PROCEEDING

IEEE COMNETSAT 2012

2012 IEEE International Conference on Communication, Networks and Satellite

ISBN : 978-1-4673-0889-2

BALI, INDONESIA, 12-14 JULY 2012

IEEE
INDONESIA SECTION

2012 IEEE International Conference on Communication, Networks and Satellite

(ComNetSat 2012)

Bali, Indonesia
12 – 14 July 2012

IEEE

IEEE Catalog Number: CFP1231S-PRT
ISBN: 978-1-4673-0888-5

TECHNICAL PROGRAM

July 12, 2012

KEYNOTE SPEECH
Convergence of Communications towards Smart Era
Prof. Byeong Gi Lee

July 14, 2012

PLENARY SPEECH
Cognitive Radio Networks: Design Issues and Security Threats
Prof. Pramod K. Varshney

Room A

July 13, 2012
09.00 - 10.30

CT1.1: Internet Computing

<u>CT1.1.1</u> **Auction-based vs. Incentive-based Multiple-Cloud Orchestration Mechanisms**
Ganesh Neelakanta Iyer, Ramkumar Chandrasekaran, Bharadwaj Veeravalli
National University of Singapore

<u>CT1.1.2</u> **Efficient Normal Peers Group Recovery in Hierarchical Peer-To-Peer**
Sri Wahjuni, Kalamullah Ramli, Anak Agung Putri Ratna
University of Indonesia

Room B

July 13, 2012
10:45 - 12:00

CT2.1: Mobile & Wireless Broadband

CT2.1.1 Sum Rate Maximization for Spectrum Sharing Multiuser MIMO Network under Rayleigh Fading
Jie Tang, Georgia Bournaka, Sangarapillai Lambotharan
Loughborough University

CT2.1.2 Mitigating Sybil Attacks in Vanets
Khaled Mohamed Rabieh, Marianne Azer, Mahmoud Allam
Nile University

CT2.1.3 Blind Spectrum Sensing for Cognitive Radio
Kris Sujatmoko, Gunawan Wibisono, Dadang Gunawan
University of Indonesia

CT2.1.4 A Taxonomy for Congestion Control Algorithms in Vehicular Ad Hoc Networks
Mohammad Reza Jabbarpour Sattari, Rafidah Md Noor, Hassan Keshavarz
University of Malaya

CT2.1.5 Adaptive Beamforming Schemes for SC-FDMA Systems with Insufficient Cyclic Prefix
Shiuan-Wei Huang, Kuo-Ching Fu, Yung-Fang Chen
Department of Communication Engineering, National Central University

CT2.1.6 The Development of 3D Curve Dynamic Path Planning Simulation in Cluttered Environment
T B Adji, Hendri H Triharminto, N A Setiawan, A S Prabuwono
Gadjah Mada University

Room A

July 13, 2012
10:45 - 12:00

CT3.1: Quality, Security, and Management

CT3.1.1 End User Based Measurement System For Cellular Packet Data Network Performance
Zahir Zainuddin, Ady Wahyudi Paundu
Hasanuddin University

CT3.1.2 Offline Delegation Protocol for Mobile RFID
Jia-Ning Luo, Ming-Hour Yang, Ming-Chien Yang, Ming-Chi Tseng
Ming Chuan University

CT3.1.3 A Searchable Encryption Scheme for Outsourcing Cloud Storage
Jyun-Yao Huang, I-En Liao
Department of Computer Science and Engineering at National Chung Hsing University

Room B

July 13, 2012
13:30 - 15:00

CT2.2: Mobile & Wireless Broadband

CT2.2.1 **Performance Analysis of Open and Closed Loop Spatial Multiplexing in LTE Downlink Physical Layer**
Pooja S Suratia, Satish K Shah
Department of Electrical Engineering, The Maharaja Sayajirao University of Baroda

CT2.2.2 **Capacity Analysis of Device-to-Device Resource Reusing Modes for Cellular Networks**
Rui Chen, Xuewen Liao, Shihua Zhu, Zhonghua Liang
Xi'an Jiaotong University

CT2.2.3 **Mobile Geotagged Data Gathering for Disaster Remediation**
Sally Elizabeth Goldin, Kurt T Rudahl
King Mongkut's University of Technology Thonburi

CT2.2.4 **Time-Oriented Care-of Address for Mobile IPv6 Networks**
Yu-Hsin Cheng, Shang-Juh Kao, Fu-Min Chang
Department of Information Networking and System Administration, Ling Tung University

CT2.2.5 **An Efficient FemtoCell Deployment Scheme for Mitigating Interference in Two-tier Networks**
Hsiu-Lang Wang, Shang-Juh Kao
Department of Computer Science and Engineering, National Chung-Hsing University

Room A

July 13, 2012
13:30 - 15:00

CT4.1: Quality, Security, and Management

CT4.1.1 **Detecting Policy Misconfigurations in Temporal Domain Using Object Priority**
Madhu Sankeerth Dammati, Samrat Mondal
Indian Institute of Technology Patna

CT4.1.2 Performance Evaluation of Available Bandwidth Estimation Tools in FTTH Networks
Mun Leong Chan, Su Wei Tan, Ahmad Tajuddin Samsudin, Ahmad Fuad Mohamed Bandi
Multimedia University

CT4.1.3 Analysis Against Secret Redundancy Mechanism for RFID Authentication Protocol
N.W. Lo, Kuo-Hui Yeh, Hsuan-Yu Chen
Department of Information Management, National Taiwan University of Science and Technology

Room B

July 13, 2012
15:30 - 17:00

CT1.2: Architecture & Protocols

CT1.2.1 FPGA Implementation of ANN for Reactive Routing Protocols in MANET
Satish K Shah, Dharmistha Doodhnath Vishwakarma
M.S.University of Baroda

CT1.2.2 Credit-based Low Latency Packet Scheduling Algorithm for Real-time Applications
Lyu-Han Chen, Hsiao-Kuang Wu, Ming-I Hsieh, Jorng-Tzong Horng, Gen-Huey Chen
National Central University

CT1.2.3 Fault Notification Extension in Support of BSS 2G Siemens
Linawati
Udayana University

CT1.2.4 A Fast and Seamless Route Repairing Algorithm for Ad-hoc Networks
Shih-Chang Huang
National Formosa University

Room A

July 13, 2012
15:30 - 17:00

CT2.4: Mobile & Wireless Broadband

CT2.4.1 High Isolation Compact WLAN/WiMAX Antenna
Bing Yu, Junxiang Ge
Nanjing University of Information Science & Technology

CT2.4.2 Differential BPSK Modulation for Cooperative Communication Systems in Time-Selective Rayleigh Fading Channels
Chi-Hsiao Yih
Tamkang University

CT2.4.3 An Efficient Cost-based Location Service Protocol for Vehicular Ad Hoc Networks
Chih-Shun Hsu, Shen-Wen Wu
Department of Information Management, Shih Hsin University

CT2.4.4 A Modified Low PAPR Space-Frequency Block Coding Scheme for SC-FDMA
Chih-Yao Huang, Wei-Jay Chang, Li-Chung Chang
National Taiwan University of Science and Technology

CT2.4.5 Ultra Low Profile Antenna for 2.45 GHz Wireless Communication
Erfan Rohadi, Mitsuo Taguchi
Nagasaki University and State Polytechnic of Malang

CT2.4.6 SINR Balancing Techniques for a Cognitive Radio Relay Network with Multiple Peer-to-peer Users
Georgia Bournaka, Sangarapillai Lambotharan, Fotis Lazarakis
Loughborough University

Room C

July 13, 2012
15:30 - 17:00

CT5.1: System & Applications

CT5.1.1 Flu Incident Reporting on Mobile Phone
Nichapa Panyasoponlert, Pawat Piboonudompornkul, Kurt T. Rudahl, Sally E. Goldin
King Mongkut's University of Technology Thonburi (KMUTT)

CT5.1.2 Medical Image Watermarking with Tamper Detection and Recovery Using Reversible Watermarking with LSB Modification and Run Length Encoding (RLE) Compression
Tjokorda Agung Budi Wirayuda, Adiwijaya, Febri Puguh Permana
Institut Teknologi Telkom

CT5.1.3 Facial Expression Based Computer Cursor Control System for Assisting Physically Disabled Persons
Vasanthan Maruthapillai, M. Murugappan, R. Nagarajan, Bukhari Ilias, Letchumikanth
University Malaysia Perlis

CT5.1.4 Cloud Adaboost Feedback Training Machine for Outside Available Parking Spaces Query Service
Jie-Qi Huang, Ming-Shi Wang
National Cheng Kung University Department of Engineering Science

Room A

July 14, 2012
10:45 - 12:00

CT2.5: Mobile & Wireless Broadband

CT2.5.1 An Overheard-Based Relay-Assisted MAC Protocol Using Symbol Level Network Coding
Yuh-Shyan Chen, Chih-Shun Hsu, Jhao-Yen Wei
Department of Computer Science and Information Engineering, National Taipei University

CT2.5.2 Joint Network/Channel Coding for Wireless Networks
Zid Youssef, Sonia Ammar, Ridha Bouallègue
NEST

CT2.5.3 A Delay-Bounded Routing Protocol for Vehicular Ad Hoc Networks with Traffic Lights
Yuh-Shyan Chen, Chih-Shun Hsu, Yi-Ting Jiang
Department of Computer Science and Information Engineering, National Taipei University

CT2.5.4 A Comparision of Frequency/AmplitudeModulation Scheme in Cognitive Radio Envoirnment
Freeha Azmat, Junaid Imtiaz, Muhammad Hashim, Ithisham-ul-Haq, Malik Rizwan
Bahria University

Room B

July 14, 2012
10:45 - 12:00

CT6.1: Satellite & Aerospace

<u>CT6.1.1</u> **Bench Model Design of The Electrical Power System for Iinusat-1 NanoSatellite**
Arbai Yusuf, Gunawan Setyo Prabowo
LAPAN

<u>CT6.1.2</u> **Examining Quantum Key Distribution Protocols in Laser Based Satellite Communications**
Laszlo Bacsardi, Andras Kiss, Mate Galambos, Sandor Imre
Department of Telecommunications, Budapest University of Technology and Economics

<u>CT6.1.3</u> **Introduction of the IiNUSAT Inter-Satellite Link System**
Arifin Nugroho, Nurwahidah Jamal, Suryadi Tanuwijaya
ITTI

Efficient Normal Peers Group Recovery in Hierarchical Peer-To-Peer

Sri Wahjuni[1], Anak Agung Putri Ratna[2], Kalamullah Ramli[3]
Department of Electrical Engineering, Faculty of Engineering, University of Indonesia
Depok, Indonesia
[1]sri.wahjuni@ui.ac.id, [2]ratna@eng.ui.ac.id, [3]k.ramli@eng.ui.ac.id

***Abstract*— While structured peer-to-peer (P2P) offers benefits on its scalability and efficiency in performing a successful query lookup, its stability is suffered from the dynamics of the overlay structure caused by churn. In a structured hierarchical P2P, in addition to potential decreases of the system performance, the superpeer failure cases also forced the normal peers under its responsibility to disconnect from the system.**

The ultimate goal of our work is to develop a collective rejoin algorithm that provides an efficient mechanism for the normal peers to rejoin to the system once a superpeer failure occurs. We implemented two-layer hierachy architecture, in which nodes with higher capability are placed in the upper layer and organized in a Chord ring. These nodes act as a gateway for the lower layer in which other nodes are grouped. Each group is organized in a star structure, and each member of the group is connected directly to the related gateway node. We expect our proposed architecture and algorithm produces less traffic load than the individual rejoin approach. Thus, the performance degradation caused by the normal peers rejoin process, as an impact of churn, can be minimized.

Keywords-component; collective rejoin, churn, hierarchical P2P, pervasive environment, superpeer failure

I. INTRODUCTION

Regarding to [1] the P2P architectures are categoried into unstructured and structured. The well known centralized unstructured architecture is Napster [2], while its contradictive -fully distributed- architecture is Gnutella [3]). The compromised architecture between the centralized and fully distributed is hybrid architecture like Kazaa [4]). The natural benefits of the unstructured P2P architecture are their simplicity and stability, although they suffer from high traffic load as the impact of the implementation of flooding search method. The later generation of P2P is structured architecture, that based on distributed hash table (DHT) for indexing the peer as well as for the shared-object. Some examples of this category are CAN [5], Chord [6], Pastry [7], and Tapestry [8]. The use of DHT addresses the efficiency and scalability problems which are the main problems in unstructured architecture. However structured P2P is highly influenced by the dynamics of peers, which is a natural characteristic of heterogeneous network.

Comparing to other DHT based P2P, Chord has the best scalability with the maximum nodes to retrieve to find the object is O(log *n*) for an overlay network built by *n* peers [6]. Chord implements consistent hashing to define an identifier of a node and the object key of the shared objects/services. The identifier of a node is obtained from the hash value of the node identifier (such as SHA-1) and the key identifier is obtained from the object identity (such as file's name). The identifier is ordered in a modulo 2^m ring size, for m-bit identifier, thus give $n=2^m$ is the size of the network. The key *k* is assigned to the first node that has the same value or follows the identifier k in the identifier space. This node is the successor node of key *k*, called successor(*k*). Suppose the identifiers are represented in a ring with the number of 0 to 2^m-1, then the successor(*k*) is the first node (succeed *k*) in the clockwise direction. Each node has a finger table (with maximum entry m) and a successor list. When a node sends a query, a receiving node performs a lookup in its finger table to find the suitable finger *i* entry. The query will be forwarded to the node s which is the successor of i that responsible to the queried object. Then node s sends the queried node address direct to the requester node, so the requester node is able to contact the owner of the queried

Fig 1. shows the steps when a node n joins the system at a position between *p* and *s*. It will contact the node s and tells *s* that currently s is its successor (Fig.1.a). Node *s* then updates its predecessor entry to point to new joining node n to replace the former node *p* (Fig.1.b). This mechanism will not give node *p* information about the node *n* as its new successor automatically. Each node in the Chord ring performs stabilization periodically by sending a message to its successor nodes (node also *p* sends a message to node *s*). The successor node responses by sending message contain the information of its current predecessor. By receiving this response, node *p* knows that the message contains identity of node *n* (not *p*), then *p* should to replace its successor node to *n*. The last step in this stabilization mechanism is the node *p* sends a message to node *n*, telling that *n* now is its successor, then node *n* should point its predecessor pointer to *p* (Fig.1.c).

When a node leaves the system according to the voluntary leaving or any disconnection problem (node failure), the system does not perform any related

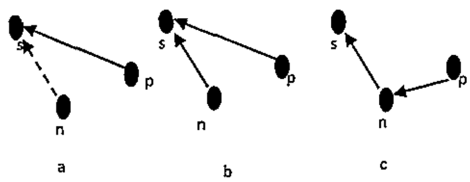

Figure 1 Node joins mechanism in Chord

stabilization mechanism. While in the voluntary one, the leaving node transfers its shared-object information to its successor node before leaving, it does not happen in node failure event. The fundamental thing in a P2P networks is the ability to maintain its overlay due to the occurrences of churn. Churn is a dynamic condition of P2P network when a node joining or leaving the system concurrently [9]. The increasing frequency of churn event (churn rate) will potentially interfer the stability of the overlay. This is a critical condition in a Chord-based P2P, since the validity of its finger table is an important point in supporting a succesfull lookup query.

In relation to the growing up of the device capabilities in accessing application included the Internet, the hierarchical P2P architecture [10-14] is a suitable architecture to handle the heterogeneity of the peers participants. The generic framework of two-level hierarchical P2P was proposed in [10]. The higher capability devices formed the first level of the hierarchy, while the lower ones grouped into clusters that formed the second level. The peers in clusters communicate to its upper level through various schemes, and communicate with other peers in other clusters through the gateway peer. In a common way, the peers are clustered based on their capabilities, local proximity, or services provided.

Although the hierarchical P2P architecture accommodate the heterogeneous of networks, the heterogeneity of the peers capabilities brings the consequence of the higher probability of churn. Besides decreases the system performanced, the superpeer failure also forces the normal peers to be disconnected from the system. Therefor the development of normal node handling algorithm is as important as the churn detecting and handling algorithm. As showed in our preliminiary research work [15] the join request into Chord ring produced a significant ammount of latencies, that in turn impacted to the degradation of successful loookup query latency. We argue that the individual recovery mechanism proposed in [12] creates very high amount of rejoin processes which in turn degrades the performance of the system significantly. On the other hand, while the backup membership in the Chord ring system as in [13] is a valuable scheme, the flooding mechanism in their lower layer clusters is too extensive in consuming the local network resources. This paper will describe our proposed algorithm to perform an efficient normal peer recovery in the case of super peer failure occurance through the collective rejoin.

In order to conduct the reader to have the clear understanding of our proposed work, we organized the paper as follows: the section I contains the introduction that will provide the background concept of the P2P overlay network. In section II we identify some current related works and the position of our research. Our proposed system architecture and algorithm are describe in section III. In section IV we summarize the conclusion of our topic discussed in this paper.

II. RELATED WORKS

Some researches based on one layer/flat Chord have been published [16-19], which are concern in how to perform a successful lookup query efficiently. In the pervasive computing network, since the nodes that involve in the system have heterogeneous capabilities, then the hierarchical P2P architecture is a suitable architecture to handle this natural characteristic of pervasive computing. The super peer concept, in which a peer with higher capacity handles more responsibility than others, was used in architecture proposed by [11].

In [10] the author proposed a generic framework of some type of two-level hierarchical architecture for DHT-based P2P. The nodes were classified into two categories, the super peers which are nodes with relative longer connection time to the system and the regular peers, those which shorter connection time. The super peers were connected each other in a ring-like structure (such as Chord) and act as a gateway for the reguler peers. The reguler peers were connected to the system through a suitable super peer in a direct connection, fully mesh, or organized in a cluster or other ring-like structure. In the work of [14], the author implemented the Chord-based structure for the top layer of the hierarchical P2P as well as the lower layer. This generic framework of hierarchical P2P [10] also implemented and modified by [20] and [21], which organized the reguler peers in lower layer using direct connection to the super peer, the same architecture as we are studying on. Both of them were focused on finding the optimal proportion of the super peer and the regular peer to reach a better performance. While [20] aim was to minimize the total traffic without overloading the super peer, the [21] focused on minimizing lookup delay of the system. To the best of our knowledge, none of them proposed a specific normal peers recovery protocol in the case of super peer failure occurance, as the ultimate goal of our research.

The closest work in spirit to ours was proposed by [13], that allowed more than one super peers in a cluster-based lower layer, in order to minimize the impact of super peer failure. The cluster structure used flooding for internal communiation, while the upper layer was organized in a Chord-based ring. Instead of implemented flooding-based cluster as they did, we organized the reguler peers in a direct connection to the super peer as in [10]. In [12], this structure was proposed as an optimal design of hierarchical DHT system. We also modified the peer atributes and failure detection mechanism to meet our protocol requirements.

In order to perform an effective recovery, a node failure detection must be performed correctly. Zhuang in [22] provide a study of some type of failure detection algorithm and their impact in node failure detection time and control overhead (bandwidth consumed for maintaining the overlay structure). From the experiments, the baseline algorithm as implemented in Chord provide a relative constant control overhead and detection time in the presence of churn.

III. SYSTEM DESIGN

A. System Architectures

The hierarchical architecture we implemented is a two-level hierarchy P2P which is the top layer is contained of super peers (those nodes with high capacity) and the lower layer is contained of groups of normal peers. In each group, the normal peer is connected directly to the super peer of the relevant group (star structure). The

TABLE I. NEIGHBOR TABLE

Neighbor Table		
Leaf node ID	*IP address*	*Max_capacity*
nn_1	IP1	0
nn_2	IP2	1
...	...	...
nn_m	IPm	0

construction of the group is based on the nearest proximity location, which was obtained when a node broadcast a *find_overlay()* message. It will send a *join_req* to a node (that is a super peer) that give the smallest round-trip-time (RTT) response.

In order to support our normal peer recovery algorithm, we proposed *neighbor_table* as an additional atribute for both the super peer and normal peer (see TABLE I). This table is used to record all the members of the group in fields of *<node_ID,IP_addr,max_capacity>*. The *max_capacity* field is in boolean data type which is used to record the superior peer among other normal peers in the group. If the peer has the highest capacity among others in the group then this field will be set to 1, otherwise to 0. The other tables as known in conventional Chord, such as finger_table, successor_list, and reference_table are still used as usual. Other additional attributes that we proposed are time (*age*) and capacity (*cap*), which are refer to how long has the node been joining the system and the bandwidth connection of the node respectively.

B. The Proposed Algorithm

In order to support our proposed algorithm in performing efficient normal peers group recovery in the presence of super peer failure, we also modified conventional Chord join and maintenance processes and proposed a new process to detect a super peer failure and performing a a collective rejoin process on behalf of the disconnected normal peers.

B.1 The nodes join algorithm

The proposed nodes join algorithm is designed to handle both joint requests come from a pure new node or a node that has been elected as a new elected super peer. For the first type of join request, the new node could be accepted as a normal peer in that group or as a new super peer. While for the second type, it will become a new super peer. Since a new node does not have any information about other existing peers, then it will broadcast the *find_overlay ()* message and contact the super peer that sent *join_offer()* message with the smallest RTT (nearest geographic proximity). From the view of existing peers, only the super peers that have the right to send *join_offer()* message as a response to the coming *find_overlay()* message. A normal peer that received *find_overlay()* message must forward it to its super peer. Since the message contained the IP address of the requester node, than the super peer may send the response directly to the requester node. The response message from the super peer contained its own IP address that can be used by the requester node to send a join request message (*join_req()*). The *join_req()* message contains the tuple of *<IP_addr,node_ID,cap,age>*.

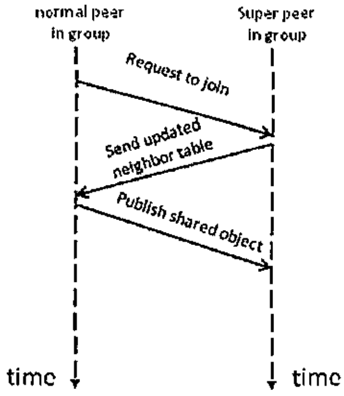

Figure 2. Node joins as normal peer

Fig. 2 shows the cheme when a node join as a normal peer.

1) A node sends a join request message (join_req()) to the closest node with regards to the aforemention nearest geographic proximity. If the joining peer meet super peer's qualifications then the join request will be processesd as in conventional Chord join request mechanism. Otherwise, the node will accepted as a new normal peer in that group, and continue to the step 2). While accepting a new member, the super peer has to update its neighbor_table.

2) The updated neighbor_table is sent to new normal peer as a response to the joining request (send_neighbor()). For other members, the updated neighbor_table will be sent along with response message (maint_int()) when they send a keep_alive() message in maintenance procedure.

3) As soon as receiving the neighbor_table (request acceptance state), the normal peer sends shared-object that it wants to offer to the system through the super peer (n_publish()).

A node will be treated as a new super peer when meet requirements as follows (in order of the priority):

1) A normal peer that has been elected by its group as a new super peer, since the failure of former super peer (recognized by its age > 0)

2) A node with the capacity (cap) value greater than current super peer.

If *SP* is a set of super peers (sp_s) and *NP* is set of normal peers (np_s), then if a node x join the system through super peer y, we can summarize the above algorithm in a formal description as below :

$$x = \begin{cases} SP & ; x.age > 0 \text{ or } x.cap > y.cap \\ NP & ; other \end{cases} \quad (1)$$

B.2 Maintenance algorithm

The maintenance algorithm consist of two types of maintenance : the inter-group maintenance and the internal group maintenance. The inter-group maintenance

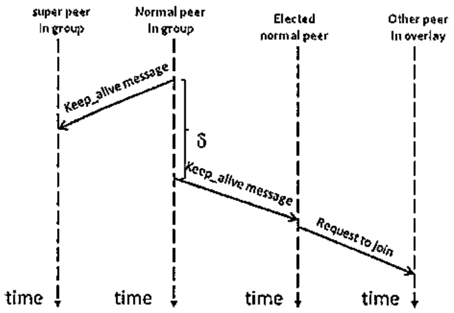

Figure 3. Super peer failure detection and group recovery

is performed as in conventional Chord, while the internal group maintenance procedure is performed as follows:

1) Normal peer p sends keep_alive() message to the super peer sp_i periodically.

2) When receiving the keep_alive() message from a normal peer, super peer sp_i responsed by sending a maint_int() message along with the updated neighbor_table (piggybacking).

B.3 Super Peer Failure Detection and Group Recovery

The above internal group maintenance is performed every Δ time and we implemented the active keep-alive based on baseline algorithm as a failure detection algorithm [22]. The detection failure and group recovery processes are performed as ilustrated in Fig.3 :

1) The normal peer p that does not accepted any response from the super peer sp_i in δ time decided that the super peer was failed.

2) Once the super peer is detected fail, the normal peer p lookup into its neighbor_table to find the peer s that has the max_capacity field set to true (elect_sp()).

3) The normal peer p contact the normal peer s by sending keep_alive() message.

4) As normal peer s received keep_alive() message from other normal peers indicated that the sender ask him to perform join process as describe in previous sub section. In this work, we optimized the use of keep_alive() message not only for maintenance process as usual but also for group recovery process.

IV. CONCLUSION

In this paper we described a collective rejoin algorithm to provide an efficient way for normal peers to rejoin to the overlay in the case of super peer failure in a two-level hierarchical P2P. We showed that in addition to the common use of peers capability information to categorize a peer, it also can be utilized to determine a new super peer to replace the failure super peer of a group. Another optimization that we proposed is the used of *keep_alive()* message in supporting group recovery process. Our near future work is to implement the algorithm in order to analize the efficiency level of the normal peers collective rejoin algorithm comparing to the individual one.

ACKNOWLEDGMENT

Sri Wahjuni thanks to Department of Computer Science, Faculty of Mathematics and Natural Sciences, Bogor Agricultural University (IPB), in which she is a faculty member, for supporting the publication of this paper.

REFERENCES

[1] J.Kangasharju, "*Peer-to-Peer Systems*", Handbook of Research on Ubiquitous Computing Technology for Real Time Enterprises pp.174-189, 2008.

[2] http://www.napster.com

[3] http://www.gnutella.com

[4] http://www.kazaa.com

[5] S. Ratnasamy, P. Francis, M. Handley, R. Karp, and S. Schenker, "*A scalable Content-Addressable Network*". Proceedings of the 2001 Conference on Applications, Technologies, Architectures, and Protocols For Computer Communications (SIGCOMM '01) pp. 161–172, 2001.

[6] I.Stoica, R.Morris, D.Liben-Nowel, D.Karger, M.F. Kaashoek, F. Dabek, H. Balakrishnan , " *Chord: A Scalable Peer-to-peer Lookup Service for Internet Applications*", Proceeding of SIGCOMM 2001, ACM Press 2001, pp. 149-160, 2001

[7] A. Rowstron and P. Druschel, "Pastry: Scalable, decentralized object location, and routing for large-scale peer-to-peer systems", Lecture Notes in Computer Science, 2218:329–350, 2001

[8] B. Y. Zhao, L. Huang, J. Stribling, S. Rhea, A. D. Joseph, and J. D. Kubiatowicz, "*Tapestry: A Resilient Global-scale Overlay for Service Deployment,*" *IEEE Journal on Selected Areas in Communications*, vol. 22, no. 1, pp. 41–53, 2004.

[9] S. Rhea, D. Geels, T. Roscoe, J. Kubiatowcz, "*Handling Churn in a DHT*", The USENIX 2004 Annual Technical Conference, Boston, MA, USA,2004.

[10] L. Garcies-Erice, E.W. Biersack, P.A. Felber, K.W. Ross, and G. Urvoy-Keller, "*Hierarchical Peer-to-Peer Systems*", Proceeding of the ACM/IFIP International Conference on Parallel and Distributed Computing (Euro-Par), Klagenfurt, Austria, 2003

[11] A.Montresor, "A Robust Protocol for Building Superpeer Overlay Topologies", Technical Report UBLCS-2004-8, 2004.

[12] S. Zoels, Z. Despotovic, W. Kellere, "*On hierarchical DHT systems – An analytical approach for optimal designs*", ScienceDirect, Computer Communications 31 pp. 576–590,2008.

[13] Z.Peng, Z.Duan, J.Qi, Y.Cao and E.Lv, "*HP2P: A Hybrid Hierarchical P2P Network*", Proceeding of the 1st International Conference on the Digital Society (ICDS'07),2007.

[14] M.Pandey, S.M.Ahmed, B.D.Chaudhary, "*2T-DHT: A Two Tier DHT for Implementing Publish/Subscribe*", Proceeding of The International Conference on Computational Science and Engineering (CSE '09) pp.158-165, Vancouver, 29-31 Aug. 2009

[15] S.Wahjuni and K. Ramli, " Considering The Nodes Join/Leave Behavior in The Analysis of Chord Stabilization in The Heterogeneous Network" , International Journal of Computer Science and Network Security, VOL.11 No.12 pp.57-61, 2011.

[16] Y.S.Shan, Y.Jiong, Kamil, S.Q.Gang, "*DR-Chord : An Efficient Double-Ring Chord Protocol*", Proceeding of The 6th International Conference on Grid and Cooperative Computing (GCC 2007), 2007.

[17] W.Ziyang and Y Siqing, "*Chord Algorithm Based on Two-way Routing Table*", Proceeding of The 2010 International Conference on Computer Application and System Modeling (ICCASM 2010),2010.

[18] D.G. Zhang, X.Y.Chen, C. Lt, L.Q.Li, "*A Pervasive Service Discovery Strategy Based on Peer to Peer Model*", Proceeding of the 2010 International Conference on Modelling, Identification and Control, Okayama, Japan, July 17-19,2010

[19] H.K.Kwak, Y.H.Yoon, C.G.Kim, K.S.Chung, "*An enhanced reactive Chord for mobile networks*", IEICE Electronics Express Vol.7 No.10, 2010

[20] S.Zoels, Q.Hofstatter, Z.Despotovic, W.Kellerer, "Achieving and Maintaining Cost-Optimal Operation of a Hierarchical DHT System". Proceeding of The IEEE International Conference on Communications ICC '09, 2009.

[21] J.Li, C.Li, Z.Fang, H.Wang, " Layer Optimization for DHT-based Peer-to-Peer Network", Proceeding of The 10th Annual Conference on Networks (ICN) 2011 pp.382-387.

[22] S.Q. Zhuang, D.Geels, I.Stoica, R.H. Katz, "*On Failure Detection Algorithms in Overlay Networks*", Proceedings of The IEEE 24th Annual Joint Conference of the IEEE Computer and Communications Societies INFOCOM, 2005.