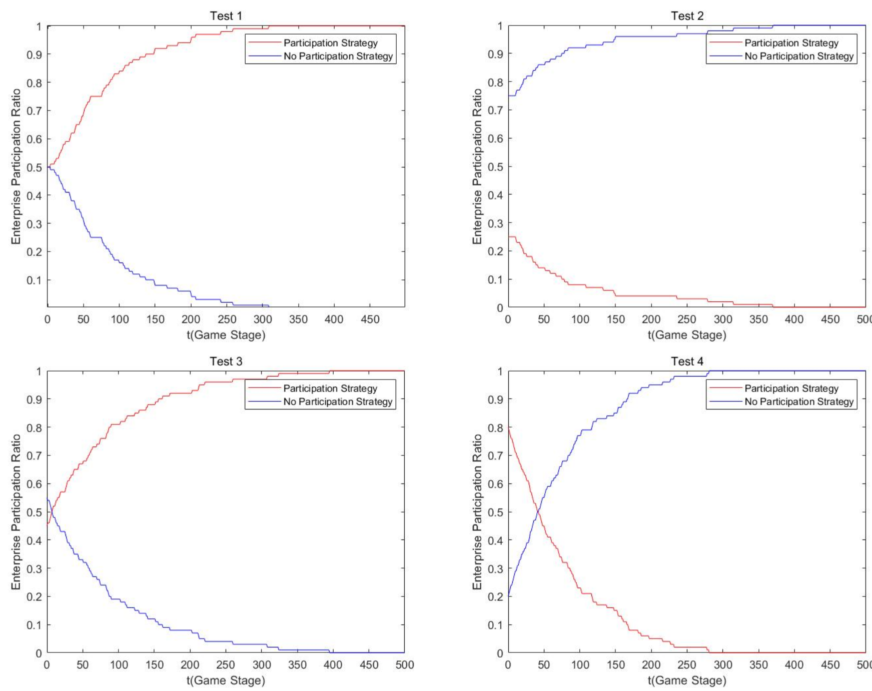

Fig. 2. Data sharing incentive model evolutionary game curve

When employing evolutionary game theory to examine the incentive mechanism of data sharing, the ESSs can be calculated by formulas. Hence, when blockchain platforms establish incentives, they can refer to the ESSs and select a set of incentive values that can enable enterprises to ultimately achieve the equilibrium of complete data sharing.

6.2 Experiments on Data Sharing among Enterprises Based on Networked Evolutionary Game

There are two types of experiments in this section, one compares the evolution of the network evolution game with the evolution game under the same model parameters, and the other compares the evolution of the network evolution game before and after adding VND.

6.2.1 Selection of DPSO parameters

Taguchi experiments are widely used for parameter tuning of PSO algorithms (Dey et al., 2017). The DPSO algorithm contains three key parameters: weight coefficients ω and acceleration factors c_1, c_2. In order to investigate the effect of these parameters on the algorithm's effectiveness, we implemented the Taguchi method of experimental design. Before implementing the Taguchi experimental design, the network architecture was structured as detailed below:

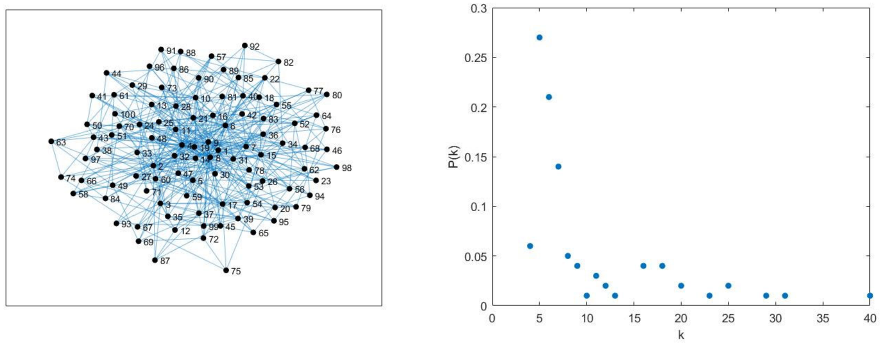

Fig. 3. The network structure and degree distribution graph

The model parameters are as follows:

Table 4
Model parameters of the networked evolutionary game

Model Parameters	x_0	C	k	α	β	m
Value	0.5	10	0.1	0.2	0.15	0.15

These parameters are set primarily to calculate the payoffs in the game. In the Taguchi experiment, four levels are set for each parameter, as shown in Table 5. The orthogonal array L_{16} (4^3) is generated according to the number of parameters and levels. Thus, there are a total of 16 combinations of parameter values. For each combination, we run DPSO independently 10 times and recorded the average optimal gain.

Table 5
Parameters values for each factor level

Parameters	Factor level			
	1	2	3	4
ω	0.4	0.6	0.8	1.0
c_1	0.5	1.0	1.5	2.0
c_2	0.5	1.0	1.5	2.0

In order to assess the performance of the combinations of parameter values, we used the average return as a metric. The average return is the mean of the total returns from the game obtained across multiple simulations using a certain set of parameter values. The orthogonal array and average returns are displayed in Table 6. The effective ranks of each parameter are presented in Table 7. Then, a plot depicting the average values of each factor level in the DPSO algorithm is generated and presented in Fig. 4.

Table 6
Orthogonal array and average return

Number	Factor level			Average Returns
	ω	c_1	c_2	
1	1	1	1	212.4479
2	1	2	2	213.7410
3	1	3	3	214.9403
4	1	4	4	215.9045
5	2	1	2	213.0815
6	2	2	1	214.9820
7	2	3	4	216.4756
8	2	4	3	217.8865
9	3	1	3	214.9575
10	3	2	4	218.1973
11	3	3	1	220.4943
12	3	4	2	222.2196
13	4	1	4	233.8446
14	4	2	3	236.1011
15	4	3	2	237.0796
16	4	4	1	**237.5760**

Table 7
Average returns and rank of each parameter

Level	DPSO		
	ω	c_1	c_2
1	214.2584	218.5829	221.3751
2	215.6064	220.7554	221.5304
3	218.9672	222.2475	220.9714
4	236.1503	223.3967	221.1055
Delta	21.8919	4.8138	0.5590
Rank	1	2	3

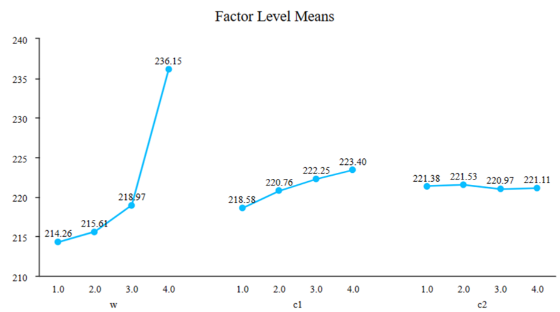

Fig. 4. Factor Level Means

As can be seen from Table 6 and Table 7, the weighting factor ω and the acceleration factor c_1 are the most significant parameters, with c_2 being relatively less significant. Therefore, based on the returns in Table 7, the suggested parameters for DPSO in order to obtain higher returns in the game are $\omega = 1, c_1 = 2, c_2 = 1$.

6.2.2 Comparison of Evolution Processes between Networked Evolutionary Game and Evolutionary Game

First, we generate a network to simulate the connections between enterprises. In this experiment, we set a total of 100 enterprises, with 5 enterprises initially fully connected, and the network is generated according to the method described in Section 4.1. The network connectivity diagram and degree distribution are shown below:

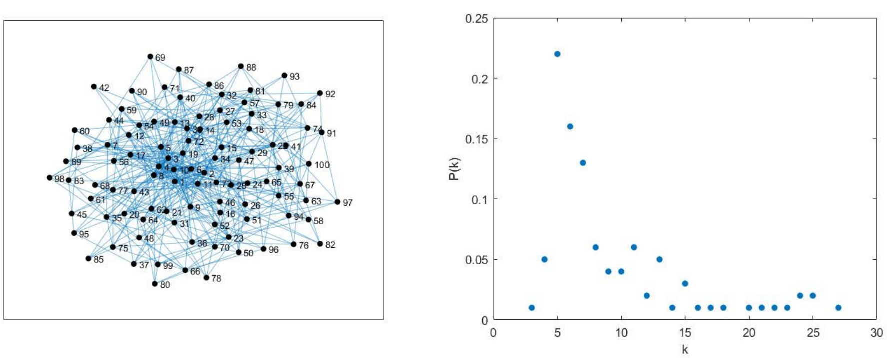

Fig. 5. The network structure and degree distribution graph

In Eq. (5) and Eq. (6), we take $\omega = 1$, $c_1 = 2$ and $c_2 = 1$. Then, the parameters of the data sharing incentive model are set as below:

Table 8
Verification parameters of the networked evolutionary game

Test items	x_0	C	k	α	β	m
Test 1	0.5	10	0.1	0.2	0.15	0.15
Test 2	0.25	10	0.1	0.2	0.15	0.25
Test 3	0.45	10	0.1	0.2	0.15	0.25
Test 4	0.8	10	0.1	0.15	0.08	0.25
Test 5	0.9	10	0.1	0.15	0.1	0.25

The experimental results are shown in the figure below:

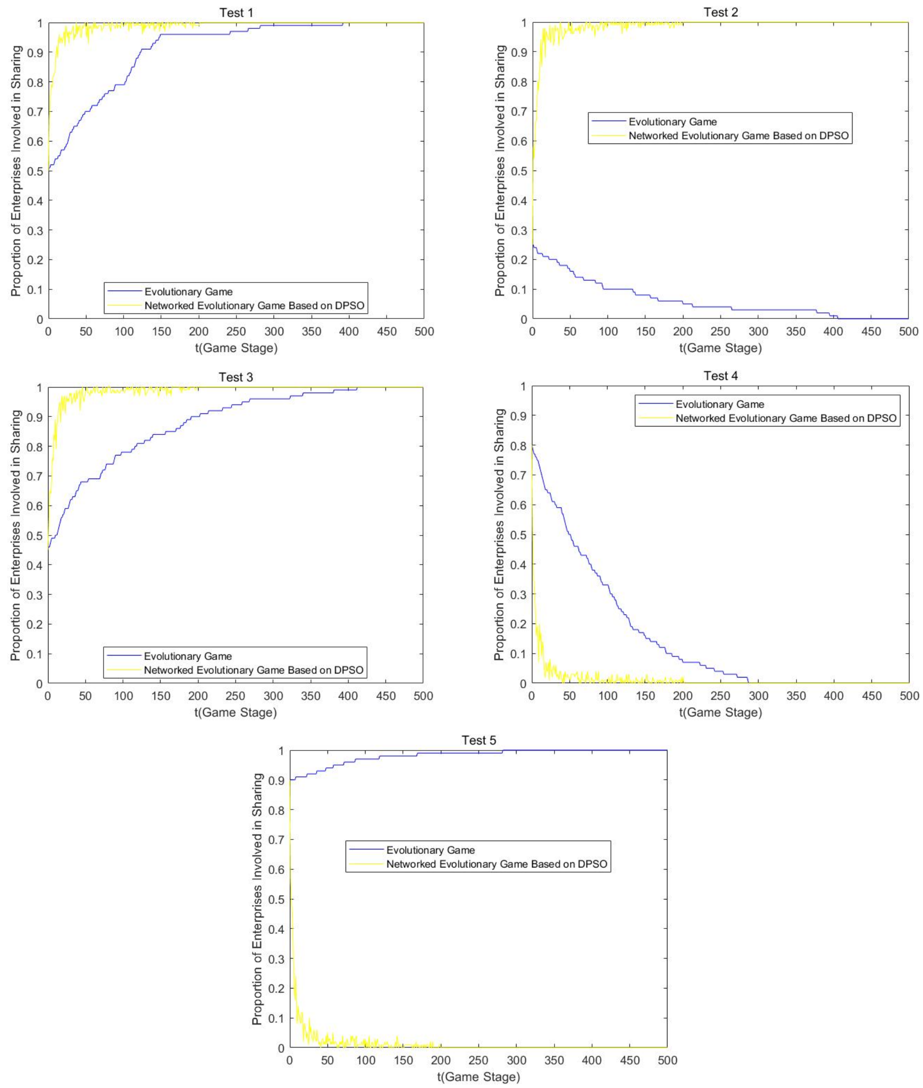

Fig. 6. Comparative curve of evolutionary game method and networked evolutionary game method under data sharing incentive model

Based on experimental results, we can see that there are significant differences between the evolutionary processes of networked evolutionary games and traditional evolutionary games. We describe the equilibriums of both as (networked evolutionary game equilibrium, evolutionary game equilibrium), which can result in four possibilities: (0,0), (0,1), (1,0), and (1,1). Due to the presence of the network and different evolutionary rules, the equilibriums of the two can be different. The conditions for using evolutionary games are relatively idealized, as they require that all companies can participate in the game and that the probability of game occurrence between any two companies is equal. However, in reality, the probability of game occurrence among companies may not be equal, and whether a game occurs is related to the network structure, that is, the connections between companies. Networked evolutionary games are more applicable to the real world. This means that in real life, the incentive values calculated using idealized evolutionary game methods may not necessarily be applicable.

We can also observe that the evolution curve of networked evolutionary games exhibits fluctuations in the early stages, which are attributed to the randomness in DPSO and the local information propagation in the game. Intelligent algorithms have randomness. The random selection in DPSO will affect the strategy changes of participants, resulting in minor changes in strategy. Furthermore, in a network setting, players can only engage in games with their directly connected counterparts and learn and propagate strategies with limited information, which also has an impact on the evolutionary process. In the later stages of evolution, the curve stabilizes at either 0 or 1. This is because when both local and global strategies become optimal, the probability of changes in the improved DPSO approaches zero. Due to the existence of networks, networked evolutionary games reach equilibrium faster compared to evolutionary games.

In the initial stages of evolution in networked evolutionary games, there may be significant fluctuations in the evolution trajectory. To address this issue, we have made several improvements.

6.2.3 Comparison of evolution process before and after adding VND

We compared the evolutionary processes before and after the addition of VND under different network sizes. The parameter settings and results are as follows:

Table 9
Verification parameters

x_0	C	k	α	β	m
0.2	100	0.1	0.2	0.1	0.2

Test items	N	n
Test 1	100	5
Test 2	500	20
Test 3	1000	40

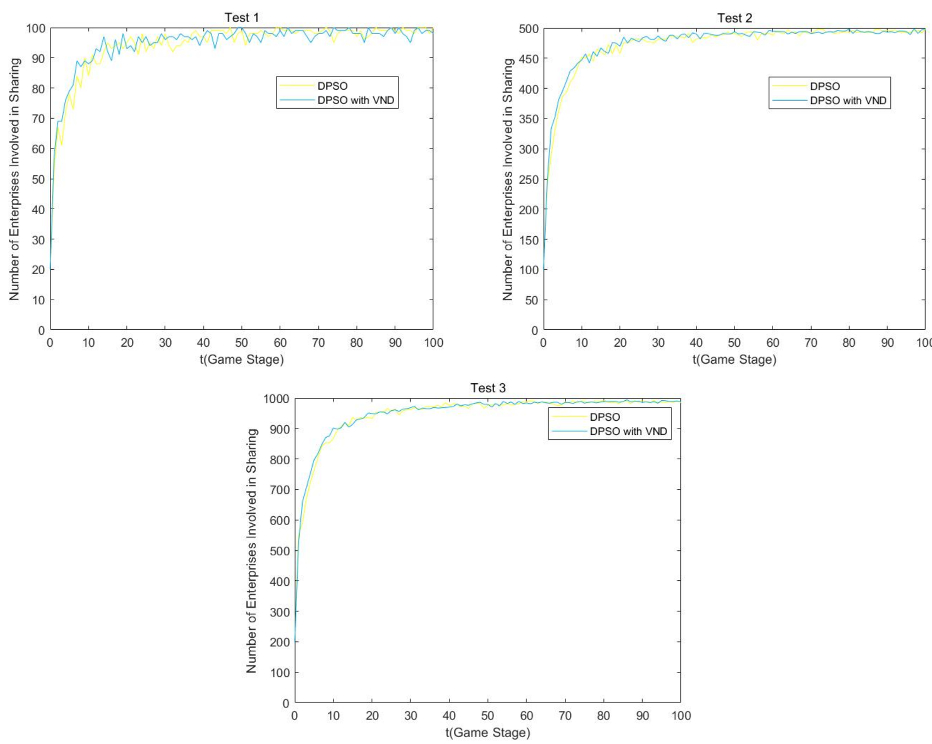

Fig. 7. Comparative curve of DPSO and DPSO with VND

We compared the evolution of the networked evolutionary game at different scales. From the experimental results, we can see that the DPSO with VND added evolves quicker at the beginning of evolution and less volatile in the later stage of evolution. This indicates that the VND is effective, and it finds a better solution than the current solution. However, due to the small size of the problem, the space for improvement of the DPSO algorithm is relatively small. So even with the addition of the VNS algorithm, the improvement of the algorithm is limited.

6.3 Experiments on DIMQL

6.3.1 DIMQL Based on Scale-free Networks

In this section, we employ the dynamic incentive method of enterprise data sharing based on Q-Learning to address the situation where several unreasonable reward settings led to the lack of participation of companies in data sharing. We compare the evolutionary process of the game with the inclusion of this method to that without inclusion. When l is close to 1, a higher priority is given to the newly gained information for the Q-table updates. When γ is close to 0, the Q-value is based on the current reward/punishment only. When γ is close to 1, the Q-value will be based on the current and the previous reward/punishment. To accelerate the learning speed and prioritize future rewards, we choose $l = 0.8$ and $\gamma = 0.8$ (Wang, 2023). In this experiment, we fix $C = 10$ and $k = 0.1$. The parameter settings and comparative results are shown in the following figure.

Table 10
Verification parameters of the networked evolutionary game

Test items	x_0	s_0	α_0	β_0	m_0	*step*	*f*
Test 1	0.8	(0.15, 0.08, 0.25)	0.05	0.05	0.15	0.05	5
Test 2	0.9	(0.15, 0.1, 0.25)	0.05	0.05	0.15	0.05	5
Test 3	0.2	(0.1, 0.1, 0.15)	0.05	0.05	0.15	0.05	5
Test 4	0.6	(0.2, 0.1, 0.3)	0.05	0.05	0.15	0.05	10

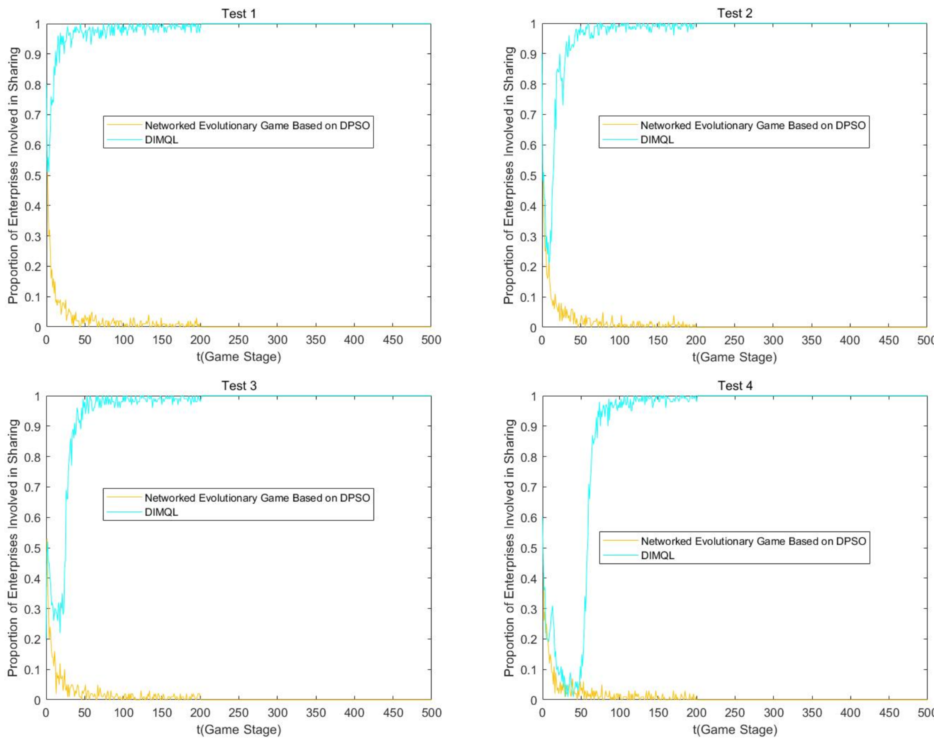

Fig. 8. Comparison chart after adding Q-Learning

From the experimental results, it is evident that the introduction of the dynamic incentive adjustment method based on Q-Learning has brought significant changes to the process of strategy evolution and equilibrium of the game.

The essence of this method lies in adjusting the enterprises' payoffs. Before the incorporation of this method, the benefits obtained from sharing data by enterprises were insufficient, making it difficult for them to maintain sharing. However, after incorporating this method, the payoffs of enterprises are dynamically adjusted. After a period of learning, the method will find suitable parameters that encourage enterprises to choose sharing in order to obtain higher payoffs, thereby altering the equilibrium of the game. In the early stages of implementing this method, due to the short learning period, appropriate parameters to incentivize sharing may not have been discovered yet, resulting in fluctuating patterns in the evolutionary curve. Additionally, these fluctuations are also related to the greedy strategy of Q-Learning and the randomness of the DPSO. In our algorithm, a greedy value is set, which makes it highly probable for individuals to move towards states with higher payoffs, but it does not guarantee that every move will be in that direction. This allows for better exploration of the Q-table to cover as many states as possible.

6.3.2 DIMQL Based on Other Networks

In the previous experiments, we modeled the connections between enterprises in a supply chain as scale-free networks. However, considering that the real connection between enterprises may not be a scale-free network, we verify the effectiveness of the DIMQL algorithm in small-world and random networks.

First, we establish a small-world network, as shown in Fig.6. This small-world network contains a total of 100 nodes, and each node is connected to each of its left and right neighbors, and then randomly reconnects each of the original edges in the network with a probability of 0.2.

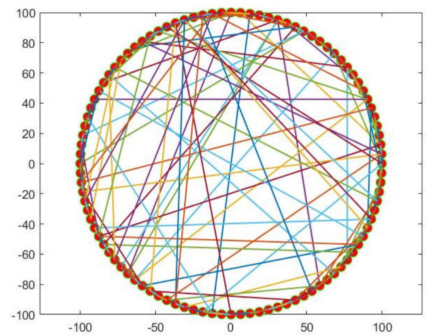

Fig. 9. Figure of small-world network

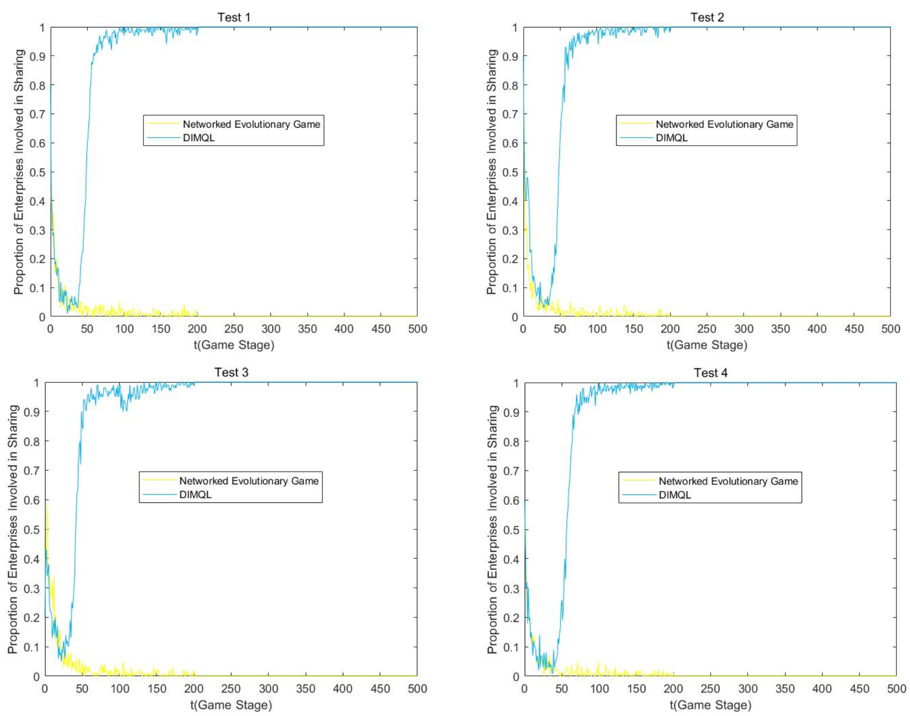

Fig. 10. Results of DIMQL in small-world network

We simulate the effect of DIMQL in a small-world network in these cases according to the numerical experiments in Table 10, as shown in Fig. 7. From the results, we can see that the DIMQL method is also effective in small-world networks. Then, we generate a random network containing 100 nodes as shown in Figure.7. Each pair of nodes is connected with a probability of 0.1.

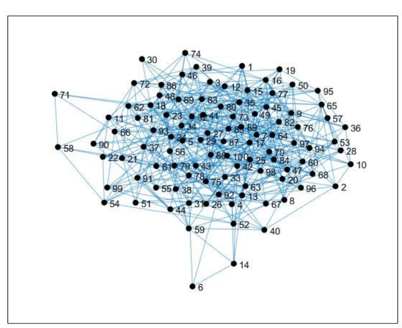

Fig. 11. Figure of random network

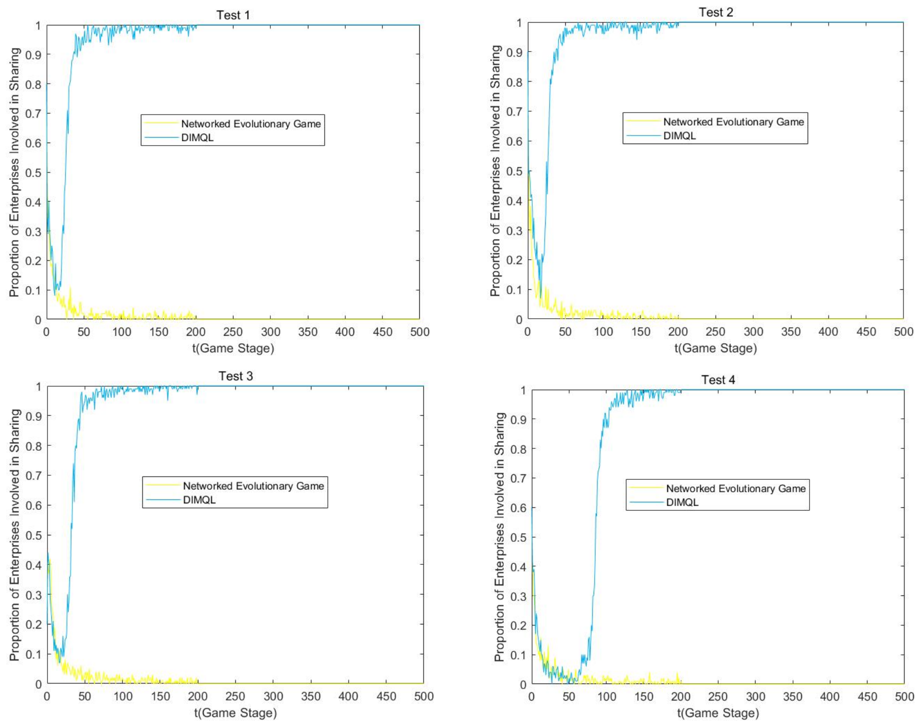

Fig. 12. Results of DIMQL in random network

We simulate the effect of DIMQL in a random network in these cases according to the numerical experiments in Table 10, as shown in Figure 8. From the results, we can see that the DIMQL method is also effective in random networks. By the performance of DIMQL in scale-free networks, small-world networks and random networks, we can see that the DIMQL algorithm is generalizable to a large number of networks.

7. Conclusion

In this paper, we investigate the incentive mechanism in blockchain data sharing. Following is a summary of the paper's primary research. Firstly, we propose an incentive model for enterprises to share data and investigate the process of strategy evolution in enterprise sharing participation using evolutionary games. Secondly, this study considers the specific connections between enterprises and designs a networked evolutionary game based on DPSO and VND to study the strategy evolution process of enterprises. Thirdly, to address the difficulty of determining reasonable incentive values in networked evolution games, DIMQL is designed. Finally, we simulate the evolution process of evolutionary game and networked evolutionary game and verify the effectiveness of the DIMQL.

There are still many issues that need to be investigated in our study. Firstly, incentive methods for enterprises to share data need to be further investigated. Secondly, the network relationship among enterprises can be optimized. Finally, the application of Q-Learning in incentives deserves further research.

Acknowledgments

This work is supported by the National Key R&D Program of China under Grant No. 2021YFB3300900.

References

Barzinpour, F & Ahmadi, B. (2013). A novel framework in complex network analysis: Considering both structure of relations and individual characteristics in closeness centrality computation. *International Journal of Industrial Engineering Computations, 4*(2), 227-240.

Bellamy, M. A., & Basole, R. C. (2013). Network analysis of supply chain systems: A systematic review and future research. *Systems Engineering, 16*(2), 235-249.

Chen, L., Zhang, D. G., Zhang, J., Zhang, T., Du, J. Y., & Fan, H. R. (2022a). An approach of flow compensation incentive based on Q-learning strategy for IoT user privacy protection. *AEU-International Journal of Electronics and Communications, 148*, 154172.

Chen, W., Mu, Y., Liang, X., & Gao, Y. (2019, July). Medical data sharing model based on blockchain. In Journal of physics:

conference series (Vol. 1267, No. 1, p. 012014). IOP Publishing.
Chen, Y., Zhang, Y., Wang, S., Wang, F., Li, Y., Jiang, Y., ... & Guo, B. (2022b). DIM-DS: Dynamic Incentive Model for Data Sharing in Federated Learning Based on Smart Contracts and Evolutionary Game Theory. *IEEE Internet of Things Journal, 9*(23), 24572-24584.
Cheng, D., Xu, T., & Qi, H. (2013). Evolutionarily stable strategy of networked evolutionary games. *IEEE Transactions on Neural Networks and Learning Systems, 25*(7), 1335-1345.
Chowdhury, M. J. M., Colman, A., Kabir, M. A., Han, J., & Sarda, P. (2018, August). Blockchain as a notarization service for data sharing with personal data store. In 2018 17th IEEE international conference on trust, security and privacy in computing and communications/12th IEEE international conference on big data science and engineering (TrustCom/BigDataSE) (pp. 1330-1335). IEEE.
Cui, J., Ouyang, F., Ying, Z., Wei, L., & Zhong, H. (2021). Secure and efficient data sharing among vehicles based on consortium blockchain. *IEEE Transactions on Intelligent Transportation Systems, 23*(7), 8857-8867.
Dey, A., Debnath, M., & Pandey, K. M. (2017). Analysis of effect of machining parameters during electrical discharge machining using Taguchi-based multi-objective PSO. *International Journal of Computational Intelligence and Applications, 16*(02), 1750010.
Du, S. (2023). An evolutionary game-theoretic analysis of cooperation strategy between SMEs and cross-border e-commerce platforms considering the cross-network effect. *Kybernetes.*
Esmaeili, P., Makui, A., Seyedhosseini, S., & Ghousi, R. (2022). The effect of probabilistic incentives to promote cooperation during the pandemics using simulation of multi-agent evolutionary game. *International Journal of Industrial Engineering Computations, 13*(3), 319-328.
Feng, H., Wang, X., Duan, Y., Zhang, J., & Zhang, X. (2020). Applying blockchain technology to improve agri-food traceability: A review of development methods, benefits and challenges. *Journal of cleaner production, 260*, 121031.
Feng, Y & Yu, S. (2023). Pricing and coordination of remanufacturing supply chain considering remanufacturing capacity and preferences under government mechanisms. *International Journal of Industrial Engineering Computations, 14*(2), 173-200.
Friedman, D. (1991). Evolutionary games in economics. *Econometrica: Journal of the Econometric Society*, *59*(3), 637-666.
Fu, G., Yu, Y., Sun, W., & Kaku, I. (2023). To reduce maximum tardiness by Seru Production: model, cooperative algorithm combining reinforcement learning and insights. *International Journal of Industrial Engineering Computations, 14*(1), 65-82.
Fudenberg, D., Drew, F., Levine, D. K., & Levine, D. K. (1998). The theory of learning in games (Vol. 2). MIT press.
Gao, L. (2019, December). Evolutionary game analysis of servitization transformation in manufacturing enterprises. In AIP Conference Proceedings (Vol. 2185, No. 1, p. 020005). AIP Publishing LLC.
Gui, Y., Du, C., & Gao, L. (2022). A Matrix Approach to the Modeling and Analysis of Network Evolutionary Games with Mixed Strategy Updating Rules. *Mathematics, 10*(19), 3612.
Guo, B., Deng, X., Guan, Q., Tian, J., & Zheng, X. (2018). An incentive mechanism for cross-organization data sharing based on data competitiveness. *IEEE Access, 6*, 72836-72844.
Guo, B., Deng, X., Tian, J., Guan, Q., & Zheng, X. (2019). A secure incentive mechanism for competitive organization data sharing: A contract theoretic approach. *IEEE Access, 7,* 60067-60078.
Guo, X., Zhang, G., & Zhang, Y. (2023). A Comprehensive Review of Blockchain Technology-Enabled Smart Manufacturing: A Framework, Challenges and Future Research Directions. *Sensors, 23*(1), 155.
GuoHua, Z., & Wei, W. (2021). Study of the Game Model of E-Commerce Information Sharing in an Agricultural Product Supply Chain based on fuzzy big data and LSGDM. *Technological Forecasting and Social Change, 172*, 121017.
Han, Y., Niyato, D., Leung, C., Kim, D. I., Zhu, K., Feng, S., ... & Miao, C. (2022). A dynamic hierarchical framework for iot-assisted digital twin synchronization in the metaverse. *IEEE Internet of Things Journal, 10*(1), 268-284.
Jiang, C., Kuang, L., Han, Z., Ren, Y., & Hanzo, L. (2017). Information credibility modeling in cooperative networks: equilibrium and mechanism design. *IEEE Journal on Selected Areas in Communications, 35*(2), 432-448.
Liu, W., Long, S., Wei, S., Xie, D., Wang, J., & Liu, X. (2022). Smart logistics ecological cooperation with data sharing and platform empowerment: an examination with evolutionary game model. *International Journal of Production Research, 60*(13), 4295-4315.
Liu, Y., Wang, H., Peng, M., Guan, J., & Wang, Y. (2020). An incentive mechanism for privacy-preserving crowdsensing via deep reinforcement learning. *IEEE Internet of Things Journal, 8*(10), 8616-8631.
Lu, Y., Li, P., & Xu, H. (2022). A Food anti-counterfeiting traceability system based on Blockchain and Internet of Things. *Procedia Computer Science, 199*, 629-636.
Nofer, M., Gomber, P., Hinz, O., & Schiereck, D. (2017). Blockchain. *Business & Information Systems Engineering, 59*, 183-187.
Shi, Y., & Rong, Z. (2022). Analysis of Q-learning like algorithms through evolutionary game dynamics. *IEEE Transactions on Circuits and Systems II: Express Briefs, 69*(5), 2463-2467.
Smith, J. M., & Price, G. R. (1973). The logic of animal conflict. *Nature, 246*(5427), 15-18.
Wang, F. K., & Du, T. (2007). Applying capability index to the supply network analysis. *Total Quality Management and Business Excellence, 18*(4), 425-434.
Wang, L., Fan, L. T., Zhang, L., Zou, R. C., & Wang, Z. (2023). Synergistic effects of adaptive reward and reinforcement learning rules on cooperation. *New Journal of Physics*.

Wang, T., Li, C., Yuan, Y., Liu, J., & Adeleke, I. B. (2019). An evolutionary game approach for manufacturing service allocation management in cloud manufacturing. *Computers & Industrial Engineering, 133*, 231-240.

Wang, Y., Nakao, A., Vasilakos, A. V., & Ma, J. (2011). P2P soft security: On evolutionary dynamics of P2P incentive mechanism. *Computer Communications, 34*(3), 241-249.

Wang, Z., Tian, Y., & Zhu, J. (2018, August). Data sharing and tracing scheme based on blockchain. *In 2018 8th international conference on logistics, Informatics and Service Sciences (LISS) (pp. 1-6). IEEE.*

Wen, Q., Gao, Y., Chen, Z., & Wu, D. (2019, May). A blockchain-based data sharing scheme in the supply chain by IIoT. *In 2019 IEEE International Conference on Industrial Cyber Physical Systems (ICPS) (pp. 695-700). IEEE*.

Xia, Z., Tan, J., Gu, K., Li, X., & Jia, W. (2020). IDS intelligent configuration scheme against advanced adaptive attacks. *IEEE Transactions on Network Science and Engineering, 8*(2), 995-1008.

Xuan, S., Zheng, L., Chung, I., Wang, W., Man, D., Du, X., ... & Guizani, M. (2020). An incentive mechanism for data sharing based on blockchain with smart contracts. *Computers & Electrical Engineering, 83*, 106587.

Yang, J., Wen, J., Jiang, B., & Wang, H. (2020). Blockchain-based sharing and tamper-proof framework of big data networking. *IEEE Network, 34*(4), 62-67.

Yang, X., Li, M., Yu, H., Wang, M., Xu, D., & Sun, C. (2021). A trusted blockchain-based traceability system for fruit and vegetable agricultural products. *IEEE Access, 9*, 36282-36293.

Yu, Y., He, Y., & Zhao, X. (2021). Impact of demand information sharing on organic farming adoption: An evolutionary game approach. *Technological Forecasting and Social Change, 172*, 121001.

Yu, Y., Jin, T., & Zhong, C. (2015). Designing an incentive contract menu for sustaining the electricity market. *Energies, 8*(12), 14197-14218.

Zhang, J., & Wu, M. (2021). Cooperation mechanism in blockchain by evolutionary game theory. *Complexity, 2021*, 1-9.