

Highly Available Hypercube Tokenized Sequential Matrix Partitioned Data Sharing in Large P2P Networks

C. G. Ravichandran^{1*}, J. Lourdu Xavier²

¹SCAD Institute of Technology, Palladam, India ²Faculty of Computer Science and Engineering, RVS College of Engineering & Technology, Dindigul, India Email: ^{*}jlourduxaviersjc@gmail.com

Received 6 April 2016; accepted 1 May 2016; published 8 July 2016

Copyright © 2016 by authors and Scientific Research Publishing Inc. This work is licensed under the Creative Commons Attribution International License (CC BY). http://creativecommons.org/licenses/by/4.0/

CC ① Open Access

Abstract

Peer-to-peer (P2P) networking is a distributed architecture that partitions tasks or data between peer nodes. In this paper, an efficient Hypercube Sequential Matrix Partition (HS-MP) for efficient data sharing in P2P Networks using tokenizer method is proposed to resolve the problems of the larger P2P networks. The availability of data is first measured by the tokenizer using Dynamic Hypercube Organization. By applying Dynamic Hypercube Organization, that efficiently coordinates and assists the peers in P2P network ensuring data availability at many locations. Each data in peer is then assigned with valid ID by the tokenizer using Sequential Self-Organizing (SSO) ID generation model. This ensures data sharing with other nodes in large P2P network at minimum time interval which is obtained through proximity of data availability. To validate the framework HS-MP, the performance is evaluated using traffic traces collected from data sharing applications. Simulations conducting using Network simulator-2 show that the proposed framework outperforms the conventional streaming models. The performance of the proposed system is analyzed using energy consumption, average latency and average data availability rate with respect to the number of peer nodes, data size, amount of data shared and execution time. The proposed method reduces the energy consumption 43.35% to transpose traffic, 35.29% to bitrev traffic and 25% to bitcomp traffic patterns.

Keywords

Peer-to-Peer (P2P), Video-on-Demand, Hypercube, Sequential Matrix Partition, Data Mapping, Data Availability

^{*}Corresponding author.

How to cite this paper: Ravichandran, C.G. and Lourdu Xavier, J. (2016) Highly Available Hypercube Tokenized Sequential Matrix Partitioned Data Sharing in Large P2P Networks. *Circuits and Systems*, **7**, 2109-2119. http://dx.doi.org/10.4236/cs.2016.79183

1. Introduction

In today's highly interconnected work, data availability and data sharing play a significant role in P2P networks. In the context of network management and traffic engineering Traffic Matrices in Peer Network (TM-PN) [1] was designed for providing traffic optimization. Typically, providing tradeoff between peer populations was the objective of Chunk Scheduling in Peer Network (CS-PN) performed through mixed strategy. However, the two methods suffer from data sharing in peer network. Peer-to-Peer (P2P) systems are distributed systems without (or with a minimal) centralized control or hierarchical organization, where each node is equivalent in term of functionality. P2P refers to a class of systems and applications that employ distributed resources to perform a critical function such as resources localization in a decentralized manner. The main challenge in P2P computing is to design and implement a robust distributed system composed of distributed and heterogeneous peer nodes, located in unrelated administrative domains. In a typical P2P system, the participants can be "domestic" or "enterprise" terminals connected to the Internet. The applications of the P2P systems in current days are PPLive and UUSEE data streaming methodologies. In live telecasting of P2P systems, lacks of multimedia data such as videos and audios are shared by millions of users at a time in a common system using P2P application protocol. The P2P system is highly responsible for delivering the data in secured manner. High traffic affects the performance of the P2P system as a serious problem in large networks. In a peer-to-peer network as shown in Figure 1, a group of computers are connected together so that users can share resources and information. There is no central location for authenticating users, storing files, or accessing resources. This means that users must remember which computers in the workgroup have the shared resource or information that they want to access. It also means that users must log on to each computer to access the shared resources on that computer. In most peerto-peer networks, it is difficult for users to track where information is located because data is generally stored on multiple computers. This makes it difficult to back up critical business information, and it often results in small businesses not completing backups. Often, there are multiple versions of the same file on different computers in the workgroup.

The main contribution of this paper is to propose an efficient Hypercube Sequential Matrix Partition (HS-MP) for efficient data sharing in P2P Networks using tokenizer method which is further to resolve the problems of the larger P2P networks. The remainder of the paper is organized as follows. Section 2 provides an overview on the related work. Section 3 describes the design of Hypercube Tokenized Sequential Matrix Partitioned (HS-MP) Data Sharing technique. The performance evaluation and its discussion with conventional techniques are presented in Section 4. Section 5 depicts the conclusion.

2. Literature Survey

Ozkasapa *et al.* [2] have proposed an energy cost model for P2P network with the objective of improving scalability through distributed greedy algorithm. However, accuracy remained unsolved. In this work, accuracy in terms of data sharing rate was performed using Matrix Partition technique. As an extension, a gossip-based protocol was designed by Emrah Çema *et al.* [3] to addressed accuracy through aggregate function. The authors proposed and developed a fully distributed Protocol for Frequent Items Discovery (ProFID) where the result is



produced at every peer. ProFID uses gossip-based (epidemic) communication, a novel pair-wise averaging function and system size estimation together to discover frequent items in an unstructured P2P network.

Ozkasapa *et al.* [4] constructed a self-organization peer network for distributed systems. The authors derived exact expressions for infection probabilities through elaborated counting techniques on a digraph. Considering the first passage times of a Markov chain based on these probabilities, the expected message delay was experienced by each peer and its overall mean as a function of initial number of infectious peers. An analytical framework for self-organizing nodes in P2P network was introduced in this paper. Another method called fair-share buffering was introduced by Ozkasapa *et al.* [5] aiming at improving the reliability of dissemination. The authors proposed a novel approach; stepwise fair-share buffering that provides uniform load distribution and reduces the overall buffer usage where every peer has a partial view of the system. They reported and discussed the comparative performance results with existing buffering approaches as well as random buffering which serves as a benchmark.

Mohammad Shojafar *et al.* [6] proposed Improved Adaptive Probabilistic Search (IAPS) algorithm to improve the response rate through optimization. IAPS used ant-colony optimization and observed file types into consideration in order to search for file container nodes with a high probability of success. The authors performed extensive simulations to study the performance of IAPS, and compared it with the Random Walk and Adaptive Probabilistic Search algorithms. Gedik *et al.* [7] applied Data stream processing for data partitioning through control algorithm to measure the cost. However all the methods mentioned above lack data sharing which is addressed in this paper through a design of Time-based Matrix Partition that maps the peer location with ID for efficient data sharing. Richard Lin *et al.* [8] proposed Flexible data sharing algorithm for cloud-enabled extended peer system using Extended Bestpeer. With the objective of ensuring data availability and accessibility of data in bio-informatics, a Distributed Hash Table with Round Trip Time (RTT) was introduced ensuring scalability.

The conventional methodologies for P2P network consumed more energy for transferring the packets between the peer nodes in the network and also the average packet latency was high due to the instability of the architecture and complex algorithm construction. In this work, we propose Hypercube Sequential Matrix Partition (HS-MP) for efficient data sharing in large P2P network based on tokenizer, which outperforms P2P streaming model. In particular, we make the following contributions in this paper:

- To provide a modeling of Dynamic Hypercube Organization and give a formula to derive the link between the peer and data at various locations in large P2P network ensuring data availability.
- To extend our framework to Sequential Self-Organizing (SSO) ID generation model and formalize data sharing based on data proximity aiming at reducing the time to obtain the data.
- To develop Time-based Matrix Partition for mapping data with the ID based on location in a horizontal and vertical manner and can measured at different time intervals.

The conventional methods [2]-[8] for P2P networks consumed more energy consumption and latency which reduces the performance of the P2P networks. It did not support larger P2P system in efficient manner. Hence, Hypercube Tokenized Sequential Matrix Partitioned Data Sharing technique is proposed in this paper for large scale P2P Network.

3. Proposed System

To improve the data sharing in large P2P networks, the design of Hypercube Tokenized Sequential Matrix Partitioned (HS-MP) Data Sharing technique is proposed and the flow is illustrated in **Figure 2**.

Figure 2 shows the block diagram of HS-MP. In order to achieve efficient data sharing in P2P network, the architecture of HS-MP is divided into three steps. They are given as:

- Construction of Dynamic Hypercube Organization,
- Design of Sequential Self-Organizing ID generation model and
- An efficient Time-based Matrix Partition technique.

The three-step framework enhances the execution performance of the system with the help of tokenizer. The tokenizer initially identifies the availability of data across the network. Once the data availability in peer network is identified, they are assigned with ID. Finally, the assigned data is mapped with peer location to ensure data sharing in large P2P network. The elaborate description about HS-MP framework is described in the forth-coming sections.

C. G. Ravichandran, J. Lourdu Xavier



3.1. Construction of Dynamic Hypercube Organization

The first step in the design of the proposed framework is the identification of data availability using Dynamic Hypercube Organization (DHO) by the tokenizer. The tokeziner (*i.e.*, a peer node also a server) assists the peer nodes in P2P network for identifying the data availability using DHO.

In DHO, let us consider a network graph "G = (V, E)", where "G" represents the graph with "V" vertices and "E" edges. In peer-to-peer network, "V" denotes the set of peer nodes represented by " $P_i = P_1, P_2, \dots, P_n$ " and "E" denotes the links represented by " $l_i = l_1, l_2, \dots, l_n$ " and is mathematically formulated as given below.

$$V \to P_i = P_1, P_2, \cdots, P_n \tag{1}$$

$$E \to l_i = l_1, l_2, \cdots, l_n \tag{2}$$

The tokenizer in the proposed framework is a server that efficiently coordinates and assists the peers in P2P network. It also maintains the list of peers that are currently in the network. The statistics regarding the nodes in the peers with their types of data is also maintained and is formulated as given below.

$$n_i \rightarrow Data_i$$
 (3)

From Equation (3), node " n_i " has the data "*Data*_j". In the proposed framework, the set of nodes/edges are arranged in dynamic structure with the objective of entering or leaving of the peer node in the P2P network at any time. In the proposed framework, all peers in P2P network are arranged using a Dynamic Hypercube Organization, where each peer comprises of many nodes, ensuring data availability at many locations. **Figure 3** shows the construction of DHO where each node consists of many peers.

As shown in the **Figure 3**, five types of data " $Data_1$, $Data_2$, $Data_3$, $Data_4$, $Data_5$ "are available in various peer nodes " P_1 , P_2 , P_3 , P_4 , P_5 , P_6 , P_7 ", in P2P network at an instant of time. In each peer, there exists either two or three nodes namely, " n_1 , n_2 , n_3 " and totally twenty three nodes exist with each node possessing different types of data. From **Figure 3**, the peer node " P_1 " has the data " $Data_1$, $Data_2$, $Data_4$ " and therefore, the following links are inferred from **Figure 3**.

$$P_1 \rightarrow Data_1, Data_2, Data_4$$
 (4)

In similar manner, every peer node has different data availability at any instant of time. With the construction of DHO, though data redundancy is seen (*i.e.*, $Data_2$ in P_4) the availability of data is made through several peer nodes as the peers may leave the P2P network at any time. In order to ensure data availability, the proposed framework replicates the data over multiple peers in P2P network with the objective increasing the data availability rate even when a peer node fails.



3.2. Design of Sequential Self-Organizing ID Generation Model

The second step in the design of the proposed framework is the construction of Sequential Self-Organizing ID generation. Once the availability of data is identified by the tokenizer, the data has to be assigned with a valid ID. The proposed framework uses a Sequential Self-Organizing (SSO) ID generation model for effective assignment of ID to the corresponding data. When a data has to be shared within the peer network, using SSO, the data is assigned in an efficient manner in such a way that the data ID closest to the ID of the requester is considered.

Using SSO in the proposed framework, proximity of data availability is also considered into account aiming at minimizing the time to obtain the data. The SSO ID generation model uses a large ID space " α_{space} ". Here, the tokenizer assigns Sequential IDs to the data whenever a peer joins the network where the IDs are administered over the entire large ID space and is mathematically provided as below:

$$\alpha_{\text{space}} = 2^n \tag{5}$$

From Equation (5) the entire ID space " α_{space} " holds up to an average of about " 2^n " nodes. The Sequential Self-Organizing ID generation is then formulated as given below:

$$P_1Data_1 \to P_1D_1ID_1, P_2Data_2 \to P_2D_2ID_2, \cdots, P_nData_n \to P_nD_nID_n$$
(6)

From Equation (6), " $Data_1$ " is assigned with the ID " $P_1D_1ID_1$ ", " $Data_2$ " is assigned with the ID " $P_2D_2ID_2$ " and so on in a Sequential manner according to the data availability and the time of data entering in to the peer network. The arrival of data in the proposed framework can be of any order. But using SSO ID generation model, the ID is assigned in a sequential manner. The resultant ID generation in the proposed framework is stored in the routing table as shown in Figure 4.

The routing table in **Figure 4** includes the information regarding the peer, the data available in the peer network and their corresponding ID which is stored in sequential manner using Self-Organizing manner.

From the routing table stated in **Figure 4**, the data available " $Data_1$, $Data_2$, $Data_4$ " in peer " P_1 " is assigned with ID as " ID_1 , ID_2 , ID_3 ". Next, the data available in, " $Data_1$, $Data_1$, $Data_1$ " in peer " P_2 " is assigned with ID as " ID_4 , ID_5 , ID_6 ". In a similar manner, every data in the peer network is assigned with the Sequential ID in an efficient manner, minimizing the execution time for measuring data availability. Finally, the last step in the design of the proposed framework is the mapping of data using Time-based Matrix Partition technique. This is called as Time-based because the Matrix Partition (*i.e.*, the mapping of location with ID) changes with respect to time and does not remain the same at all intervals of time. By applying Time-based Matrix Partition technique, the data assigned with ID is mapped with peer location for efficient data sharing in peer networks.

The time-based matrix partition technique in the proposed framework uses a hash method which partition and assigns the ID with a location in a horizontal and vertical manner. The time-based matrix partition technique applied dividing element that partitions the data from the peer nodes and assigns ID to each data.

The Time-based Matrix Partition technique uses two dividing elements, one for horizontal partition and the

C. G. Ravichandran, J. Lourdu Xavier

Peer	Data	ID
\mathbf{P}_1	Data ₁ , Data ₂ , Data ₄	$P_1D_1ID_1ID_2ID_2D_4ID_3$
\mathbf{P}_2	Data ₁ , Data ₁ , Data ₁	$P_2D_1ID_4ID_1ID_5D_1ID_6$
\mathbf{P}_3	Data ₁ , Data ₄ , Data ₄	$P_3D_1ID_7ID_4ID_8D_4ID_5$
\mathbf{P}_4	Data ₁ , Data ₂ , Data ₂	$P_4 D_1 I D_1 0 D_2 I D_{11} D_2 I D_{12} \\$
P ₅	$Data_1$, $Data_5$	$P_5D_1ID_{13}D_5ID_{14}$
\mathbf{P}_{6}	Data ₁ , Data ₁ , Data ₃	$P_6D_1ID_{15}D_1ID_{16}D_3ID_{17}\\$
\mathbf{P}_7	Data ₄ , Data ₄ , Data ₄	$P_{7}D_{4}ID_{18}ID_{4}ID_{1}9D_{4}ID_{2}0$
\mathbf{P}_8	Data ₁ , Data ₁ , Data ₅	$P_8 D_1 I D_{21} I D_1 I D_{22} D_5 I D_{23} \\$

Figure 4. Routing table for peers in P2P network.

second for vertical partition. The horizontal partition in the proposed framework represents the peer with data available (*i.e.*, P_1D_1) whereas the vertical partition represents the ID of the corresponding data (*i.e.*, " P_1D_1 " " ID_1 "). As a result, the mathematical formulation for data mapping with the objective of efficient data sharing in peer network is given as below,

$$P_i D_i, I D_i \to \left(R_i, C_j\right) \tag{7}$$

From Equation (7), " R_i " represents the row and " C_j " column representation. Figure 5 shows the Timebased Matrix Partition technique that maps the location for two peers " P_1P_2 " with IDs " $ID_1ID_2ID_3ID_4ID_5ID_6$ " respectively.

Figure 5 shows the mapping of data ID with the peer location for efficient data sharing in peer network. **Figure 6** shows the HS-MP algorithm for efficient data sharing in large P2P networks. The entire algorithm is divided into three parts. The first part concentrates on measuring the overall data availability in P2P network for each vertices and edges. The peer nodes are represented by vertices whereas the link is denotes by edges with the objective of identifying the available data in the entire network.

The second part focuses on the measuring of ID for each data according to the nodes in peer network and assigning the ID's correspondingly. Finally, data mapping with peer location is measured using the routing table that efficiently maintains the location information for each data in the peer using Time-based Matrix Partition, aiming at efficient data sharing. The flowchart of the proposed HS-MP algorithm is illustrated in Figure 6.

4. Results and Discussion

4.1. Experimental Setup

The proposed methodology HS-MP for data sharing in P2P Networks using tokenizer technique is simulated using Network Simulator 2 tool. The initial network setup parameters are illustrated in **Table 1**. In order to investigate the performance of the proposed HS-MP, the experiments are conducted in the real-world environment. The number of nodes and channels are 1000 and 100, respectively. Initial data rate is set to 1Mbps and two ray ground propagation models are used to deliver the packets from one node to another node in the network. Destination Sequence Based Distance Vector (DSDV) is used as the default routing protocol in this paper.

The peer nodes in network is moving and the velocity of the moving peer node is determined using the following equation,

$$V_{peer-node} = W * V_{t-1} + (P_{t-1} - P_t) + W * (P_{t-1} * P_t)$$
(8)

where, W denotes bandwidth of the peer node, V_{t-1} is the previous velocity of the peer node, P_{t-1} is previous location of the peer node and P_t is the current location of the peer node.

The proposed Peer-to-peer (P2P) distributed architecture is compared with the conventional techniques as Peer-to-Peer Live Streaming System [7], Traffic Matrices in Peer Network (TM-PN) and Chunk Scheduling in Peer Network (CS-PN) using the following performance evaluation parameters.

C. G. Ravichandran, J. Lourdu Xavier

Figure 5. Design of data ID mapping with peer location.



Figure 6. Pseudo code of the proposed HS-MP algorithm.

Table 1. Initial network parameters.

Parameter	Assigned value
Number of nodes	1000
Number of channels	100
Sensing region	1000 * 1000 m
Initial node energy	1000 J
Packet size	512 B
Transmission power	0.02 Watts
Received power	0.01 Watts
Routing protocol	DSDV
Data rate	1 Mbps
radio-propagation model	Two Ray Ground
Mobility speed	3 m/s
Pause time	10 s

4.2. Energy Consumption

The transmission of packet between any two peer nodes in the network, consumes certain power. The determination of energy and power consumption is important to analyze the performance of the proposed HS-MP methodology. The energy consumption for transmitting a packet can be illustrated as,

$$E_{t} = \left(E_{id} \times k\right) + \left(\gamma_{amp} \times d^{\alpha}\right) \tag{9}$$

where, k refers to the number of bits transmitted;

 α is a variable factor ranging from 1 to 5 between two peer nodes based on their channel condition;

 γ_{amp} is the amplification coefficient relying on minimum bit error rate (BER);

d is the distance between any two peer nodes;

 E_{td} refers to the energy dissipation, expressed as

$$E_{td} = \frac{V_{cc}I_c}{b_{rata}} \tag{10}$$

where, V_{cc} and I_c denotes the voltage and current consumption of the peer node, respectively; and b_{rate} is the baud rate or data transmission rate.

The consumed energy is directly proportional to the number of bits transmitted and distance between two consecutive peer nodes.

The energy required to receive a packet can be expressed as,

$$E_r = E_{rd} \times k \tag{11}$$

$$E_{rd} = \frac{V_{cc}I_c}{b_{rata}}$$
(12)

The power consumption for transmitting a packet can be expressed as,

$$P_t = \left(\gamma_{amp} b_{rate}\right) \hat{d}^{\alpha} \tag{13}$$

Similarly, the power consumption required for packet bit reception can be represented by,

$$P_r = \frac{P_t}{d^{\alpha}} = \left(\gamma_{amp} b_{rate}\right) \cdot \left(\frac{\hat{d}^{\alpha}}{d^{\alpha}}\right)$$
(14)

$$P_r = \gamma_{amp} b_{rate} \left(\frac{\hat{d}}{d}\right)^{\alpha}$$
(15)

$$P_r = P_x \cdot (q)^{\alpha} \tag{16}$$

where, $P_x = \gamma_{amp} b_{rate}$ and $q = \frac{\hat{d}}{d}$.

 \hat{d} and d represents the estimated distance and actual distance between the two peer nodes, respectively.

The parameter makes an impact on the energy transmission between the two peer nodes, and the part of the transmission energy will be lost if $\hat{d} > d$.

For analysis the performance of the proposed HS-MP for P2P Networks in synthetic traffic, the experiments are done over the transpose traffic, bitrev traffic, bitcomp traffic and shuffle traffic. In transpose traffic mode, each peer node sends messages only to a destination with the upper and lower halves of its own address transposed. Each peer node sends only to whose address is bit reversal of the sender's address in bitrev traffic mode. Each peer node sends messages only to a with one's complement of its own address under bitcomp traffic mode and each peer node sends messages to other peer nodes with an equal probability in shuffle traffic mode. Figure 7 shows the Normalized energy consumption of real time traffic.

The proposed method reduces the energy consumption 43.35% to transpose traffic, 35.29% to bitrev traffic and 25% to bitcomp traffic patterns for CS-PN method. The proposed method also reduces the energy consumption 33.75% to transpose traffic, 24.65% to bitrev traffic and 12.3% to bitcomp traffic patterns for TM-PN method.



4.3. Data Availability (DA)

It is used to ensure the proposed framework with which the data is available at a required level of performance. The data availability rate is the product of number of peer nodes in large P2P network and the overall data size. The mathematical formulation for data availability rate is given as below:

$$DA_{rate} =$$
No. of peer nodes * $Data_{size}$ (17)

where, " DA_{rate} " is the data availability rate with size of data and " $Data_{size}$ " is the different number of peer nodes. It is expressed in terms of Giga Bytes (GB). The performance of the proposed system is efficient if the data availability rate is high.

The average data availability rates are 714 GB for HS-MP method, 705 GB for TM-PN method and 581 GB for CS-PN method. The proposed method increases the data availability rate to 1.26% for TM-PN method and increases the data availability rate to 18.62% for CS-PN method. **Figure 8** shows the performance analysis of data availability.

4.4. Average Packet Latency

The time taken to execute or measures the data availability for the whole P2P network is called as Average packet latency. It is measured in terms of milliseconds. The performance of the proposed system is efficient if the average packet latency is high.

$$ET = Time\left(P_i\left(Data_i\right)\right) + \left(P_{i+1}\left(Data_j\right)\right) + \dots + \left(P_{i+n}\left(Data_n\right)\right)$$
(18)

where, i = 1 to *n*; *n* represents the number of peer nodes in P2P network.

The execution time for peer node " P_1 " using the three methods HS-MP, TM-PN and CS-PN are measured and illustrated in Figure 9.

The proposed method consumes Power consumption of 0.51 mW, 0.55 mW and 0.57 mW for transpose traffic, bitrev traffic and bitcomp traffic, respectively. The average packet latency for transpose traffic, bitrev traffic and bitcomp traffic are 16.5 ms, 19.9 ms and 21.5 ms, respectively.

4.5. Precision

The data sharing rate is measured in terms of precision. It is defined as the ratio of relevant data retrieved to the irrelevant data retrieved.

$$P_{ds} = \frac{Data_r}{Data_r + Data_{ir}} *100\%$$
⁽¹⁹⁾

where, "*Data_r*" is data relevancy and "*Data_{ir}*" is data irrelevancy, the measure of precision for data sharing is obtaining using " P_{ds} ". The data sharing between peer nodes is high if precision is high. The value of precision for the proposed system and conventional systems are given in Figure 10.



Figure 8. Performance analysis of data availability.







5. Conclusion

In this paper, the problem of efficient data sharing in large P2P networks by mapping peer location with ID is solved effectively by proposed method. Three Figures of merit have been considered for system design, namely, the data availability rate, time for measuring data availability and data sharing rate for large P2P networks. Performances results reveal that the proposed HS-MP framework provides higher level of precision and data sharing rate and also increase the robustness by consuming less processing time on obtaining the data. Compared to the existing P2P data streaming methods, the proposed Hypercube Sequential Matrix Partition (HS-MP) framework is comparatively superior to the state-of-art works. The proposed method consumes energy consumption of 0.51 mW, 0.55 mW and 0.57 mW for transpose traffic, bitrev traffic and bitcomp traffic, respectively. The average packet latency for transpose traffic, bitrev traffic and bitcomp traffic, respectively. In systems.

Acknowledgements

The authors would like to thank their friends and colleagues for their constant help and support throughout the study and to obtain the results.

References

- Xu, K., Shen, M., Cui, Y., Ye, M.J. and Zhong, Y.F. (2014) A Model Approach to the Estimation of Peer-to-Peer Traffic Matrices. *IEEE Transactions on Parallel and Distributed Systems*, 25, 1101-1111. http://dx.doi.org/10.1109/TPDS.2013.179
- [2] Özkasapa, O., Çaglar, M., Ule Yaz, E. and Küçükçifçi, S. (2010) An Analytical Framework for Self-Organizing Peer-to-Peer Anti-Entropy Algorithms. *Performance Evaluation*, 67, 141-159. http://dx.doi.org/10.1016/j.peva.2009.09.009
- [3] Çema, E. and Özkasapb, Ö. (2012) ProFID: Practical Frequent Items Discovery in Peer-to-Peer Networks. *Future Generation Computer Systems*, **6**, 1544–1560.
- [4] Ozkasap, O., Caglar, M., Cema, E., Ahi, E. and Iskender, E. (2010b) Stepwise Fair-Share Buffering for Gossip-Based Peer-to-Peer Data Dissemination. *Computer Networks*, 53, 2259-2274. <u>http://dx.doi.org/10.1016/j.comnet.2009.03.021</u>
- [5] Ozkasap, O., Cem, E, Cebeci, S.E. and Koc, T. (2013) Flat and Hierarchical Epidemics in P2P Systems: Energy Cost Models and Analysis. *Future Generation Computer Systems*, **36**, 257-266. http://dx.doi.org/10.1016/j.future.2013.09.009
- [6] Shojafar, M., Abawajy, J.H., Delkhah, Z., Ahmadi, A., Pooranian, Z. and Abraham, A. (2013) An Efficient and Distributed File Search in Unstructured Peer-to-Peer Networks. *Peer to Peer Network Application*, 8, 120-136. http://dx.doi.org/10.1007/s12083-013-0236-0
- [7] Gedik, B., Schneider, S., Hirzel, M. and Wu, K.L. (2014) Elastic Scaling for Data Stream Processing. IEEE Transactions on Parallel and Distributed Systems, 25, 1-10. <u>http://dx.doi.org/10.1109/TPDS.2013.295</u>
- [8] Richard Lin, C.H., Wen, C.H., Lin, Y.C., Tung, K.Y., Lin, R.W. and Lin, C.Y. (2013) A P2P Framework for Developing Bioinformatics Applications in Dynamic Cloud Environments. *International Journal of Genomics*, 2013, 1-9. http://dx.doi.org/10.1155/2013/361327



Submit or recommend next manuscript to SCIRP and we will provide best service for you:

Accepting pre-submission inquiries through Email, Facebook, Linkedin, Twitter, etc

A wide selection of journals (inclusive of 9 subjects, more than 200 journals)

Providing a 24-hour high-quality service

User-friendly online submission system

Fair and swift peer-review system

Efficient typesetting and proofreading procedure

Display of the result of downloads and visits, as well as the number of cited articles

Maximum dissemination of your research work

Submit your manuscript at: http://papersubmission.scirp.org/