Incentive Mechanism for P2P Networks Based on Feature Weighting and Game Theory

Min Du[*], Danlei Du

School of Electronics and Information Engineering, Hunan University of Science and Engineering, Yongzhou 425100, China

Corresponding Author Email: duminemail@huse.edu.cn

ABSTRACT

There are mainly three types of problems in Peer-to-Peer (P2P) networks such as free-riding nodes, group deception, and node bias. To solve this, the authors proposed an incentive mechanism for the P2P networks based on feature weighting and game theory. The mechanism first used the five comprehensive coefficients of node degree, node clustering coefficient, local clustering coefficient, all clustering coefficients, and correlation coefficients to form a feature clustering matrix through linear fusion; then, in order to maximize the overall revenue, a sparse matrix of revenue was generated through feature classification, noise reduction, mapping and iteration, highlighting the status of fully cooperative nodes; afterwards, combining group evolution and constraint rule sets, the authors determined group node dynamic adjustment rules, response service rules, and message query and forwarding rules, to maximize service efficiency; finally, a multi-layered P2P dynamic service system was constructed to promote the active evolution of nodes, and avoid the negative migration. The simulation experiment was also performed to verify this mechanism. The research findings provide an effective idea for stimulating node behaviors, reducing negative node migration, and excellent node selection.

1. INTRODUCTION

Because of its characteristics of openness, dynamics, heterogeneity, peer-to-peer, resource sharing, and no need of centralized server support, P2P networks can eliminate information resource islands and service bottlenecks in C/S. It shows its uniqueness in the aspects of cooperative work, distributed information, resource sharing, and large-scale parallel computing etc. [1, 2]. However, the nodes in P2P are not subordinate to any entity in the network, and they use the method of autonomous management and self-determined resource contribution, which results in a lack of centralized authentication and authorization mechanism, and also a large number of selfish nodes in real networks [3, 4]. Shneidman and Parkes [3] in 2005 found after measuring the Gnutella system that free-riding nodes in the P2P network had reached 85%, while 1% of nodes provided queries at a rate of up to 50%, and frequent responses from these nodes lead to excessive load and severe congestion. Meanwhile, only by minimal cost can the malicious groups emerge and carry out attacks in the network [5], thus exacerbating the bias of selfish nodes and causing a serious decline in network resource availability. For this, an effective incentive mechanism should be designed to encourage nodes in P2P to voluntarily and actively provide services, restrict free-riding nodes, prevent group fraud, slow down or avoid negative migration of nodes, and further improve the trust mechanism of P2P networks, which is one of the important issues related to the effective survival of P2P networks.

In recent years, scholars at home and abroad have studied the incentive mechanism of P2P network nodes from various aspects of node behavior incentive mechanism [6-8], algorithm design [9-11], model design [12-14], and strategy design. Ouyang et al. [7] designed an incentive cooperation model based on the punishment mechanism to encourage nodes' participation in cooperation, and motivate selfish nodes to actively participate in network activities through different behavioral strategies such as enthusiastic contributions to the system, greedy downloads, and contribution-income balance. Hu et al. [8] established service response rules, query forwarding rules, and neighbor selection rules to better encourage resource sharing of nodes by setting differences in global trust values between nodes, but ignoring the group attacks of malicious nodes to obtain high global trust values. Liu and Yi [15] discovered the dynamic process of social network evolution through multi-community dynamic network evolution from the micro perspective. Funasaka et al. [16] found that the immediate reconstruction of neighboring nodes does not consider the deterioration of download behavior, so it proposed the idea of adjusting the neighboring nodes and changing the group where the nodes belong to using the tit-for-tat strategy, but this method failed to consider the group deception issue of P2P nodes. The above studies are all conducted in the absence of malicious nodes' group deception; usually only one feature attribute is considered and other features are ignored, without breaking the clustering limit. In fact, P2P nodes have typical clustering characteristics. The emergence of a large number of completely uncooperative nodes (mainly incl. free-riding nodes and malicious nodes) not only causes overload and congestion of high-performance nodes when frequently responding to many service requests, and the decline in the availability of network resources, but also accelerates the negative bias of rational cooperation nodes, and results in a more obvious negative migration.

In view of the above, this paper proposes the incentive mechanism for P2P networks based on feature weighting and

evolutionary game. This mechanism can help nodes actively evolve, break through clustering restrictions, and avoid negative migration. Based on clustering coefficient settings, feature income mapping, and rule setting, a multi-layered P2P dynamic service system was constructed by node adjustment, response service, and query forwarding rules etc.

2. SYSTEM SETTING

2.1 Related definitions of clustering coefficients

At present, there is no uniform standard for selection of network feature attributes. This paper focuses on the construction of clustering coefficients based on Kondor and Lafferty [17].

(1) Node i and its degree k_i; the degree is the number of edges connected by node i. In principle, the larger k_i, the higher the importance of the network.

(2) The node clustering coefficient C(V_i), i.e., the ratio of the k_i edges adjacent to node i to the $C_{k_i}^2$ edges of the possible complete graph, as shown in Eq. (1):

$$C(V_i) = \frac{k_i}{C_{ki}^2} \tag{1}$$

(3) The local clustering coefficient $C(k)$ represents the ratio of the average number of edges (in a neighbor of the node with k degree) to the maximum possible number of edges, which is described as Eq. (2):

$$C(k) = \frac{avg(k)}{C_k^2} \tag{2}$$

where, $avg(k)$ is the average number of edges for the neighbor of a node with degree k.

(4) The global clustering coefficient C represents the ratio of the average number of edges to the maximum number of edges that exist between the neighbors of nodes in the network. It is described as Eq. (3):

$$C = (\sum_{k=0}^{max}(k-1) \bullet k \bullet C(k))/(A_k^2 - A_k^1) \tag{3}$$

where, A_k^i is the origin of order i with degree k.

(5) Vector correlation coefficients $\rho_{X,Y}$ describe the degree of linearity between the two reaction vectors X and Y, as described in Eq. (4):

$$\rho_{X,Y} = \frac{1}{n-1}\sum_{i=1}^{n}(\frac{X_i - \overline{X}}{D_X}) \bullet (\frac{Y_i - \overline{Y}}{D_Y}) \tag{4}$$

where, $\overline{X}, \overline{Y}$ are the average values of the vectors, and D_X, D_Y are the variances of the vectors X, Y. In value range $\rho_{X,Y} \in [-1, 1]$, when $|\rho|$ is closer to 1, the linear correlation between the two vectors is higher.

2.2 Solving the feature fusion matrix

Generate the fusion matrix in the following process:

(1) Classify different attributes according to the node degree, and calculate the feature clustering matrix in the initial state;

(2) Perform thresholding for the feature clustering matrix from the previous step to obtain attribute matrices of different states [10]; record the accurate number of matrices and record the accuracy rate as m;

(3) Select the threshold with the best classification effect in each attribute, and perform linear fusion with the correlation coefficient matrix to construct and generate a feature fusion matrix A and thus improve the classification accuracy.

The node degree matrix of optimal threshold is shown in Eq. (5):

$$D = \begin{bmatrix} d_{11} & d_{12} & ... & d_{1n} \\ d_{21} & d_{22} & ... & d_{2n} \\ ... & ... & ... & ... \\ d_{n1} & d_{n2} & ... & d_{nn} \end{bmatrix} \tag{5}$$

The clustering coefficient matrix of optimal threshold is shown in Eq. (6):

$$CC_A = \begin{bmatrix} c_{11} & c_{12} & ... & c_{1n} \\ c_{21} & c_{22} & ... & c_{2n} \\ ... & ... & ... & ... \\ c_{n1} & c_{n2} & ... & c_{nn} \end{bmatrix} \tag{6}$$

The correlation coefficient matrix is given in Eq. (7):

$$P = \begin{bmatrix} \rho_{11} & \rho_{12} & ... & \rho_{1n} \\ \rho_{21} & \rho_{22} & ... & \rho_{2n} \\ ... & ... & ... & ... \\ \rho_{n1} & \rho_{n2} & ... & \rho_{nn} \end{bmatrix} \tag{7}$$

The feature fusion matrix is given in Eq. (8).

$$A = \begin{bmatrix} A_{11} & A_{12} & ... & A_{1n} \\ A_{21} & A_{22} & ... & A_{2n} \\ ... & ... & ... & ... \\ A_{n1} & A_{n2} & ... & A_{nn} \end{bmatrix} = \begin{bmatrix} m(d) \bullet d_{11} & ... & m(P) \bullet \rho_{1n} \\ m(d) \bullet d_{21} & ... & m(P) \bullet \rho_{2n} \\ ... & ... & ... \\ m(d) \bullet d_{n1} & ... & m(P) \bullet \rho_{nn} \end{bmatrix} \tag{8}$$

3. FEATURE INCOME MAPPING

The resource sharing of nodes in P2P is actually a process of resource co-building and sharing through mutual game of individuals. After a period of gaming, the resources will generate k multipliers, and $k>0$. Considering that the most commonly used method in P2P is to establish strong connections and resource sharing from directly adjacent nodes that also tend to be more important. Thus, it will more fully reflect the clustering performance to effectively map the feature fusion matrix to the feature income matrix.

The feature fusion matrix can highlight the original features maximally. However, in the adjustment process of real-time change, more features are randomly added or de-noised. In order to retain the original features of the task and the output characteristics after adding or removing noise, it's necessary increase the times of adding or eliminating the noise as much as possible [18], which can be achieved by using feature income mapping plus multiple iterations.

3.1 Grouping and mapping

Considering that the sparse similarity matrix can better highlight the clustering performance [11], feature zeroing can be used at random to reduce noise. Also, because the data of the same clustering feature differ only in a few features, random noise reduction for multiple times on the same data can be analogized to the common representation of the same clustering feature in different states.

3.1.1 Feature classification and noise reduction

Considering the "contribution-reward" rule of the nodes and different prominence of the features in the feature fusion matrix A_{ij}, the feature ordering was performed: $A_{(1)} \geq A_{(2)} \geq \ldots \geq A_{(n^2)}$

In addition, the probability of noise reduction for each original feature should be 1-α higher than that of the total nodes, where α<(0,1), and α is the threshold. With the first α•n^2 features that satisfy the ordering unchanged, the remaining features are all set to 0, to generate a sparse matrix. It is shown in Eq. (9):

$$A_{ij} = \begin{cases} A_{ij} & At\ A_{ij} = A_{(k)}, and\ k \leq \alpha \bullet n^2 \\ 0 & Others \end{cases} \quad (9)$$

3.1.2 Feature feedback and income mapping

Different behavioral strategies of nodes generate different reward feedback, and affect the further behavior of nodes, so as to achieve the consistency between the nodes and the overall interests finally.

Therefore, highlight the features of fully contributing nodes by strengthening the tightness of this type of node group, and improve the overall capital of the network; perform the income mapping of the feature fusion matrix after the noise reduction, and realize the maximum benefit of public resource capital [19]. The specific process is as follows:

(1) For existing de-noised sparse matrix $A = [A^1, A^2, \ldots, A^n] \in A^{\tilde{n} \times \tilde{n}}$ ($\tilde{n}$ is the number of nodes), add the feature constant signal multiplier k, and then $A = [A; k] \in A^{(\tilde{n}+1) \times \tilde{n}}$, among which $k \in R^{1 \times \tilde{n}}$. mSDA [11] was used to optimize feature targets.

(2) Perform noise reduction for m times, and obtain the target optimization result as shown in Eq. (10):

$$\min_K \frac{1}{2m\tilde{n}} \sum_{i=1}^m \|A - K\tilde{A}_i\|_F^2 \quad (10)$$

where, A_i is the i-th version of A after adding the signal multiplier k, K is the feature matrix of the resource multiplied signal, and F is the process of multiple feature noise reductions.

(3) Perform multiple iterations of the features on a row basis to generate a revenue matrix M that has both the unique revenue characteristics and the common characteristics among the revenues, as shown in Eqns. (11) and (12):

$$h^i = [A^1, \ldots, A^{\tilde{n}}] \quad (11)$$

$$M = [h^0; \ldots; h^g] \quad (12)$$

where, h^i is the income data source for the i-th group and M is the overall revenue matrix generated by all the g groups.

(4) Measure the expected value of node income. During the network evolution, nodes generate certain expectations based on their own conditions and the surrounding environment, in order to motivate the active networks of the nodes. In the game process, define the node i's revenue expectations as shown in Eq. (13):

$$E(M_{ij}) = \frac{k_{ij} \bullet m_{ij}}{\tilde{n} \sum_{n1=1}^{\tilde{n}} C_{\tilde{n}}^{n1}(1 - \frac{k_{ij}}{\sum_{n2=1}^{\tilde{n}} k_{ij}}) \bullet \frac{k_{ij}}{\sum_{n2=1}^{\tilde{n}} k_{ij}}} \quad (13)$$

4. GROUP EVOLUTION RULES

The rules are set to guide the nodes and generate different "reward" feedback. This can further regulate the behavior of the nodes, prevent negative migration, and achieve consistency of the overall interests of the nodes and the group.

4.1 Grouping rules

The nodes $N_1, N_2, \ldots, N_n$ were arranged according to the revenues: $M_{(1)} \geq M_{(2)} \geq \ldots \geq M_{(n)}$ from large to small. Using the idea of equal sharing, n nodes were divided into $\lceil \sqrt{n} \rceil$ groups.

$M_{(1+(l-1) \bullet \lceil \sqrt{n} \rceil)}, \ldots, M_{(l \bullet \lceil \sqrt{n} \rceil)}$ (1) If $n/\lceil \sqrt{n} \rceil$ is an integer, distribute the $n/\lceil \sqrt{n} \rceil$ nodes into the l-th group, l=1,2,…, $\lceil \sqrt{n} \rceil$, according to the revenues from large to small,

$M_{(1+(l-1) \bullet |n/\lceil \sqrt{n} \rceil|)}, \ldots, M_{(l \bullet |n/\lceil \sqrt{n} \rceil|)}$ (2) If $n/\lceil \sqrt{n} \rceil$ is not an integer, then distribute the $n/\lceil \sqrt{n} \rceil$ nodes into the l-th group, l=1,2,…, $\lceil \sqrt{n} \rceil - 1$, according to the revenue from large to small,

(3) The $\lceil \sqrt{n} \rceil$-th group is allocated with the remaining nodes.

4.2 Initialization of group evolution

At the beginning of the evolution, at t=0, the nodes were evenly distributed to groups. To ensure the connectivity between the nodes and the group, there was at least one connected edge between the groups. According to the principle of clustering together, the services received by the nodes in the group G_i were mainly from the group inside, and the nodes in the group had to provide services to their upper nodes to obtain multiple benefits, thereby improving the overall efficiency. It's also agreed that when the average revenue G_i of the group is greater than the group G_j, the service level and response speed of the nodes in this group shall be higher than G_j.

4.3 Dynamic adjustment of group nodes

4.3.1 Overall analysis

According to their features, nodes with similar revenues and similar behavioral capabilities and behavioral characteristics were placed in the same group G. A large number of independent revenue factors of this type of nodes comprehensively affected the overall revenue of the group, while the individual revenue of each node played a minor role in the group. The influencing process of this random change approximately follows a normal distribution. Its characteristics are shown in Eqns. (14) and (15):

$$\frac{\sum_{k=1}^{\lfloor n/\lfloor\sqrt{n}\rfloor\rfloor} X_k}{\sqrt{n}\sigma} \overset{Approximately}{\sim} N(0,1) \tag{14}$$

$$\sigma = \sqrt{\frac{1}{m_G - 1}\sum_{i=1}^{m_G}(E(M_i) - E(G))} \tag{15}$$

where, σ is the threshold of group adjustment.

4.3.2 Node adjustment

Within unit time, node i was allocated to a group G according to its revenue, that is, exchanging resources with the m_G nodes in the group. It also sent out requests for service connections from other groups with a small probability. The rules of node adjustment are as follows:

(1) Within the group, if the node incomes are $E(M_i) < E(G) - \sigma$, the node is downgraded to the next group and stops applying for service requests from the community and the higher-level community;

(2) Within the group, if the node incomes are $E(M_i) > E(G) + \sigma$, the node is ascended to the previous group, and stops applying for service requests from this community and lower-level community to apply for a service request.

Based on this rule, about 31.74% of the nodes in each group can be upgraded or descended.

5. CONSTRAINT RULE SETTING

5.1 Response service rules

Nodes were aggregated according to the principle of clustering. Nodes in the same group provided services to each other, while the group escorts the nodes. The incentive rules are explained as follows:

(1) If $E(M_i) > E(M_j) > E(G)$, it means that node j in the same group has most likely enjoyed the service of node i before, and when node i has a service request, node j must respond accordingly.

(2) If $E(M_j) \geq E(M_i) > E(G)$, it means that it is less likely that node j in the same group has enjoyed the service provided by node i, but node i has made a certain contribution to the group, although the contribution is smaller than node j; when i has requested service, Node j should respond to the service of node i with a probability of $\Delta = \frac{E(M_i)}{E(M_j)}$; when the probability value Δ is smaller, the probability of responding to the service is lower.

(3) If $E(M_i) \geq E(G) - \sigma > E(M_j)$, j hasn't had the same level of service request as node I, and node j needs to be downgraded to the next group. Its service should be obtained from the downgraded group or a lower level group.

(4) The newly added nodes are likely to be free-riding nodes, or malicious nodes that constantly change identities repeatedly and join the network to avoid providing high-quality services or system punishment for a long time. In this study, the strategy adopted was for the system to preemptively prevent such nodes from enjoying any network services, assign such nodes to the lowest group, and starting upgrading from the lowest group.

5.2 Query message and forwarding rules

Set message forwarding rules based on the revenue expectations of the group. Set the message forwarding probability p_{ji} as shown in Eq. (16):

$$p_{ji} = \begin{cases} 0 & \begin{array}{l}\textit{Message to be}\\ \textit{found in the}\\ \textit{same level or}\\ \textit{lower}-\textit{level group}\end{array} \\ \left(E(G_j)/E(G_i)\right)^{1-\frac{1}{j-i}} & \begin{array}{l}\textit{Message to be}\\ \textit{found in the upper}\\ -\textit{level group}\end{array} \end{cases} \tag{16}$$

(1) If the message can be queried in the same group, no forwarding is required. The length of the query propagation path is 0, the probability of message forwarding is 0, and the response probability is 1.

(2) If the message can be queried in the upper-level group, and $E(G_i) > E(G_j)$, then the group G_j will unconditionally forward the query message with a higher income than its own group, and it's also agreed that with the number of forwarding times increasing, the propagation path shall be longer, and the response probability at this time is smaller.

(3) If the message can be queried in the lower group, that is, $E(G_i) > E(G_j)$, the group G_i will not forward the message; at this time, the query propagation path length is $j - i$, and the response probability is 1.

6. SIMULATION EXPERIMENTS

Based on the mechanism described in this paper, a simulation environment was constructed to test the performance. The simulation scenario was filing sharing and file download; the criterion is whether the file download is successful. Considering malicious behaviors such as malicious nodes and node bias, the following three types of nodes were designed according to design requirements:

(1) Unified cooperative nodes (UC). No matter what strategy other type of node adopts, this type of node always provides good service and honest evaluation, and there is no dynamic behavior and collusion behavior.

(2) General cooperative nodes (RC). These nodes strictly follow the rules proposed, but there may be a situation where the negative offset is intensified, when high-quality services cannot be obtained for a long time.

(3) Unified uncooperative nodes (UN). This type of nodes does not provide shared files, or only share a small number of file nodes that can be hardly accessed by anyone, regardless of any strategy adopted by the other type of nodes. In fact they are nodes and malicious nodes, of which the malicious nodes have two typical and complex collusion attacks: first, according to the role assignments, some malicious nodes are responsible for improving the evaluation of certain malicious nodes, increasing the revenue of such nodes in a deceiving way, and achieving their group improvement of efficiency and speed; second, the nodes regularly exchange roles to maintain their respective high node and group returns, thereby deceiving high-quality services and exacerbating the bias of rational cooperation nodes.

There were 600 network nodes in the simulation environment, of which 120 were UC nodes at the 20% threshold. The total number of files was 1,200, and the number of connections between nodes was 3 to 5, and the number of groups was 25. Then, 1,200 files were randomly distributed to 600 nodes, and each file was guaranteed to be owned by at least one UC node. During the simulation process, each node made a download request for 10 files. Next, the node receiving the request could respond to it, and from all the responding nodes select the node with the highest revenues in the same group or a higher-level group for file download. If it succeeds, the node owns the file; otherwise, it cannot. Considering group deception and negative offset of nodes, several cycles were used in the simulation analysis to take the average value. The simulation also used Inter i7-9700 3GHZ CPU, 8G memory, PeerSim platform, and C++ language. For comparison, the method in literature 7 was also implemented. Table 1 lists the designed simulation parameters.

Table 1. Simulation parameters

Parameter type	Name	Value
Network environment	Total nodes	600
	Node degree	3-5
	Initial topological structure	Random topology
Nodes	Node types	UC\ RC\ UN
	Number of groups	25
	Cycles of service request	time/hour
	Response cycle of services	time/hour
Files	Total files	1200
	Distribution of shared files	Uniform random
Thresholds α		0.2

6.1 Revenue simulation of various nodes

In the experiment, it's assumed that the ratio of UC, RC, and UN nodes was 1: 1: 1, that is, 200 nodes of each type. The simulation results are shown in Figure 1.
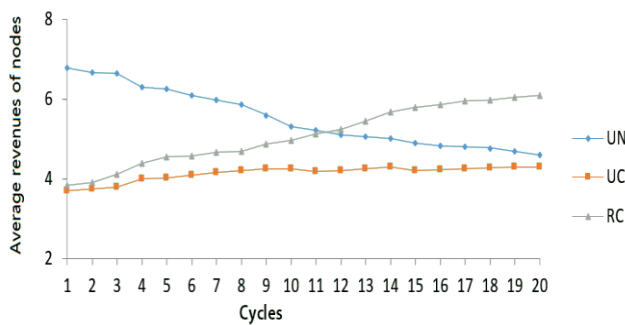

Figure 1. Performance comparison for the three types of nodes

At the initial stage of the simulation, all nodes in the network were not differentiated and did not provide differentiated services. So, it could not effectively perform game evolution, and differentiate the service levels. At this time, the UC and RC nodes were in a fully cooperative state and working hard to serve the UN nodes, while the UN node contributed the files that have a small number or is rarely accessed, and cannot effectively serve the UC and RC nodes, so that the UC and RC nodes will lose their bandwidth and computing power but obtaining no services. Therefore, the average revenue of UN nodes was much higher than that of UC and RC nodes.

As the simulation continued, it's also found that the node revenues with large contributions increased, and the differentiation mechanism took effect. Then, the UC node with relatively low revenue might continue to provide services for the UN node, the RC node chose to reject the UN service request, and the revenue of the UN node decreased significantly. The revenue of the UC and RC nodes increased, but that of the UC node showed a slow trend. After 20 cycles, the RC node had the highest return, verifying the effectiveness of the differentiation mechanism, and accelerating the bias of the RC node; whereas, the average revenue of UC nodes was slightly lower than that of UN, indicating that there are many UN nodes in the network. Even if not providing any service, the revenues of UN nodes are equal to or slightly higher than those nodes that strive to provide services. This is also the reason why in P2P most nodes choose to free-riding.

6.2 Simulating the change law of UN nodes

6.2.1 Ratio setting of initial nodes

In reality, the number of UC nodes is very small, not exceeding 20%, so the ratio of UC nodes was set to 5%. The number of UN nodes is far more than 50%. Therefore, by constantly changing the ratios of UN nodes (10%-80%), the authors calculated the average revenue value of various types of nodes and the bias of RC nodes.

6.2.2 Analysis for the average revenue of the three types of nodes before the use of incentives

The average revenues of the three types of nodes before the use of measures are shown in Figure 2(a).
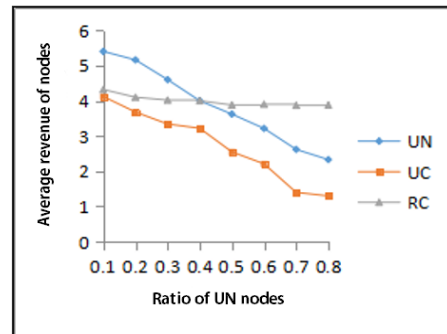
(a) Before the use of incentives
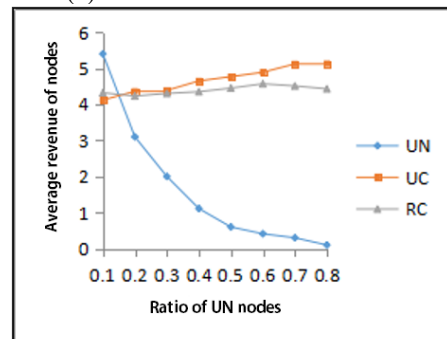
(b) After the use of incentives

Figure 2. Comparison of the average revenue between the three types of nodes under different ratios of UN nodes

(1) With the ratio of UN nodes between 10% and 30%, the average revenue of three types of nodes was around 4, which indicates a good state of cooperation. Also, UC and RC nodes make larger contributions, and remain a balance between the returns and contributions.

(2) With the ratio of UC nodes increasing (from 30% to 80%), the average revenue of RC nodes still remained the same, because RC nodes establish connections with each other and provide services to each other, and the average revenue of RC nodes varies with UN. However, the UC nodes frequently responded to the UN nodes for a long time, causing excessive load and heavy congestion, which decreased its average returns significantly. For UN nodes, because they have not been served by the UC node for a long time, their average return value also fell at the same rate as UC nodes.

6.2.3 Analysis for the average revenue of the three types of nodes after the use of incentives

The average returns of the three types of nodes after the used of incentives are shown in Figure 2 (b).

(1) With the ratio of UN nodes between 10% and 30%, once a UN node was found, the non-cooperation and non-response strategy were adopted immediately. At this time, the UC and RC nodes have made great contributions and are still in a good cooperative state; the average revenue of nodes was also around 4. But, once a free-riding situation was found in the UN node, its average revenue immediately dropped to about 2.

(2) With the ratio of UC nodes increasing (from 30% to 80%), the nodes with more contributions were closely connected in the form of group, and they provided services to each other through intra-group connections. Especially for UC nodes, they have great contributions and their service object is also a UC node or an RC node with the same contribution, so the average revenue of the UC node steadily rose to 5.2, which not only avoids the overload caused by frequent response to the UN node, but also enables the UC node itself to obtain high-quality services. Nevertheless, the UN nodes were both free-riding nodes and whitewashed malicious nodes. Therefore, the system preemptively assigned the UN nodes to low-revenue groups and prevented them from sharing any resources from the UC and RC. As a result, the UN nodes can only obtain inferior services internally, and their average returns dropped to almost 0.

(3) The use of incentives has intensified the bias of RC nodes. They have moved closer to UC nodes and tended to be similar to UC behavior patterns in order to obtain efficient services and efficient services, and thus contribute to the system. RC nodes have an upward trend, from an initial 4.3 to a later 4.5.

(4) Even if there was a group deception behavior by some malicious nodes, they might have obtained a high return from the role assignments in the early stage of formation through the collusion attack of other malicious nodes or group deception, and then allocated to the higher-level group during the group initialization process. However, according to the adjustment rules, the service objects of such nodes were not in the group or the upper-level group, and their revenue would quickly fall below the threshold of the average revenue level of the group, so they were re-adjusted to the low-income group. This type of nodes is mainly a group with low return evaluation. Due to the low evaluation weights of the group, it is very unlikely that the node level can be improved through the evaluation within the group. Therefore, this type of node has been in low-revenue group for a long time. For this, the

collusion nodes were either isolated or withdraw from the malicious group because of its own interests, and took active actions to obtain higher returns, which has also proven to be correct in real life.

6.3 Network feature measurement

At the initial stage, 600 nodes were divided into 25 groups, and each group had 24 nodes. Also, each node was set to be connected with other nodes at a probability of 0.1. Then, related analysis was conducted on the force distribution of stabilized network nodes, network stability, and the overall network benefits.

(1) Analysis of node forces

If a node has a greater force on other nodes, it shall be more important. Figure 3 shows the acting force of nodes in the stabilized evolution environment with n=600 and 80% of free-riding nodes. It can be seen from the figure that only a small number of nodes had stronger forces (72 nodes with the force greater than 0.6), and it accounted for 12% of the total, without exceeding 20%; most nodes were weak as free-riding nodes. Besides, there were 262 nodes with the force [0.2, 0.4], and concentrated near the mean (0.307), accounting for 42% of the overall distribution. This is not only consistent with Pareto's law in reality, but also with the characteristics of scale-free networks.
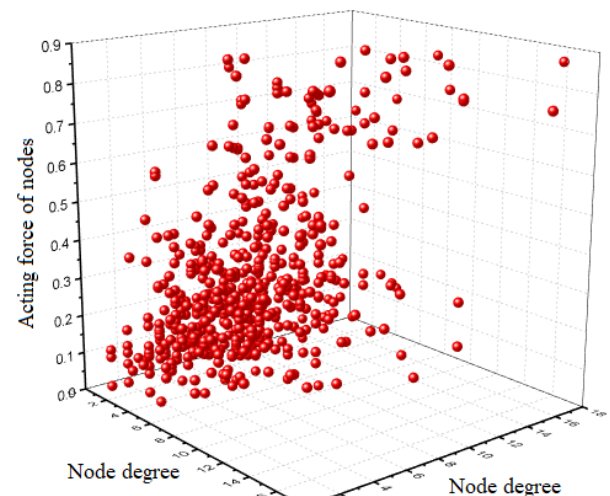

Figure 3. The acting force of network nodes in the stabilized evolution environment

(2) Analysis of network stability

The revenues of nodes are closely related to node preferences, node cooperation, and factors affecting the network environment. The network environment parameters have a great impact on network stability. If nodes can set reasonable network parameters to automatically adjust policies and expectations, they can obtain higher returns and better adaptability, and then maintain the effectiveness and stability of network.

At the beginning of the evolution, the use of random selection strategy could ensure a high frequency of cooperation. However, with the increase in the proportion of UN nodes, RC nodes might have short-term betrayal, but for personal interests, they chose to give up the betrayal behaviors and work with RC or UC nodes to ensure that the benefits are multiplied. Figure 4 shows the influence of the multiplication

factor k and noise T on the correlation coefficient ρ. Initially, the small k has a large impact on ρ; when it becomes large, it doesn't; also, when the noise T is small, it is not conducive to network stability, but the stable network has a weak effect on T.
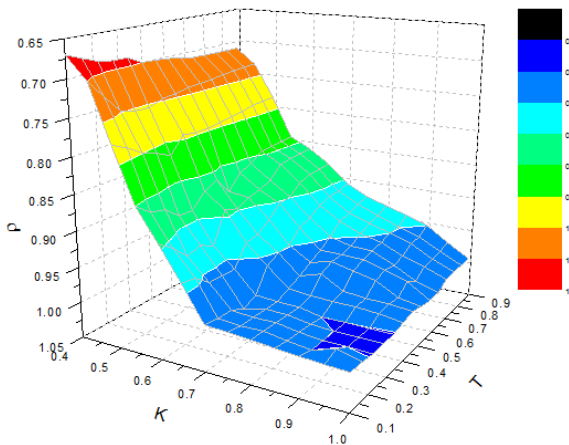

Figure 4. Comparison on impact analysis of different parameters under evolutionarily stable environment

(3) Comparative analysis of overall network revenue

To clearly reflect the overall network revenue of the three types of nodes (UN, RC, and UC) under different proportions, the authors set three network environments dominated by different nodes; network X: UC node 20%, RC20%, UN60%; network Y: UC node 20%, RC60%, UN20%; network Z: UC node 60%, RC20%, UN20%. Their overall revenue of the stabilized network was then calculated.

It can be seen from Figure 5 that the X network mainly included free-riding nodes and malicious nodes, and its overall revenue was extremely low, much lower than the Y network and the Z network; the Z network was mainly the UC node, and its overall network revenue was close to 5; the overall revenue of Y network mainly dominated by RC was higher than that of Z network. This indicates that the method proposed in this paper can exacerbate the bias of RC nodes, which shall obtain poor service because the malicious nodes are mainly in the low-income group despite the collusion attacks of the malicious nodes.
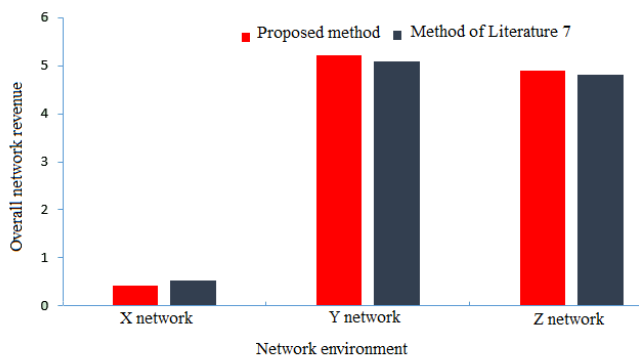

Figure 5. Comparison for the overall network revenue based on different proportion of nodes

Meanwhile, Figure 5 also shows that compared with the methods in Literature 7, the overall revenue obtained by our method is slightly higher, when the sum of UC nodes and RC nodes accounted for a high proportion in the network; the overall venue obtained by the method of Literature 7 is higher than our method, when UN nodes occupy a relatively high proportion in the network, This means that the method proposed in this paper is superior to that in Literature 7 in terms of motivating the bias of RC nodes, but slightly poorer than literature 7 in the restriction of UN nodes.

7. CONCLUSIONS

From the perspective of evolution, this paper proposes an incentive mechanism for P2P network based on feature weighting and game theory. This mechanism first uses the five comprehensive coefficients of node degree, node clustering coefficient, local clustering coefficient, all clustering coefficients, and correlation coefficients to form a feature clustering matrix through linear fusion; then, in order to maximize the overall revenue, a sparse matrix of revenue was generated through feature classification, noise reduction, mapping and iteration, highlighting the status of fully cooperative nodes; afterwards, combining group evolution and constraint rule sets, the authors determined group node dynamic adjustment rules, response service rules, and message query and forwarding rules, to maximize service efficiency; finally, a multi-layered P2P dynamic service system was constructed to promote the active evolution of nodes, and avoid the negative migration. This study provides an effective idea for stimulating node behaviors, reducing negative node migration, and excellent node selection. However, due to the complexity of node behaviors in a P2P environment, it is almost impossible to fully describe and identify all the behaviors of nodes. Therefore, next study should focus on how to measure network performance, refine rules, increase the number of simulated nodes, and expand the network scale to make it closer to a real network, in order to satisfy the more complex and changing needs in reality.

REFERENCES

[1] Yue, G.X., Li, R.F., Chen, Z., Zhou, X. (2011). Analysis of free-riding behaviors and modeling restrain mechanisms for peer-to-peer networks. Journal of Computer Research and Development, 48(3): 382-397.

[2] Yu, Y.J., Jin, H. (2008). A survey on overcoming free riding in peer-to-peer networks. Chinese Journal of Computers, 31(1): 1-15. https://doi.org/10.3724/SP.J.1016.2008.00001

[3] Shneidman, J, Parkes, D. (2003). Rationality and self-interest in peer to peer networks. The 2nd Int'l Workshop on Peer-to-Peer Systems (IPTPS 2003). Berkeley, CA, USA, pp. 139-148. https://doi.org/10.1007/978-3-540-45172-3_13

[4] Hughesk, D., Coulson, G., Walkerdine, J. (2005). Free riding on gnutella revisited: The bell tolls. IEEE Distributed Systems Online (S1541-4922), 6(6): 1-18.

https://doi.org/10.1109/MDSO.2005.31

[5] Han, X.H., Xiao, X.Q., Zhang, J.Y., Liu, B.S., Zhang, Y. (2014). Sybil Defenses in DHT networks based on social relationships. Jouanal of Tsinghua University (Science and Technology), 54(1): 1-7.

[6] Qu, D.P., Wang, X.W., Huang, M. (2013). Selfish node detection and incentive mechanism in mobile P2P networks. Jouanal of software, 24(4): 887-899. https://doi.org/10.3724/SP.J.1001.2013.04290

[7] Ouyang, J.C., Lin, Y.P., Zhou, S.W., Li, W. (2013). Incentive mechanism based on global trust values in P2P networks. Jouanal of System Simulation, 25(5): 1046-1052.

[8] Hu, J.L., Zhou, B., Wu, Q.Y. (2011). Research on incentive mechanism integrated trust management for P2P networks. Jouanal of Communications, 32(5): 22-31. https://doi.org/10.1016/j.cageo.2010.07.006

[9] Lu, Q., Liu, B., Hu, H.P. (2016). Some critical issues of content sharing network. Jouanal of Communications, 37(10): 158-171.

[10] Yang, N., Zhang, D.K. (2019). Research on the sorting technique of brain networks based on multiform eigenvectors. Computer Engineering and Applications, 55(24): 96-101, 246.

[11] Zhang, X.T., Zhang, X.C., Liu, H. (2019). Weighed multi-task clustering by feature and instance transfer. Chinese Journal of Computers, 42(12): 2614-2630. https://doi.org/10.1016/j.neucom.2017.04.029

[12] Zhao, L.X., Tian, Y., Deng, L.Y., Ling, X., Tian, Y., Deng, L.Y. (2010). Dynamic trust model with incentive mechanism in p2p environment. Applicaiton Research Of Computers, 27(1): 251-254.

[13] Li, J.T., Jing, Y.N., Xiao, X.C., Wang, X.P., Zhang, G.D. (2007). A trust model based on similarity-weighted recommendation for P2P environments. Journal of Software, 18(1): 157-167.

[14] Du, M., Wang, X.L. (2012). P2P trust model of dynamic optimization for group competition based on entropy method. Computer Engineering and Applications, 48(17): 123-128.

[15] Liu, Q., Yi, J. (2013). The Research of the social network evolution based on the evolutionary game theory. Acta Physica Sinica, 62(23): 238902-1-208902-9.

[16] Funasaka, J., Yasuoka, H., Ishida, K. (2007). Dynamic peer grouping method conforming with tit-for-tat strategy for P2P file distribution systems. IEICE Transactions on Communications, 90-B(4): 809-816. https://doi.org/10.1093/ietcom/e90-b.4.809

[17] Kondor, R.I., Lafferty, J. (2002). Diffusion kernels on graphs and other discrete input spaces. Proceedings of ICMl, 315-322. http://www.cs.cmu.edu/~lafferty/pub/diffusion-kernels.ps

[18] Vincent, P., Larochelle, H., Bengio, Y., Manzagol, P.A. (2008). Extracting and composing robust features with denoising autoencoders. Proceedings of the 25th International Conference on Machine Learning, Helsinki, Finland, pp. 1096-1103. https://doi.org/10.1145/1390156.1390294

[19] Kempe, D., Kleinberg, J., Tardos, E. (2003). Maximizing the spread of influence through a social network. Proceedings of the ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD), Washington, USA, pp. 24-27. https://doi.org/10.1145/956750.956769