An Optimization Model for the Interconnection among Peers of the P2P Network

1Huang Yongsheng, 1Tian Xiaoyu and 2Tang Zhongbin
1Tangshan Key Laboratory of Informationization Technologies and Engineering Control,
School of Management, Hebei United University, China
2Department of Fundamental Courses, Xuzhou Air Force College, China

Abstract: In peer to peer (P2P) network with large number of peers, better service performances usually mean more network resources. Optimizing interconnection scheme among the peers is a solution to enhance efficiency of the file searching and achieving with reduced scale of interconnections. In this study, by proper formatting the interconnection relation and information about file replica searching, the Particle Swarm Optimization (PSO) algorithm is employed to optimize the interconnection among the active peers. As the variation of the active peers and the file replicas distribution in the P2P network, the optimization procedure is time consuming. In order to shorten the iteration procedure, the competitive neural network algorithm is selected to involve the optimization. Simulative experimental results indicate the proposed model can achieve enhanced and stable optimization efficiency and effectiveness. Meanwhile, the network loading will be decreased as the scale of interconnections among peers is reduced.

Key words: P2P, file replica searching, particle swarm optimization, neural network

INTRODUCTION

Information sharing is becoming an important activity in daily life. Taken as file sharing technology, peer to peer (P2P) network technology is one of the most widely used network technologies (Koenigstein et al., 2012). As a peer needs a file, it sends out search request to find the file replicas in the other peer. More file replicas are found with less time and network loading cost means the service performance is better (Beraldi et al., 2009). To find more requiring file replicas with less time and network loading cost, proper organizing the interconnection among the peers is one of the main approaches (Cao and Fujita, 2012).

The number of hops a search request walking from a peer to another before finding a file replica influences the time cost directly, which requires that the peers with requiring file replicas can be reached at less number of hops (Feng et al., 2011). In P2P network, a peer has more connections with the other peers and then it can reach the other peers with less number of hops. More connections means peers should send out more search requests and then the network loading is added rapidly (Hisiao and Su, 2012). Consequently, more connections between peers and less network loading are conflicted.

To solve the contradiction, reducing connections between peers and enhancing the functions of some peers is a widely accepted approach (Erola et al., 2011; Doukeridis et al., 2010). In the article (Garbacz et al., 2010), the architectures of P2P network with super peers are proposed utilizing multifunctional network technologies based on the heterogeneity of the nodes. Adding connections between peers and meanwhile selecting a portion of connections to send search requests is another feasible approach (Huang et al., 2010; Dhurandher et al., 2011). Based on the file sharing pattern exhibiting the power-law property, Kucharzak et al. (2011) proposes an optimization model in which searching an object in the P2P network efficiently takes a small constant hops by searching progressively and effectively on the similarity of peers.

In this study, the optimization to the interconnection among peers is taken as the only essential factor beforehand. The number of the connections between two peers will be continuously adjusted according to the variation of the active peers and the file replica distribution in the P2P network. Based on designing the format of interconnection relation and information about file replica searching, the particle swarm optimization algorithm is employed with the competitive neural network algorithm involved to reduce the time cost of the optimization procedure. Simulative experimental results demonstrate the enhanced efficiency and effectiveness of the proposed model.

Corresponding Author: Huang Yongsheng, Tangshan Key Laboratory of Informationization Technologies and Engineering Control, School of Management, Hebei United University, China
DESCRIPTION TO THE INTERCONNECTION AMONG PEERS AND INFORMATION ABOUT FILE SEARCHING

Interconnections among the peers: The set of the peers in the network can be denoted as \( P = \{ p_1, p_2, \ldots, p_n \} \). The connection relationship among the peers can be expressed by a matrix \([CN]_{n \times n}\):

\[
\begin{bmatrix}
  c_{11} & c_{12} & \cdots & c_{1n} \\
  c_{21} & c_{22} & \cdots & c_{2n} \\
  \vdots & \vdots & \ddots & \vdots \\
  c_{n1} & c_{n2} & \cdots & c_{nn}
\end{bmatrix}
\]

In the matrix, \( c_{ij} \) represents the connection relationship between the peer \( p_i \) and the peer \( p_j \). The value of \( c_{ij} \) is denoted by the following formula:

\[
c_{ij} = \begin{cases} 
1, & p_j is connected to p_i \\
0, & p_j is not connected to p_i
\end{cases}
\] (1)

In the P2P network, a peer \( p_i \) is connected to another peer \( p_j \) means \( p_j \) can dispatch or relay a search request to \( p_i \). As a result, it is not necessary that the connection between two peers is bidirectional. In other words, if \( p_i \) is connected to \( p_j \) does not mean \( p_j \) is connected to another peer \( p_i \). Consequently, both \( c_{ij} = c_{ji} \) and \( c_{ij} + c_{ji} \) are probable. The connection of a peer itself is meaningless and then \( c_{ii} \) is set to 0.

Dispatching procedure of the search request: The peers share their file resource in the P2P network. When a peer has not a file and wants to achieve the file, it will search the file from the other peers in the P2P network. If the file is found in some of the other peers, it can download the file from the peers and will acquire the file finally. Utilizing the technology of file block transmission, more file replicas are found, that is, more peers with searched file are found, means the file will be achieve more rapidly. Consequently, the peers with searched files are found more quickly and more peers with required file replicas are found mean the efficiency of the P2P network is higher (Endo et al., 2012).

In the running P2P network, when a peer needs to search a file, a search request will be sent to the peers which the peer is connected to. In order to widen the search scope, the peers which have received the search request should relay the search request to another peers which the peers are connected to. Correspondingly, the search scope is wider, more search requests will be relayed in the network and the load of the network will be heavier (Li et al., 2008). Consequently, the search scope should not be too wider. In the relaying procedure of the search request, a search request is relayed from a peer to another is named as a hop and the maximum number of hops is usually set beforehand when searching. On the other hand, for sake of avoiding relaying the search request repeatedly, a peer should not relay identical search request which it has received and relay before.

For convenient description, let \( ch = p_0, p_1, \ldots, p_n \) denote a relay sequential chain of a search request. In the denotation, \( 1 \leq c_1 < c_2 \ldots < c_n \leq n \). To express the relay procedure of a search request in the P2P network \( \text{len}(ch) \) is denoted as the length of the chain, that is, the number of the hops in the chain. Additionally, \( \text{seq}(ch, p_x, p_y) \) is put forward to represent the order of \( p_x \) and \( p_y \) in the sequential chain \( ch \):

\[
\text{seq}(ch, p_x, p_y) = \begin{cases} 
-1, & p_x is in front of p_y \\
1, & p_x is at the back of p_y
\end{cases}
\]

(2)

To a search request dispatched by the peer \( p_t \), initially, a chain set \( CH \) which consists of a group of relay sequential chains will be achieved after the search procedure ends. If the maximum number of hops for search request is set to \( h \) which is a positive integer, then the length of each chain in the set \( CH \) is equal of less than \( h \). In order to avoid relaying the search request repeatedly, for every two search chains \( ch_1 = p_0, p_1, \ldots, p_x \) and \( ch_2 = p_0, p_1, \ldots, p_y \) of a search request, then:

If \( p_t \) is equal to \( p_0 \), the element number in front of \( p_1 \) in the chain \( ch_x \) is identical with the element number in front of \( p_0 \) in the chain \( ch_y \) and meanwhile every two elements indicated as follows is identical:

\[
\begin{array}{cccc}
  P_0 & P_1 & \cdots & P_x \\
  \uparrow & \uparrow & \cdots & \uparrow \\
  P_0 & P_1 & \cdots & P_y
\end{array}
\]

In order to avoid calculating repeatedly, the Chain Abbreviation (CA) is defined as following:

If exists two chains \( ch_1 = p_0, p_1, \ldots, p_x, p_{xy}, \ldots, p_{y} \) and \( ch_2 = p_0, p_1, \ldots, p_y, p_{xy}, \ldots, p_{y} \) and the elements from the first one to \( p_x \) of the two chain are identical, the result of the calculation CA to \( ch_1 \) and \( ch_2 \) is two chains \( <p_0, p_1, \ldots, p_x, p_{xy}, \ldots, p_{y}> \) and \( <p_0, p_1, \ldots, p_y, p_{xy}, \ldots, p_{y}> \). In other words, To two chains with a certain number of identical elements in the front of them, the CA to them will delete the identical elements of one chain with the other unchanged.

If no element at the back of \( p_x \) in the chain \( ch_x \) is identical with the element in the chain \( ch_y \), the characteristic above means the search request is relayed
from $p_i$ to $p_n$ (or $p_n$) and two branches after $p_n$ (or $p_a$) appear and form the chains $c_n$ and $c_a$. To a search chain with the length less than the maximum number of hops $h$, the last element of the chain should have appeared in another search chain and consequently the chain does not expand to avoiding searching in the field having been searched. In order to meet the need of the algorithm below, a search chain will stop expanding when it searches to a peer (this peer will not be taken as element of the chain as it has been taken as element of another) having been searched by other chains.

To a chain set $CH$ consisting of all search chains of a search request, the abbreviated CH denoted as a set $abbr(CH)$ consists of elements of the chains achieved by CA to every two search chains. As a result, every two chains in the chains achieved by CA to every two search chains have not identical elements and then the elements in $abbr(CH)$ are the peers of a search request being relayed to without repeat.

**DESCRIPTION OF THE OPTIMIZATION FACTORS**

When a peer needs to acquire a file and dispatches a search request, more file replicas are found, the replicas are found in less hops and less peers are searched, the efficiency and the effectiveness of the P2P network are better.

To a file searched, the number of the replicas in all peers in the network is denoted as NR. As NR is varied corresponding to the differentiation of the files, the abstract value is improper to comprehensively represent the search ability of the P2P network and then the recall ratio to the replica search RRRS is put forward, which is calculated by:

$$RRRS = \frac{RF}{NR} \times 100\%$$

(3)

In the formula above, $RF$ is the replicas found by the procedure of the file searching.

The replicas being found in less hops means the needed file are found at the cost of less time expenditure and then the peer with needed file will achieve response and the file more rapidly. In a search chain $c_i$, the first element is the peer dispatching the search request initially. Corresponding, the sequential number of a peer in the chain $c_i$ can be used for the number of hops a replica is found. If the sequential number of $p_k$ in the chain $c_i$ is $k$, then:

$$\text{hop}(abbr(CH),p_k) = k - 1$$

(4)

$CH$ is a chain set according to a search request and $c_i$ is an element of the set. As more than one replicas are found according to a dispatched search request, the average value is utilized to evaluate the efficiency of the P2P network in file searching:

$$AH = \sum_{p_k \in \text{file}(CH)} \frac{\text{hop}(abbr(CH),p_k)}{|P_i|}$$

(5)

In the formula above, the denotation $P_i$ is the peer set which consists of the peers found the file replicas in for a search request and the set $CH$ is the search chain set according to the search request. $|P_i|$ is the element number in the set $P_i$.

In the P2P network, the number of the connected peers with every peer is large enough, a search request dispatched by a peer can be received all the other peers with less hops. For example, if every peer is connected by all the peers, the search request will arrive at all the other peers by one hop. However, too invalid connection will add the loading of the network when search requests dispatched. Consequently, better efficiency usually means less peers are searched with larger RRRS and smaller $AH$. The number of peers which a search request are relayed to is the element number of the set $abbr(CH)$ denoted as $|abbr(CH)|$.

To evaluate the efficiency of the P2P network, the factors of $|abbr(CH)|$, RRRS and $AH$ are interconnected and comprehensively utilized. The integrated factor for efficiency evaluation is represented as follows:

$$\text{IFactor} = \lambda_1 \times \text{RRRS} - \lambda_2 \times \text{AH} - \lambda_3 \times |abbr(CH)|$$

(6)

In the formula above, the parameters of $\lambda_1$, $\lambda_2$ and $\lambda_3$ are positive constant variation which can be chosen according to the importance rank of the three factors considering designated P2P network.

**PARTICLE SWARM OPTIMIZATION ALGORITHM WITH COMPETITIVE NEURAL NETWORK INVOLVED**

**Optimization to the interconnection among the peers with PSO algorithm:** As description above, the interconnection among the peers in a P2P network can be expressed by a matrix $[CN]_{i,n}$. Utilizing the definitions and formulas above, the IFactor value can be calculated by the corresponding $[CN]_{i,n}$. As the matrix $[CN]_{i,n}$ is varied, the value of IFactor will changed accordingly.

To search the matrix $[CN]_{i,n}$ which can achieve optimized IFactor value, the particle swarm optimization
(PSC) algorithm (Zheng and Liu, 2010) is used to adjust the matrix \([CN_\text{en}]\) and optimize the interconnection among the peer. In this proposed optimization algorithm, the iteration formulas are expressed as follows:

\[
\begin{align*}
\tau(t+1) &= \tau(t) + c_1 \tau(t) + c_2 \chi(t) \cdot (CN(t))_ng - (CN(t))_ng(t) - (CN(t))_n(t) - (CN(t))_n(t) \\
\text{and} \quad (CN(t+1))_n &= g[(CN(t))n + \tau(t+1)]
\end{align*}
\]

(7)
(8)

In the formulas above, \((CN(t))_n\) and \((CN(t+1))_n\) are the matrices of \([CN_\text{en}]\) which indicate the interconnection among the peers at the \(t\)th and \((t+1)\)th iteration of the iterating particle, respectively. Accordingly, \((\tau(t))_n\) and \((\tau(t+1))_n\) are the variations at the \(t\)th and \((t+1)\)th iteration, respectively. \(P(CN(t))_n\) is the matrix which achieves the extreme value of the IFactor until the current iteration for this iterating particle and \(g[(CN(t))_n\) is the matrix which achieves the extreme value of the IFactor until the current iteration for the iterating particle group. \(c_1\) and \(c_2\) are the learning ratios which are constant number and then \(r_1(t)\) and \(r_2(t)\) are random number whose value are less than and larger than 0. The addition of the matrices in the proposed algorithm is redefined as follows:

\[
(CN_{\text{en}})_n + (CN_{\text{en}})_n = (CN_{\text{en}} + CN_{\text{en}})_n
\]

(9)

\[
(CN_{\text{en}})_n + CN_{\text{en}} = \begin{cases} 
0, & \text{both } CN_{\text{en}} \text{ and } CN_{\text{en}} \text{ are } 0 \text{ or } 1 \text{ which is one of } CN_{\text{en}} \text{ and } CN_{\text{en}} \text{ is } 0, \text{ and the other is } 1 \\
1, & \text{one of } CN_{\text{en}} \text{ and } CN_{\text{en}} \text{ is } 0, \text{ and the other is } 1 
\end{cases}
\]

(10)

To the multiplication between a matrix and a number in the iteration formulas, the calculation is redefined as:

\[
\delta(c_r, CN_{\text{en}}) = \begin{cases} 
0, & c_r, CN_{\text{en}} < \frac{\epsilon}{2} \\
1, & c_r, CN_{\text{en}} \geq \frac{\epsilon}{2}
\end{cases}
\]

(11)

(12)

In the P2P network, the optimization procedure to the interconnection among the peers should be re-executed repeatedly for two reasons. Firstly, a peer of the network is not always active. In other words, not all peers are active. In a designated time period, the optimization is directed against the peers who are active at this time. Secondly, the distribution of the sharable files and their replicas will vary with the running of the network and then the optimal interconnection scheme will vary accordingly. On the other hand, the optimization is a large time cost procedure because of the complexity of the PSO algorithm. In order to enhance the real-time effectiveness of the optimization, the Competitive Neural Network (CNN) algorithm is chosen to involve the optimization procedure.

**Matrix alteration for the input of the neural network:** In P2P network, the active peers are usually part of the usable peers. Consequently, the optimization only for the interconnections among the active peers is useful in real-time running P2P network. In other words, the real-time optimization is only pointed to a portion of the matrix \([CN_\text{en}]\). For example, the active peer set is \({P_1, P_2, P_3, P_4, P_5}\) the columns and rows with order number 3, 5, 6, 8, 9 are extracted as follows to be optimized further:

\[
\begin{bmatrix}
\begin{array}{cccc}
CN_{13} & CN_{15} & CN_{16} & CN_{18} & CN_{19} \\
CN_{33} & CN_{35} & CN_{36} & CN_{38} & CN_{39} \\
CN_{53} & CN_{55} & CN_{56} & CN_{58} & CN_{59} \\
CN_{63} & CN_{65} & CN_{66} & CN_{68} & CN_{69} \\
CN_{83} & CN_{85} & CN_{86} & CN_{88} & CN_{89} \\
CN_{93} & CN_{95} & CN_{96} & CN_{98} & CN_{99}
\end{array}
\end{bmatrix}
\]

As the input of the neural network is a vector, the matrix is turned into a vector by connecting the rows of the matrix one by one in order. Then, the matrix listed above can be turned into the following vector:

\[
\begin{bmatrix}
CN_{13}, CN_{15}, ..., CN_{18}, CN_{19}, CN_{33}, CN_{35}, ..., CN_{38}, CN_{39}, CN_{53}, CN_{55}, ..., CN_{58}, CN_{59}, CN_{63}, CN_{65}, ..., CN_{68}, CN_{69}, CN_{83}, CN_{85}, ..., CN_{88}, CN_{89}, CN_{93}, CN_{95}, ..., CN_{98}, CN_{99}
\end{bmatrix}
\]

For the sake of convenient description, the matrix \([CN_\text{en}]\) which is pointed to all available peers, is called as the matrix for the entire peers (MEP). And then the matrix for the active peers is called as MAP. Correspondingly, the vectors transformed from the matrices are named as the vector for the entire peers (VEP) and the vector for the active peers (VAP).

As distinct active peers result in distinct VAP, the vector adjustment is put forward to better utilize the optimization information. To two active peer sets \(P_1\) and \(P_2\), the VAP of \(P_i\) can be adjusted to be used to the interconnection optimization of \(P_i\) if the element number of the intersection between \(P_1\) and \(P_2\) is large enough. The adjustment procedure is divided into two steps. Firstly, the adjusted VAP (AVAP) of \(P_i\) for the interconnection optimization of \(P_i\) consists of the elements identical to \(P_i\). The elements of the vector are divided into two parts. The first part consists of the elements according to the peers in the intersection between \(P_1\) and \(P_2\). The second consists of the elements according to the peers in \(P_i\) other than the peers in the intersection between \(P_1\) and \(P_2\). Secondly, the values of the two parts in AVAP are set separately. The values of
the elements in the first part are set by the values of the according elements in the VAP of $P_2$. The values of the elements in the second part are all set to 0.

Figure 1 shows an example that indicates a AVAP transformed from the VAP of $P_2$ according to $P_1$. In the example, $P_1 = \{p_2, p_3, p_6, p_8\}$ and $P_2 = \{p_3, p_5, p_6, p_8\}$. The elements and their values in the VAP of $P_2$ are listed in the first row and the second row of the Figure and the fifth and the forth row show the elements and their values in the transformed AVAP. The arrows in the third row point to the partial elements of the AVAP whose value are set by the according value of the VAP of $P_1$. The elements with value 0 are the elements in the second part, whose values are set to 0 directly.

**The optimization with competitive neural network algorithm involved:** With the running of the P2P network, the active peers in the network vary continuously. The optimization procedure will be executed time by time again. As a result, a series of active peers sets and the corresponding VAP with optimized interconnection information will be achieved. With the variation of the active peers and the file replicas distribution, the P2P system can not bear frequent execution of the optimization procedure for the complex calculation of the PSO algorithm. To reduce the calculation quantity of the optimization procedure, the competitive neural network algorithm is selected to involve the optimization procedure.

As shown in Fig. 2, the competitive neural network is divided into two layers (Meyer-Base and Thammler, 2008). The first layer is called as the feed forward layer, in which the information of the active peers in all available peers is taken as the input by turning them into an active vector (AP). In AP, every element is corresponding to an available peer and the value of the element is defined by the following formula:

$$v_{ap} = \begin{cases} 0, & \text{the corresponding peer is not active} \\ 1, & \text{the corresponding peer is active} \end{cases}$$

As both the active peers and the file replicas distribution act on the optimization of the interconnection among the active peers, the variation to each of them means the adjustment to the optimized interconnection. Consequently, more similar active peers and the file replicas distribution result in more similar optimized interconnection. In first layer, $|AP|$ is the element number in AP and then the input is a $|AP|$ dimension vector. $aw$, is the weight matrix which consists of S vectors of past APs with adjustment.

The variation to the active peers can be reflected by AP. As the variation to file replicas distribution is gradually implemented with time eclipsing, the adjusted AP (AAP) is put forward to embody them. The value of an element in AAP is corresponding to that in AA and adjusting by the following formula:
\[ av_p = \frac{t_i v_p}{t_p} \] (14)

\( t_i \) is an element of the order number sequence \( TS = (1, 2, 3, \ldots) \) in which an element \( i \) denotes that the optimization procedure has been executed \( i \) times. With the optimization executing one by one time, more order numbers are added to the sequence. In the formula, \( t_i \) is the order number generated by the current optimization procedure and \( t_p \) is the order number generated by the past optimization procedure in which \( V_p \) appeared. Thus, the output of the first layer, \( o_i \), expresses the similarities between the input AP and every one of the selected AAPs which are formed by adjusting the selected past AP.

The second layer of the competitive neural network is called as the competitive layer, by which the AAP most similar with the input AP is found. The weight matrix \( W \) is set by the following formula (Nie and Cao, 2009):

\[ w_{ij} = \begin{cases} 1, & \text{if } i = j \text{, otherwise } 0 \leq w_{ij} \leq 1 \end{cases} \] (15)

\( w_{ij} \) is the element of \( W \) at the intersection of the \( i \)th row and the \( j \)th column. The first layer output \( o_i \) is used to initialize the second layer, that is to say:

\[ o_1 (0) = o_i \] (16)

Then, the output of the second layer is updated according to the following recurrence relation:

\[ a_1 (t + 1) = \text{poslin}(W a_1 (t)) \] (17)

With the competitive neural network involved, the APs and the corresponding optimized VAP achieved by the optimization are recorded along with the order number and used to the following optimization. When a new AP is waiting to be optimized, a new order number is generated and a certain number of the past APs are chosen and adjusted to AAPs and then the AAP most similar to the new AP is achieved by the competitive neural network algorithm. According to analysis above, the past optimized VAP corresponding to the achieved AAP is approximate to the optimized VAP of the new AP. The PSO algorithm is initialized by the following formula:

\[ CN(i) = \text{MAP(VAP)} \] (18)

MAP(VAP) denotes the MAP according to the VAP. As the achieved AAP is approximate to the optimized VAP of the new AP, the optimized VAP according to the new AP will be achieved with less times of iteration and consequently, the PSO with the competitive neural network involved can enhance the efficiency of the optimization procedure.

**Simulative experiments and analysis:** For P2P network, the efficiency comparison with the models out the renewed model involved is usually taken into account to demonstrate the efficiency of a renewed model (Sendhil and Nagarajan, 2012; Ramachandran and Sikdar, 2010). In this study, simulative experiments are employed in order to testify the efficiency and the effectiveness of the proposed model. In the optimization procedure with the PSO algorithm, a series of interconnection schemes are selected to iterate to achieve optimized scheme.

The comparison between the optimized scheme and the series of interconnection schemes before achieving optimized scheme can indicate the effectiveness of the proposed model. The results in Fig. 3 come from a series of simulative experiments. In the experiments, the P2P networks with 300, 500, 700, 900, 1100, 1300, 1500, 1700 available peers have been simulated. For every simulative P2P network, the optimization procedure has been executed 200 times accompanying with variation of the active peers and the file replicas distribution. The IFactor is calculated in every optimization procedure with \( \lambda_1, \lambda_2 \) and \( \lambda_3 \) being set to 100, 1, 1, respectively (as RRRS is a percentage, \( \lambda_1 \) is set to 100). And meantime, the IFactors corresponding to the series of interconnection scheme before optimized scheme for every optimization procedure has been recorded and the mean of the IFactors has been calculated.

In Fig. 3, the IFactor value for every P2P network with designated number of available peers is the average value of the IFactors according to 200 times of optimization procedure execution. Correspondingly, the mean IFactor value for every P2P network with designated number of available peers is the average value of the mean of the IFactors generated in the 200 times of optimization procedure execution. An effective optimization model usually means an obvious efficiency enhancement in comparison to the past proposed model (Szekeres et al., 2011). The experimental results shown in Fig. 3 indicate the IFactor Value is larger than the Mean IFactor Value for every P2P network with designated number of available peers and he IFactor Value is much larger than the Mean IFactor Value with the number of available peers increasing, which means the proposed model can achieve better effectiveness and the effectiveness is much better with the number of available peers increasing comparatively. It is common that the performance will decrease with the scale of the P2P network increasing.
(Sharifi and Khorsandi, 2013). However, the IFactor Value almost does not reduce with the number of available peers increasing, demonstrating the effectiveness stability of the proposed model is almost irrelevant to the number of available peers.

Frequent execution of the optimization accompanying with the variation of the active peers and the file replicas distribution requires the optimization procedure to be implemented rapidly. The efficiency of the proposed model with the competitive neural network algorithm is testified by simulative experiments.

For sake of showing the efficiency of the proposed model, the time cost of every optimization procedure in the simulative experiments relative to Fig. 3 is recorded. The mean value corresponding to designated number of available peers are calculated and utilizing these results the efficiency comparison between the proposed model with the competitive neural network algorithm (With CNN) and the optimization only utilizing the PSO algorithm (Without CNN) are demonstrated in Fig. 4. The experimental results indicate that the time cost with CNN is smaller than that of without CNN for every simulative P2P network with designated number of available peers and accompanying with the number of the available peers increasing the time cost with CNN is much smaller and almost stable comparative to that of without CNN, which demonstrates the efficiency of the proposed model.

CONCLUSION

In P2P network, the interconnection relation among peers is the decisive factor to the performance of the network by influencing the time cost and the recall ratio of file replica searching. In this study, the interconnection among peers and the information relative to the performance of the P2P network are formatted by designated description. And then an optimization algorithm utilizing the PSO algorithm with the competitive neural network algorithm involving is put forward based on the variation characteristics about the active peers and the file replicas distribution of the P2P network. A series of simulative experiments have been implemented and the experimental results indicate the proposed model can achieve better efficiency and effectiveness for the optimization of the P2P network.

ACKNOWLEDGMENT

This study was funded by the Science and Technology Department of Tangshan City (No.12140201 A-7).
REFERENCES


