Chiang Mai J. Sci. 2014; 41(5.2) : 1392-1408
http://epg.science.cmu.ac.th/ejournal/
Contributed Paper

Patrol Packet Algorithm: A Modified Flooding Technique for High Performance Search

Monlica Wattana and Pattarasinee Bhattarakosol*
Innovative Network & Software Engineering Technology Laboratory, Department of Mathematics and Computer Science, Faculty of Science, Chulalongkorn University, Bangkok 10330, Thailand.
*Author for correspondence; e-mail: pattarasinee.b@chula.ac.th

Received: 7 August 2012
Accepted: 21 May 2013

ABSTRACT

Since the Internet is a large information for people all over the world, the search mechanism is a most important tool for the Internet users. Currently, many methods are proposed to determine the resource location, such as Flooding algorithm and Random walk. Unfortunately, the performance of these algorithms has some defects that cannot provide a full and correct list that the users' require. Therefore, this paper proposes an effective search algorithm, called Patrol Packet (PTP) algorithm. PTP was developed under the implementation of the Information Search Protocol (ISP) where the search criteria can be determined. Thus, the results obtained from the PTP algorithm can satisfy users' requirement with a quick response time. Since the PTP algorithm is obtained from a modification of the flooding algorithm, the number of distributed bytes over the communication channel is reduced when compared to the original algorithm. Consequently, congestion can be minimized. In addition, the performance testing of the PTP shows that this algorithm is suitable for distributed search where a large number of query messages spreads over the communication channel.

Keywords: search algorithm, flooding algorithm, distributed system

1. INTRODUCTION

Presently, the Internet is an important resource for people all over the world; users can access the vast amounts of information from various locations virtually without limitation. However, this information is stored in platforms using different technologies that are based on the owners' policies. Thus, obtaining access to necessary information over the Internet can be complicated and time consuming. In addition, the returned results may not be as accurate as expected. Therefore, searching for the required information needs a qualified search mechanism to filter unwanted search results. Thus, many search mechanisms, including search protocols, are implemented to shorten the search time and cost. The efficiency of these implementations can be evaluated using the response time and users' satisfaction of the returned results [1]. In addition the number of queries and the size of each query can be applied as performance indicators [2]

since a large amount of queries in the communication channel can cause congestion. Therefore, the sending query and its size must be appropriate for the communication situation.

In order to obtain a suitable search result, a protocol named Information Search Protocol (ISP) was proposed in the year 2006 [3]. This ISP enables users to enter a part of the required URL and keywords, e.g. "mit.edu" and "research" where the result of search appears as "web.mit.edu/research/". Thus, the ISP narrows the search and reduces the search time for the required information results list. As a result, the ISP and its search mechanism can increase the efficiency of the search engine, or so-called smart protocol, for specific content searches over the Internet.

Considering the search mechanism of the ISP, every query message is broadcasted to every database server to find the full pathname of the required URL. The search results are then sent back to the client. This broadcast technique of all queries; the flooding technique is started from the client to its neighbors and continuously sends to other neighbors of the receiving sites. Even though these messages are not re-sent to the client, some neighbors may receive it more than once. Consequently, congestion occurs when there are a large number of queries at one time.

In order to solve the problems mentioned above, this paper proposes a performance search technique that is called the Patrol Packet (PTP) Algorithm. The proposed algorithm is implemented to support the ISP's search process in order to guarantee the search results. This algorithm controls the sending queries to all nodes over the distributed sources so the congestion can be avoided and the response time of the search is reduced.

2. RELATED WORKS

There are a number of systems that are distributed systems which share resources distributed over the Internet, such as peer-to-peer systems [4], naming systems, and distributed databases [1]. Thus, looking for a source is a vital process in a distributed system. The many proposed mechanisms to send the query messages to the correct resources' locations must minimize the number of transferring query messages. Unfortunately, various search techniques have different trade-offs in their desired characteristics [5, 6].

The search techniques of distributed systems are basically classified into two search methods: the informed search, and the blind search [2, 6]. In the informed search method, each node has routing information to refer to the location of the required data because each node collects some routing information that contains parameters for selecting the appropriate neighbor nodes to relay query messages to.

There are various schemes in the informed search. Regardless of the scheme, most of the informed search techniques use distributed hash tables (DHTs), which are lookup tables of distributed resources (i.e. Chord [7]). Using this search method has many benefits, e.g. the reply time of the search message is less than the blind search because the number of sending queries is smaller than the blind search since only the selected servers will receive the queries. The informed search requires some search information for choosing the servers, so if the information is incorrect then the search cannot succeed. Another defect of this search method is that it has a high cost to maintain the routing information. The implementing cost of the informed search is to prepare the parameter of the routing information before starting the search.

In contrast to the informed search, in the blind search, each server node does not collect information that supports the search process. This method uses flooding techniques for sending queries to other servers over the network. For example, Breadth First Search (BFS) is used in peer-to-peer systems. A node sends a query message to its no-received message neighbors. The starting node initializes a time-to-live (TTL) value in the query message. The TTL is decremented at each hop, with the packet being discarded when the counter becomes zero. The TTL is used for limiting flooding query messages. The other method is Random Breadth First Search (RBFS) [8], which improves the flooding technique by randomly selecting a portion of its neighbors to send the query messages to. RBFS reduces number of query messages that is passed on in the network. Still another technique is Normalized Flooding [9], which uses the semi-flooding technique because it sends the query messages to a set of selected neighbors. The number of selected neighbors (d) is the minimum degree of a node in the network. If a server has neighbors degrees higher than d, the message will be delivered only to d nodes in its neighborhood that are randomly selected. Similar to the flooding technique, the implementation of Iterative Deeping [2, 5, 10] uses a depth-first search to set the depth in the starting search. The nodes relay the query message using a breadth-first search. The search finishes when the depth reaches the maximum limit or when the server has found the required data.

Although most of the blind search mechanisms have applied the concept of the flooding technique, there are some other alternatives to that can be applied. One of these choices is the Random walk method [5]. The nodes in this method forward one query message to their neighbors that are randomly chosen from all neighbors. The forwarding message is sent to a randomly chosen neighbor in each step. The query message is called a walker. The positive result of this method is that it reduces the number of flooding messages; however, there is a delay in returning results to users. After the installation of the Random walk method, the k-walks algorithm is proposed [11]. With this algorithm, the server sends k query messages to an equal number of randomly chosen neighbors and then each neighbor sends out one message to the next neighbor that is randomly selected. Another example of the random walk technique is a two-level random walk [12]. There are two policies for the random walk. At the first level, the server selects k_1 random walk with $TTL_1 = l_1$. When the TTL_1 is zero at a particular server, the second policy will be started. The server selects k_2 random walk with $TTL_2 = l_2$. The random search method's benefit is to reduce the query message's overhead. However, the random search does not reach all of servers in the system; it selects some nodes that have a higher probability of finding the required data. Thus, query message may not arrive at the right location and the required information might not be found.

In order to reduce the number of sending query messages, a hierarchical mechanism is considered, such as a naming system. Most of the naming systems use a hierarchical mechanism to search for an IP address. A well-known naming system is the Domain Name System (DNS) [13-15], which maps a host name to a numerical IP address. The query messages are sent to the parent and children of the node. Thus, when sending a query message, the sending overhead can be reduced while messages are sent to all nodes. Unfortunately, this hierarchical search mechanism takes a longer time than the

flooding mechanism when the required data exists in other branches of the hierarchical structure.

Considering the existing search techniques that have been mention above, no technique sends the query message to all nodes on the system except the flooding and hierarchical mechanisms. If the query messages reach all nodes, the correctness of the results can be guaranteed. Due to the fact that the required information is always stored in different branches, the hierarchical model is not the most suitable search method, for the reasons mentioned above. Thus, the flooding technique is applied in this research. In addition, applying the flooding technique is flexible because its process can start without preparing the routing information.

As a result of applying the flooding mechanism, this technique can cause congestion based on the distribution of query messages. Therefore, the suitable paths must be determined before the queries are delivered. In order to find a suitable path, the original concept of using Patrol Packet (PTP) algorithm for finding the sending path of a query message had been proposed in [16]. Thus, this paper has objective to prove that the proposed concept of [16] is efficient as expected that the re-bounding of the queries can be eliminated. Furthermore, the sending overhead and updating the parameter of routing information problems are also solved. According to these objectives, the full system is implemented and evaluated as the following details.

3. Overview of Information Search Protocol (ISP)

Information Searching Protocol (ISP) [3] is a search protocol that aids users who know a partial pathname of a required URL and keywords. As mentioned earlier, if a user enters "mit.edu" and "research", the result of searching shows as web pages with the word "research" in the "web.mit.edu". The architecture of the ISP consists of two main processes. The first process is called the Embedded Agent for Information Search Protocol (EAISP). The second process performed in the search engine system, is called the Global Search Engine System (GSES).

The architecture of the ISP system is presented in Figure 1 and the details of the ISP are explained below.

3.1 Embedded Agent for Information Search Protocol (EAISP)

This process is located at the client site where a required URL and keywords are received from a user, and it will be encapsulated into the ISP format before transferring to the second process. The format of the ISP is presented in Figure 2.

According to Figure 2, the length of the ISP is a dynamic value depending on the number of keywords and the result list. Moreover, there are two terminate values to indicate the end of the query payload, or the end of the result payload. When a result is returned to the client, the result payload is extracted and the full pathname of the URL(s) is sent to the client's user interface of the browser. Each attribute in Figure 2 is described in Table 1.

3.2 Global Search Engine System (GSES)

The GSES consists of three modules: Protocol Interpretation Module (PIM), Search Module (SM), and Routing Module (RM). In addition, a Global Database (GDB) storing the full path of URLs and containing the keywords is implemented. Following are descriptions of the functions of each module.

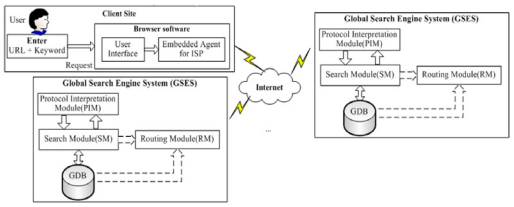

Figure 1. The architecture of the ISP.

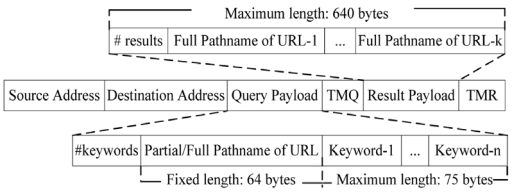

Figure 2. The ISP format.

Table 1. Syntactic and semantics of the ISP format.

Attribute name	Length (byte)	Remark
Source Address	4	The IP address of the user system
Destination Address	4	The IP address of the database search engine
#Keyword	1	Number of search keyword, usually is equal to the value of #keywords. Total number of keywords must not exceed 5 words.
Partial/Full Pathname of URL	64	Some parts or full pathname of the required URL. Users can enter only one value.
Keyword-*i*	15	Search keywords
TMQ	1	Terminate key, indicates the end of query text, fixed value in hexadecimal is 'FF'.
#result	1	The total number of URLs listed from the search mechanism. The value of this attribute is k.
Full Pathname of each URL	64	The pathname of the web, may have more than one result, total number of the URLs is not exceed 10 URLs
TMR	1	Terminate key, indicates the end of result text, fixed value in hexadecimal is 'FF'.

3.2.1 Protocol interpretation module (PIM)

This module is responsible for extracting the ISP in order to obtain the partial pathname of a URL and keywords of the required content. After obtaining the extraction result, this result is used to create a search command in an SQL statement. Additionally, when the PIM receives the required data from the SM, these data will be encapsulated and sent back to the client.

3.2.2 Search module (SM)

This module receives the SQL statement from the PIM. The results for the retrieving process can be a full pathname of a URL or a set of full pathnames of URLs. However, there is a chance that the required URL does not exist in the GDB. In this case, the SM will automatically transfer to the Routing Module; otherwise, the results are sent to the PIM.

3.2.3 Routing module (RM)

When the required URL does not exist

in the GDB, the SM will send a message to the RM. The RM will retrieve IP addresses of other GSESs from the GDB, create ISPs with new details of the destinations, and transfer the ISP to other GSESs that have links to the current GSES. The boundary of sending messages is controlled by the time to leave (TTL) technique, where the TTL value will be reduced by one until zero is reached and the sending process is terminated. Then, the last GSES will send the ISP with a NULL-result back to the client.

3.2.4 Global database (GDB)

Each GSES has an installed database named the Global Database (GDB). The data in the GDB consists of fields that indicate full pathnames of URLs, the IP address of each URL, and defined keywords under each URL. Moreover, the GDB has the addresses of the GSESs that have links to this GSES.

4. STATEMENT OF THE PROBLEM

Since the ISP is extracted in the PIM for a partial pathname of the required URL with specific keywords, this information will be sent to the SM for retrieving process. Nevertheless, some required URLs may not available in the GDB; then, the SM will send a message to the RM to retrieve IP addresses of other nearby GSESs from the GDB. Once the new IP addresses are obtained, new ISPs are created and sent out to the new destinations.

Consider the process of the RM; the flooding technique. This process can cause congestion and lead to a delay in receiving requested information. Thus, this paper proposes a new search mechanism that can avoid these problems.

5. PROPOSED METHOD

In order to eliminate congestion and delay problems of the ISP, the ISP and GSES are adapted to work on the new routing algorithm. The routing algorithm uses a small packet to check the status of the GSES. The details of the proposed routing algorithm are elaborated below.

5.1 Small Packet Format

In order to find the proper path of the ISP, a small packet named the Patrol Packet (PTP) is proposed. The PTP runs above UDP to support the routing algorithm for transferring the ISP. Therefore, the transferring speed is faster. The PTP consists of fields to indicate the sending and receiving status of the ISP over the sending path to protect the resubmitting of the ISPs to the GSES. The format of the PTP is shown in Figure 3 and the meaning of each attribute is described in Table 2.

Table 2. Syntactic and semantic of the PTP format.

Attribute name	Length	Remark
Source Address	(byte)	IP address of the client.
Destination Address	4	IP address of the destination GSES.
Sender Address	4	IP address of the source GSES.
Query ID	4	Number of search identification
Status	1 1	Status of Sending or Receiving; defined as0: PTP is sent , 1: PTP is received with permission to send ISP
Option	1	Null bits

According to Table 2, the length of the PTP is as small as 16 bytes over the Internet. Moreover, determining paths before sending by the PTP definitely reduces the number of sending ISPs through the network. Consequently, the congestion problem from the ISP and the rebound problem are eliminated.

5.2 Modification of The ISP

In order to protect against the re-bounding problem, two new fields are added into the ISP, as shown in Figure 4. The meanings of these two fields are presented in Table 3.

Table 3. Syntactic and semantic of the new attribute.

Attribute name	Length (byte)	Remark
Query ID	1	Number of search identification
Flag	1	Status of ISP transfer; defined as1: the sender is client, thus GSES does not send PTP and can send the ISP.0: the sender is GSES, thus GSES must send PTP to check status of ISP.

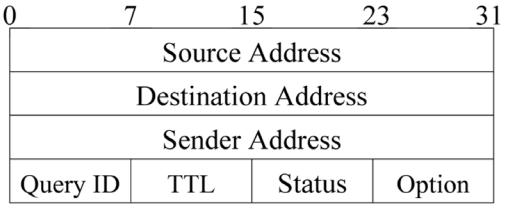

Figure 3. The format of the PTP.

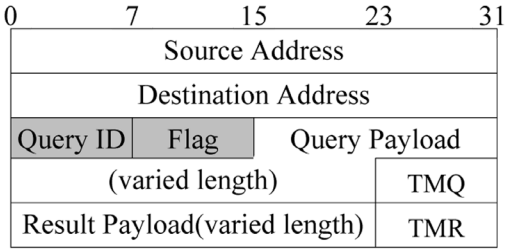

Figure 4. The new ISP format.

5.3 Modified Global Search Engine System (MGSES)

As in the ISP format, the modification to the GSES must be performed. The new GSES, called as the Modified Global Search Engine System (MGSES), is added to Classified Module (CM) and Modified Routing Module (MRM), and Modified Global Database (MGDB). The architecture of the MGSES is displayed in Figure 5. Each module is described as follows.

5.3.1 Classified module (CM)

The CM is responsible for classifying a PTP and an ISP using the length of the packet. If the length of the packet is 16 bytes (128 bits), the packet will be sent to the RM. If the length of the packet is greater than16 bytes, then the packet will be sent to the PIM. Otherwise, the packet is dropped.

5.3.2 Modified routing module (MRM)

The MRM is responsible for finding the right path to send the ISP packet; with the PTP packet. Therefore, there are two different situations to be managed by this module.

Situation 1: This situation manages the path for an ISP. This module will start its

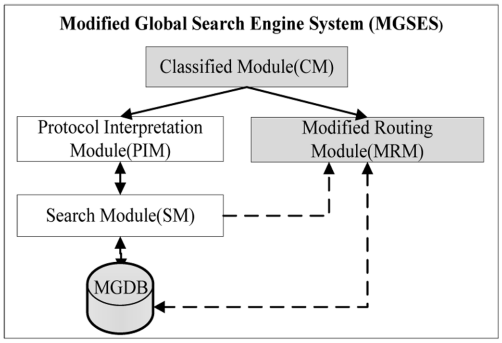

Figure 5. The modification of GSES.

process when the required URL does not exist in the MGDB. Under this circumstance, the ISP must be sent out to other MGSESs. The ISP is sent directly to the client to all neighbor MGSESs next from the current MGSES. Otherwise, PTP packets must be created and sent out to find the travelling path and the second situation occurs.

Situation 2: This situation occurs to manage the sending PTPs. Since a PTP is used to find the sending path of the ISPs without a redundant sending process, a flag value is set up to determine the suitability of the available paths. Thus, the value of the status field can be "0" or "1". The value of the status field of PTP is set to "0" when it is used for the paths being looked up, and the value of the status field will change to "1" when the path is selected by the received MGSES.

5.3.3 Modified global database (MGDB)

The MGDB contains two groups of data in the database. The first group is the list of the searched URLs and keywords. The second group is the routing path of the sending ISP. The data in the first group consists of two tables that are the URL table (URL_T) and the keyword table (KEYWORD_T). The URL_T indicates the full pathname of the URL names (URL), and the KEYWORD_T defines URL_ID and keywords under each URL.

The data in the second group consists of three tables; the neighbors table (NEIGHBOR_T), the routing table (ROUTING_T), and the ISP table (ISP_T). The NEIGHBOR_T indicates the IP address of the neighbors (NEIGHBOR_ADDR). The ROUTING_T indicates ISP identification (ISP_ID), the neighbor identification (NEIGHBOR_ID), the status of a small packet (STATUS), and time stamp of arrived small packet. The ISP_T indicates the query's identification of the ISP (QUERY_ID), the client address (SOURCE_ADDR), the query of a URL and keyword (QUERY), and the time stamp of the ISP (TIME_STAMP).

5.3.4 Example of the PTP algorithm

This example describes the sending process of a PTP by the PTP algorithm of the MRM where there are four MGSES in the system, which will be called A, B, C, and D. Each step of the PTP algorithm is explained in Figure 6. Details of all processes are explained below.

Firstly, the ISP from the client is sent to the nearest MGSES, A (follow dash-line (1)). The SM searches for the required URL and updates the ISP_T in the MGDB. If the MGDB has the required URL, then A sends the result back to the client.

If the required URL is not found, the value of the flag field in the ISP is changed from "1" to "0" and the ISP is sent to A's neighbors which are B and C. In the example, the required URL is not found on A, so the ISP is sent to B and C (follow dash-line (2) and (3)).

After B and C receive the ISP, the search for the required URL and the ISP_T in their MGDB will be updated. In the

example in Figure 6, B cannot find the required URL in its MGDB so the RM of B must send the PTP to B's neighbors, including A and C (follow dash-line (4) and (5)). Based on the information from A and C, there will be no reply back to B from both sites. Thus, the ISPs will not be resubmitted to both A and C.

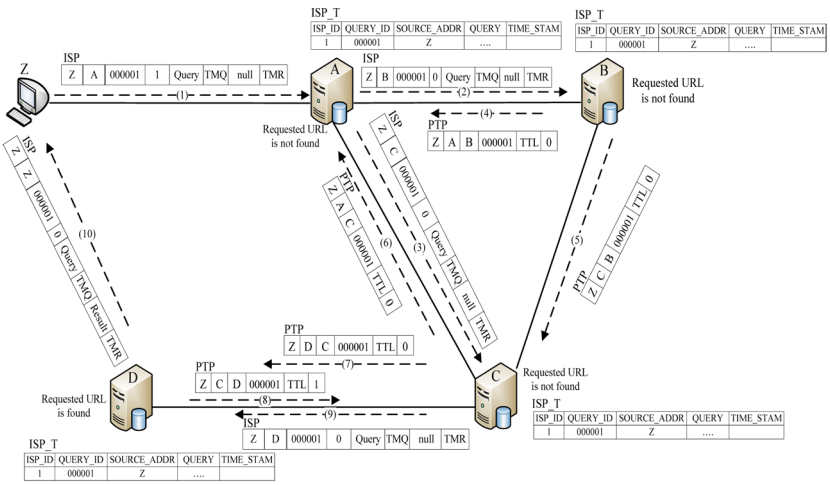

Figure 6. Example of transferring an ISP using the PTP algorithm.

In the situation of C, after C received the ISP from A then the ISP_T is updated and the search mechanism is performed. Based on Figure 6, C does not contain the required URL. Thus, the PTP is initialized and sent to C's neighbors, A and D (follow dash-line (6) and (7)). As in the situation of B, A will not reply for the receiving PTP. Thus, there is only one site, D, that C will receive the reliable PTP with the value of status field equals to "1" (follow dash-line (8)). Then, C will send the ISP to D (follow dash-line (9)). Under this scenario, D contains the required URL. Therefore, the search results will be inserted into the ISP and this ISP will be sent back to the client (follow dash-line (10)).

6. PERFORMANCE EVALUATION

In order to evaluate the performance of the proposed PTP algorithm, the comparisons between the PTP algorithm and two standard methods are performed. The two standard methods are flooding and hierarchical mechanisms. The evaluation among these three mechanisms is performed by simulation method. In addition, the theoretical analysis is performed to confirm the efficiency of the PTP algorithm. Details of these evaluations are presented below.

6.1 Simulation and Result

In this section, the simulation is performed by implementing the three routing algorithms mentioned above. The three algorithms are simulated on a virtual machine environment. The implementation for the evaluation system environment is performed on an HP 2 Quad Cores with XEON Processors and 16 GB main memory running Ubuntu 10.04 Desktop. Each guest on yhe VM is implemented on Intel Core 2 Duo 2.66GHz Processors and 512 MB main memory running Ubuntu 10.04 Desktop. The database management system employs MySQL Server version 5.1, and the phpMyAdmin running on Apache2 Web server is used to deal with

MySQL Server. All mechanisms maintaining the system are developed using Java.

There are two simulations. The first simulation is to increase the number of search engines for a client. n nodes of the search engine are implemented for n = 3, 5, 7, 9, 11, 13 and 15. The second simulation is to increase the number of clients. m clients are performed for m = 1, 3, 5, 7, 9, 11, 13 and 15. There are ten nodes of search engines that are implemented on this simulation

In the proposed solution, this search engine is named as MGSES while others use only the global search engine system (GSES). Under each search engine, the URLs and keywords are recorded. Each distributed GSES/MGSES contains 400 records of URLs and keywords.

Moreover, two search engines, MGSES and GSESs, are connected with the mesh topology for simulating the flooding and the PTP algorithms while the hierarchical structure is implemented for the hierarchical algorithm. Three routing algorithms are tested using ten sets of required URLs and keywords. These required URLs and keywords are encapsulated in the ISP at the client site and delivered to the MGSES or GSES.

The performance metrics in the two simulations are the number of bytes and the response time. The number of bytes refers to bytes of all packets that are sent to every search engine node. The response time is the time interval counting from when the first packet was issued from the client until the results were returned. The performance metrics are measured by programming.

The number of bytes

The first simulation uses ten sets of required URLs and keywords. In addition, the simulation system will record all sending bytes over the network. Figure 7 shows the comparison of the average number of bytes among three algorithms at n nodes of MGSES/GSES. The result shows that the number of bytes of the flooding algorithm is more than the number of bytes of the proposed algorithm and the number of bytes of the hierarchical algorithm is minimal. When the number of search engine nodes is increased, the numbers of bytes of the flooding algorithm are much higher than the number of bytes of the PTP algorithm. The proposed algorithm can reduce the number of bytes of the flooding algorithm by more than 50%. The number of sent bytes by the hierarchical algorithm is slightly increased when the number of GSES nodes is raised.

The second simulation uses ten nodes of MGSES/GSES and ten sets of required URLs and keywords per one client for sending the required message. Figure 8 shows the comparison of the average

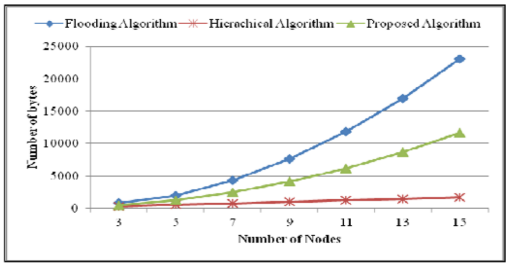

Figure 7. Comparison of the number of sent bytes at n MGSES/GSES nodes.

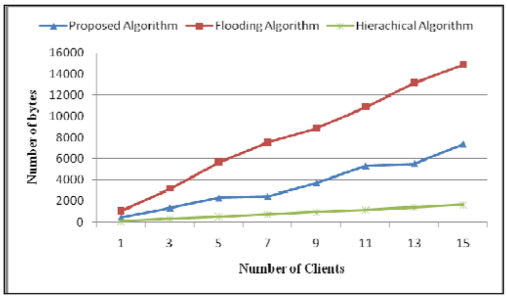

Figure 8. Comparison of the number of sent bytes of three algorithms at m client.

number of bytes among three algorithms at *m* clients. In addition, the simulation system will record all sending bytes over the network. The result shows that the number of bytes of the hierarchical algorithm is minimal and the number of bytes of the flooding algorithm is more than the number of bytes of the proposed algorithm. When the number of clients is increased, the number of bytes of the flooding algorithm is much higher than the number of bytes of the PTP algorithm. The proposed algorithm can reduce the number of bytes of the flooding algorithm by more than 50%. The number of sent bytes by the hierarchical algorithm is slightly increased when the number of clients is raised.

The response time

Another parameter of the search performance is the response time; that is the period of time from when the client sends the query message out until the client receives the result message. The result of first simulation is shown in Figure 9; the response times of the three algorithms when the number of MGSES/GSES nodes is increase. The response time of the hierarchical algorithm is higher than the response time of the flooding algorithm and the proposed algorithm. The response time of the proposed algorithm is slightly higher than response time of the flooding algorithm because the proposed algorithm checks the status of MGSES before sending query messages.

The result of the response time of three algorithms in a second simulation is shown in Figure 10; when the number of clients is increased, the result is similar the first simulation. The response time of the proposed algorithm is slightly higher than response time of the flooding algorithm because the proposed algorithm checks the status of MGSES before sending query messages. The response time of the hierarchical algorithm is higher than the response time of the flooding algorithm and the proposed algorithm.

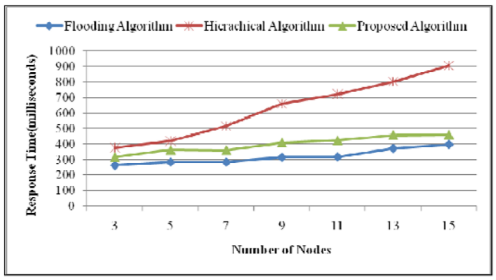

Figure 9. Comparison of the response times of three algorithms at *n* MGSES/GSES nodes.

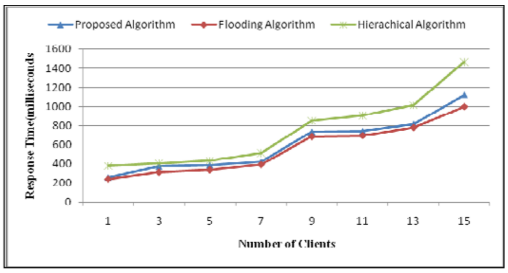

Figure 10. Comparison of the response times of three algorithms at *m* clients.

6.2 Theoretical Analysis

The number of bytes

Since the PTP algorithm is used to search the routing path of the ISP, the size of the PTP must be small. The size of the PTP is 16 bytes and the size of the ISP is a dynamic value depending on the length of keywords. In the ISP format, the length of the previous ISP is more than 89 bytes, and the length of the modified ISP is more than 91 bytes because two bytes of the Query ID and the Flag field are added. The volume of bytes of the PTP is much smaller than those of the ISP. Therefore, the PTP easily and quickly checks the ISP sending

and receiving statuses in other search engines and also reports to the current MGSES. The current MGSES will consider the suitable MGSESs to send.

The number of the sending ISPs refers to the total ISP packets that are sent to search the required URL in the network from the first search engine to others until the required URL is found.

This section considers the first simulation which is when the MGSESs/GSESs is increased. Moreover, this section analyzes the case that the required URL is not found.

The description of the variables is shown in Table 4. The calculation of the number of delivered ISPs is shown in Equation 1 and the total bytes of all sending ISPs is shown in Equation 2.

$$\sum_{i=1}^{n} link(node_i)$$

is the summation of link to neighbor of $node_i$.

$$NumberISP = \sum_{i=1}^{n} link(node_i) \quad (1)$$

$$NumberByte \geq (\sum_{i=1}^{n} link(node_i)) \times 89 \quad (2)$$

The PTP reduces the number of sending the modified ISPs, shown in Equation 3.

$$NumberISP = n - 1 \quad (3)$$

The total delivery of PTPs is the number of PTP packets that are sent to search for the routing path in the network from the second search engine to all others until the required URL is found.

The number of PTP packets can be calculated as in Equation 4. In Equation 4,

$$\sum_{i=1}^{n} link(node_i)$$

is subtracted (n-1) because the PTPs are not sent from the first MGSES.

$$NumberPTP = \sum_{i=1}^{n} link(node_i) - (n-1) \quad (4)$$

Therefore, the total bytes of all sending in the PTP and the ISP are shown in Equation 5.

$$NumberByte \geq (\sum_{i=1}^{n} link(node_i)) - (n-1) \times 16 + (n-1) \times 91 \quad (5)$$

In the hierarchical algorithm, the volume of sending query messages is the number of GSES nodes that is n -1. Thus, the overall bytes can be calculated as in Equation 6.

$$NumberByte \geq (n-1) \times 89 \quad (6)$$

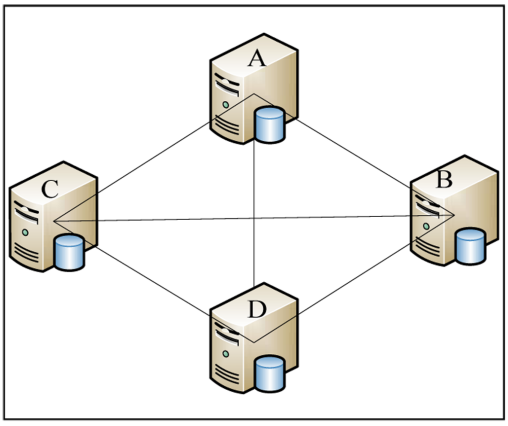

Figure 11. The mesh topology.

The simulation uses the mesh topology in the flooding algorithm and the proposed algorithm. Thus, the summation of links to the neighbor of each node is $n(n-1)$. In Figure 11, link(A) = 3, link(B) = 3, link(C) = 3, link(D) = 3. The summation of links to the neighbors of each node is 12. Therefore, total bytes of the previous algorithm is shown in Equation 7.

$$NumberByte \geq (n(n-1)) \times 89 \quad (7)$$

Total bytes of the PTP algorithm are shown in Equation 8.

$$NumberByte \geq \{(n(n\text{-}1))\text{-}(n\text{-}1)\}\times 16 + (n\text{-}1)\times 91 \quad (8)$$

The Equation 6, 7, and 8 can represent the value of n using n = 3, 5, 7, 9, 11, 13, and 15. Figure 12 shows the calculation results and the comparison of bytes under the experiment. It can be concluded that results from both methods are the same.

The response time

The time complexity of each algorithm is calculated for the response time of the search algorithm. The algorithm of the flooding algorithm, the hierarchical algorithm, and the PTP algorithm are shown in Figure 10, Figure 11, and Figure 12, respectively.

According to the flooding algorithm in Figure 10, the time complexity of this algorithm is $O(n^2)$ where n is the number of nodes. For the PTP algorithm in Figure 12, the time complexity is $O(n^2+n)$ because the PTP algorithm uses the flooding technique for status checking of all sending query messages. Based on the mesh topology, the response time of the flooding algorithm and the PTP algorithm are O(1) because the client just sends query messages to its nearest neighbors and the nearest neighbors forward to others.

As a result, the complexity of the response time is counted as a constant value. Therefore, the response time of the flooding algorithm and PTP algorithm are $O(1)$. According to the hierarchical algorithm in Figure 11, the time complexity of this algorithm is $O(\log n)$.

Thus, the response time of the flooding algorithm and PTP algorithm is better than the response time of the hierarchical algorithm.

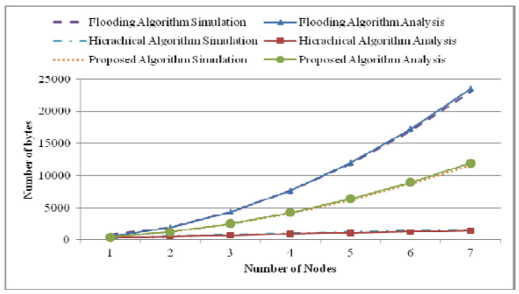

Figure 12. Comparisons of bytes among the simulation and the equations.

The avoidance congestion analysis

This topic shows that the proposed algorithm can avoid the congestion in a network channel. The details of this proof are described as follows.

Let b_t be the required bandwidth at time t for t = 1,2,3,.., Δt.

Let u_i be user i that uses the system, for i =1,2,3,..., m where m is the number of users.

Let q_{ij} be the used bandwidth of a query message j of user i at time t for j = 1,2,3,.., p where p is a number of query message of user i.

Table 4. The description of variables.

Variable	Description
NumberISP	The total number of sending ISP
NumberPTP	The total number of sending PTP
NumberByte	The total number of bytes of all sending
$node_i$	The GSESs node number i
N	The number of GSESs
$link(node_i)$	The number of link to neighbor of $node_i$

```
Flooding Algorithm
  Let v be the current node
  Let s be the source node
  IF v = s Then
      Send the query message to all neighbors of v.
  Else
      Let R be the set of all the query message received
      Let q be the query message received
      Let w be the neighbor of v that sent p
      IF q found data at v Then
        Send result to client
      Else
        IF q is in R Then
          Do Nothing, since q was already processed by v
        Else
          For each neighbor y of v such that y ≠ w do
              send q to y
          End
        End
      End
  End
```

Figure 13. The flooding algorithm.

```
Hierarchical Algorithm
  Let v be the current node
  Let q be the query message
  Hierarchical_search(v, q)
  IF q found data at v Then
    Send result to client
  Else
    For each children y of v do
      Send q to y
      Hierarchical_search(y, q)
    End
  End
```

Figure 14. The hierarchical algorithm.

Let r_{ij} be the used bandwidth of search result of the query message j of user i at time t.

Let s be a number GSESs in the mesh topology. Thus, the total required bandwidth is:

$$\sum_{t=1}^{\Delta_t} b_t = b_1 + b_2 + \ldots + b\Delta_t$$

If user (u_i) sends the query message j then the system will return the result message r_{ij} at time Δt.

There are two situations to occur when the query massages are sent.

```
PTP Algorithm
Function Sending the PTP
  Let v be the current node
  Let s be the source node
  IF v = s Then
    Send the PTP to all neighbors of v.
  Else
    Let R be the set of all the PTP received
    Let p be the PTP received
    Let w be the neighbor of v that sent p
    IF p is in R then
      Do Nothing, since p was already processed by v
    Else
      Sent the PTP for requesting the query message
      Receive the query message.
      IF p found data at v Then
        Send result to client
      Else
        For each neighbor y of v such that y ≠ w do
          send p to y
        End For
      End
    End
  End
Function Receiving the PTP for requesting the query message
Send the query message to the neighbor
that request the query message
```

Figure 15. The PTP algorithm.

Situation 1: If the nearest GSES of the client has the required URL, the search results are returned to the client. The query messages are not sent to other GSESs.

Thus, the total required bandwidth is:

$$\sum_{t=1}^{\Delta_t} b_t = \sum_{i=1}^{m} \sum_{j=1}^{p} [q_{ij} + r_{ij}].$$

Situation 2: If the nearest GSES of the client does not have the required URL, the query messages are sent to other GSESs. Then, the MGSES that has required URL will return the result message r_{ij}.

Thus, the total required bandwidth is:

$$\sum_{t=1}^{\Delta_t} b_t = \sum_{i=1}^{m} \sum_{j=1}^{p} [q_{ij} + q_{ij}(\text{s-1}) + q_{ij}(\text{s-1})(\text{s-2}) + r_{ij}] \quad (9)$$

Let $bPTP_t$ be the used bandwidth at time t when the system applies the PTP algorithm.

The congestion will not occur if:

$$\sum_{t=1}^{\Delta_t} b_t < C$$

Let $bPTP_t$ be the used bandwidth at time t when the system applies the PTP algorithm.

Let PTP_{ij} be the used bandwidth of a PTP packet j of user i.

There are also two situations that occur when the query massages are sent.

Situation 1: If the nearest MGSES of the client has the required URL, the search results are returned to the client. The query messages are not sent to other MGSESs. Thus, the total required bandwidth of PTP algorithm is the same as the previous algorithm, that is:

$$\sum_{t=1}^{\Delta_i} b_t = \sum_{i=1}^{m} \sum_{j=1}^{p} [q_{ij} + r_{ij}].$$

Situation 2: If the nearest MGSES of the client does not have the required URL, the PTP_{ij} *is* sent to other MGSESs to look for the path to send the query message. Then, the MGSES that has the required URL will return the result message r_{ij}.

Thus, the total required bandwidth of the PTP algorithm is:

$$\sum_{t=1} bPTP_t = \sum_{i=1} \sum_{j=1}^{p} [q_{ij} + q_{ij}(s\text{-}1) + PTP_{ij}(s\text{-}1)(s\text{-}2) + r_{ij}] \quad (10)$$

In order to prove that the use of PTP can avoid the congestion when compared with the use of the normal flood mechanism,

$$\sum_{t=1}^{\Delta_i} bPTP_t < \sum_{t=1}^{\Delta_i} b_i$$

must be elaborated.

Assume that :

$$\sum_{t=1}^{\Delta_i} bPTP_t < \sum_{t=1}^{\Delta_i} b_i \text{ is not true.}$$

Then,

$$\sum_{t=1}^{\Delta_i} b_i - \sum_{t=1}^{\Delta_i} bPTP_t \le 0.$$

Based on (9) and (10), then,

$$\sum_{i=1}^{m} \sum_{j=1}^{p} [q_{ij} + q_{ij}(s\text{-}1) + q_{ij}(s\text{-}1)(s\text{-}2) + r_{ij} - q_{ij} - q_{ij}$$

$$(s\text{-}1) - PTP_{ij}(s\text{-}1)(s\text{-}2) - r_{ij}] \le 0$$

$$\sum_{i=1}^{m} \sum_{j=1}^{p} [q_{ij}(s\text{-}1)(s\text{-}2) - PTP_{ij}(s\text{-}1)(s\text{-}2)] \le 0$$

$$(s\text{-}1)(s\text{-}2) \sum_{i=1}^{m} \sum_{j=1}^{p} [q_{ij} - PTP_{ij}] \le 0$$

Consider the packet sizes of both q_{ij} and PTP. The packet size of q_{ij} is more than 89 bytes but the packet size of PTP is only 16 bytes. Thus, the packet size of q_{ij} is approximately 5 times larger than the PTP packet size. Then, the equation is not true because:

$$\sum_{i=1}^{m} \sum_{j=1}^{p} q_{ij} > \sum_{i=1}^{m} \sum_{j=1}^{p} PTP_{ij}$$

Thus, the proof has shown the contradiction Therefore,

$$\sum_{t=1}^{\Delta_i} bPTP_i < \sum_{t=1}^{\Delta_i} b_i < C.$$

In conclusion, the PTP algorithm can avoid congestion in network channels.

Moreover, if the number of MGSESs/ GSESs increases, the PTP algorithm can decrease the required bandwidth when compared with the previous algorithm. Furthermore, if the number of users (u_i) increases, the PTP algorithm also can decrease the required bandwidth when compared with the previous algorithm.

7. DISCUSSION

Due to the fact that there is a large amount of shared information resources distributed over the Internet, search protocols with efficient search algorithms are significant factors for users' successful searches. Unfortunately, the available search mechanisms do not always return the desired results. Moreover, some search mechanisms may cause network congestion and high delay.

Thus, the search mechanism, called Patrol Packet Algorithm (PTP), developed for Information Search Protocol (ISP), is proposed. The evaluation results of the PTP show that the sending queries are fewer than when using the flooding mechanism, while it is higher than in the use of the hierarchical model. These are effects from the implemented network architecture since the available links are not equivalent. In the comparisons of queries that are issued to the searching nodes, the total number of bytes of the PTP is smaller than the flooding algorithm because small packets of the PTP are sent for path checking before the actual delivery of queries. Conversely, the total bytes of the PTP are larger than the hierarchical implementation. Contrary to the previous results, the response time from the proposed PTP is larger than the flooding method but it is smaller than using the hierarchy system. This is the consequence of sending mechanisms since the PTP applied the flooding with a prior path checking process.

8. CONCLUSIONS

The information available on the Internet is distributed over various web sites. Thus, the search method is important for information retrieval. However, the existing search engine techniques generally obtain large web site lists. Thus, it is time consuming for users to find the specific information that is desired. Therefore, the ISP [3] is proposed for a specific search by entering a part of the required URL and some keywords to find a full path name of the expected URL. As a consequence, the search results consist of a short list of the more specifically required URLs. However, the defect of the previous ISP mechanism is the routing system.

In order to fix this defect, the PTP is proposed and implemented in this paper to decrease the ISP duplication and rebounding to the same GSES. The simulation result of the PTP algorithm and the routing of the previous ISP show that the simulation result of the proposed routing algorithm reduces by more than 50% the amount of transferred bytes in a communication channel. Thus, the PTP algorithm can avoid the congestion caused by transferring the ISP in the system.

According to the results of the performance testing of the PTP, this algorithm is suitable for the search requirements over a distributed system where there are a large number of query messages spread over the communication channel. Moreover, the PTP algorithm can be applied to other systems that send the query messages to all nodes for finding the distributed resources, such as peer-to-peer systems and distributed databases.

ACKNOWLEDGEMENTS

This research is financially supported by the 90th Anniversary of Chulalongkorn University Fund (Ratchadapisek Sompote Endowment Fund) and the University Development Committee (UDC) Scholarship Program of the OHEC. We would like to thank Mr. Tony Norris Criswell (English Department, Faculty of Humanities and Social Sciences, Khon Kaen University) for correcting the English.

REFERENCES

[1] Doshi P. and Raisinghani V., Review of Dynamic Query Optimization Strategies in Distributed Database, *Proceedings of the 3rd International Conference on Electronics Computer Technology (ICECT),* 2011; **6**: 145-149.

[2] Tsoumakos D. and Roussopoulos N., Analysis and Comparison of P2P Search Methods, *Proceedings of the 1st International Conference, Scalable Information Systems*, 2006.

[3] Bhattarakosol P. and Preechaveerakul L., Information Searching Protocol: A Smart Protocol for Specific Content Search over Internet, *Proceedings of SPIE's International Symposium on Optics East 2006 (IT405)*, 2006.

[4] Winter J., Routing of structured queries in large-scale distributed systems, *LSDS-IR* 2008; 11-18.

[5] Lv Q., Cao P., Cohen E., Li K. and Shenker S., Search and Replication in Unstructured Peer-to-Peer Networks, *Proceedings of 16th International Conference on Supercomputing,* 2002; 84-95.

[6] Tsoumakos D. and Roussopoulos N., Adaptive Probabilistic Search for Peer-to-Peer Networks, *Proceedings of the 3rd International Conference on P2P Computing,* 2003; 102-109.

[7] Stoica I., Morris R., Karger D., Kaashoek F. and Balakrishnan H., Chord: A Scalable Peer-to-Peer Lookup Service for Internet Applications, *ACM SIGCOMM Conference*, 2001.

[8] Kalogeraki V., Gunopulos D. and Zeinalipour-Yazti D., A Local Search mechanism for Peer-to-Peer Networks, *Proceedings of the 11th International Conference on Information and Knowledge Management*, 2002; 300 - 307.

[9] Rozlina M. and Buckingham C.D., Preprocessing for Improved Query Routing in Super-peer P2P Systems, *IEEE Region 5 Technical, Professional, and Student Conference*, 2008.

[10] Yang B. and Garcia-Molina H., Improving Search in Peer-to-Peer Systems, *Proceedings of the 22nd IEEE International Conference on Distributed Computing (IEEE ICDCS'02),* 2002.

[11] Dorrigiv R., L′opez-Ortiz Al. and Prałat P., Search Algorithms for Unstructured Peer-to-Peer Networks, *Proceedings of 32nd IEEE Conference on Local Computer Networks (LCN'07),* 2007; 343-352.

[12] Jawhar I. and Wu J., A Two-level Random Walk Search Protocol for Peer-to-Peer Networks, *Proceedings of the 8th World Multi-Conference on Systemics, Cybernetics and Informatics*, 2004.

[13] Mockapetris P.V., Domain names-concepts and facilities, RFC 1034, November, 1987.

[14] Mockapetris P.V., Domain names-implementation and specification, RFC 1035, November, 1987.

[15] Mockapetris P.V., Development of the Domain Name System; *Proceedings of ACM SUGCOMM'88*, New York USA, 1988; 123-133.

[16] Watttana M., Bhattarakosol P., Patrol Packet (PTP) for Routing Algorithm of Information Searching Protocol (ISP), *Proceedings of the 1st International Conference on Future Networks,* 2009; 244-248.