A Hierarchical DHT for Fault Tolerant Management in P2P-SIP Networks

Ibrahima Diané Ibrahima Niang Bamba Gueye
Université Cheikh Anta Diop de Dakar
Département de Mathématiques et Informatique
Dakar, Senegal
ibrahima.diane@ucad.edu.sn iniang@ucad.sn bamba.gueye@ucad.edu.sn

***Abstract*--This paper focuses on fault tolerance of super-nodes in P2P-SIP systems. The large-scale environments such as P2P-SIP networks are characterized by high volatility (*i.e.* a high frequency of failures of super-nodes). Most fault-tolerant proposed solutions are only for physical defects. They do not take into account the timing faults that are very important for multimedia applications such as telephony. We propose *HP2P-SIP* which is a timing and physical fault tolerant approach based on a hierarchical approach for P2P-SIP systems.**

Using the Oversim simulator, we demonstrate the feasibility and the efficiency of HP2P-SIP. The obtained results show that our proposition reduces significantly the localization time of nodes, and increases the probability to find the called nodes. This optimization allows to improve the efficiency of applications that have a strong time constraints such as VoIP systems in dynamic P2P networks.

Keywords: VoIP, P2P, SIP, DHT, Fault tolerance

I. INTRODUCTION

P2P networks follow a different paradigm compared to client-server based systems. The key underlying attribute is that each node (or peer) participates in the network by offering and using services at the same time [1]. The typical topologies of P2P networks are three types: central server P2P, unstructured P2P and structured P2P. The unstructured P2P and structured P2P are both purely distributed topology and most of research works are based on them. The unstructured P2P network is based on random graph and uses flooding [2]. The unstructured P2P model is scalable under churn (*i.e.* high dynamicity). Nevertheless, the use of flooding mechanism during the lookup phase leads to no-scalable and inefficient models. The structured P2P network is based on DHT (Distributed Hash Table). By "structured" we mean that the P2P overlay topology is structured and controlled. In such case, the available resources are placed in specified locations. In the DHT, each item has a key and a value. The lookup service determines the node that is responsible for a given key. This method is more efficient than flooding and has a better scalability [3, 4, 5, 6].

Recently, few works [7, 8, 9] propose to use P2P technology in the context of Voice over IP (VoIP) or Internet Telephony, in particular with the *Session Initiation Protocol* (SIP). However, *Skype* [7] uses its own, but not standard, protocol to initiate and modify sessions. The main idea is to replace the centralized components of the client/server based SIP protocol with a P2P system (P2P-SIP). In so doing, a higher level of robustness and scalability can be achieved.

Many works [10, 11, 12] have been proposed to overcome the limitations of super-nodes with respect to fault management. Nevertheless, they are only related to file sharing but not VoIP services. In file sharing systems, *physical faults* are more critical than *timing failures*. Timing failures can be defined as a short-lived failure. In other words, shortly after a failure the node becomes available. In contrast to P2P-SIP based VoIP, timing failures of super-nodes should be addressed in order to achieve a good quality of service. By definition, a super-node is any node that also serves as one of that network's relayers, handling connection for the other users.

It is worth noticing that any technique which deals with fault management should include theses two proprieties: (i) *Fault detection*: in order to provide countermeasures, the first step, that a system should realize, is to detect that a specific functionality is or will be faulty. (ii) *Fault Recovery: a*fter the system has detected a fault, the next step is to prevent or to recover from it. The main technique to achieve this goal is to replicate the components (e.g super-nodes) of the system that are crucial for its running. Indeed, it is mandatory to react rapidly during the potential failures of such critical components.

In P2P-SIP systems [8, 9, 13], the proposed schemes do not the difference between physical and timing failures. After a given delay, if a super-node does not send either *refresh messages* to its successors, or acknowledgments to its ordinary nodes, it will be considered as physically down. In fact, P2P-SIP systems use an approach based on a logical ring to implement failure detection. This ring determines the patterns of exchanged "*keep alive messages*" and their direction. The goal of the keep alive messages is to discover nodes that are offline. This mechanism limits the number of keep alive messages issued on the network and offers good properties of scaling. Existing solutions [8, 9, 13] that deal with super-nodes fault tolerance are generally based on replication techniques. Indeed, replication allows management of multiple copies which can diverge, i.e. have different values at a given time but eventually converge towards the same values.

In P2P-SIP telephony, existing mechanisms for super-nodes fault tolerance have several drawbacks [8, 9]. In fact, solutions do not fully support all types of failures. The limitations are twofold:

Firstly, there is no automatic recovery mechanism after the failure of a given super-node because it is only "attached" to one *Ordinary Node* ("ON"). During the failure of a given super-node, its attached ON could not make a call until the next refresh where it can choose a new super-node. This is not suitable for applications that have time constraints such as VoIP.

Secondly, the proposed solutions [9, 14] do not take into account the timing failures of super-nodes which are an important aspect for multimedia applications such as telephony.

In this paper, we present an efficient fault-tolerant approach, called *HP2P-SIP*, based on P2P-SIP VoIP systems. We use a hierarchical DHT based on Chord [15] to implement our approach. Since the timing failures of super-nodes were not addressed by existing solutions [10, 11, 16], we propose a mechanism to detect, manage, and recover these faults in transparent manner with respect to users. The main idea is to setup a three-tier architecture in contrast to previous approaches [10, 16]. The goal is to use a subset of nodes called "*light super-node"* in a third level of our architecture. It should be noted that the first level is formed by ONs, and the second one is composed by super-nodes. A light super-node is a node that stores the registration of a set of ONs in order to react when super-node's failures happen in the network. Theses registrations are sent to the light super-node by the set of super-nodes that own this set of ONs. Put simply, each light super-node allows to recover super-node's failures by managing a set of ONs. With this three-tier approach, we mitigate the delay for discovering a node, and increase the timing fault detection. Therefore, we augment the probability to establish a communication.

The rest of this paper is organized as follows. Section II describes related works in the field of fault tolerance. Section III presents the HP2P-SIP approach. In Section IV we evaluate the performance of HP2P-SIP. Finally, we conclude and present some research perspectives in Section V.

II. RELATED WORKS

Stoica et *al.* propose Chord [15] which is a ring-based structured peer-to-peer architecture where each node is assigned an m-bit identifier. Note that *m* is the number of bits generated by a standard hashing algorithm such as SHA-1. Chord uses a distributed hash tables (DHTs) and each node keeps track of its predecessor, successor. Furthermore, each node maintains a table or a state called finger table containing *m* entries. It should be noted that in Chord, these entries represent the m successors of a given node.

The traditional SIP based VoIP systems employ SIP registrars, SIP proxy servers, and STUN servers. It means that users who want to participate in session should register with a registration server using their identifiers. With the help of registrar servers, users can localize other session partners in the network and also initiate sessions with them through proxy servers [17].

The deployed SIP infrastructures rely on centralized entities which do not provide scalability and are not tolerance. Recent researches [8, 9, 13] propose to use P2P approach in SIP protocol. The idea is to replace the centralized components with a distributed P2P system. The name P2P-SIP is commonly used for these systems. A key SIP functionality, which is implemented by the use of P2P system, is the registration and lookup of user's localization information. This phase is a crucial component for the call establishment process.

Peer-to-Peer Session Initiation Protocol (P2P-SIP) [8, 9, 14] is proposed to combine SIP and P2P by leveraging the inherent advantages of Distributed Hash Table (DHT) such as scalability, robustness, etc. The goal is to enable multimedia session in a distributed manner. It exists two approaches for combining SIP and P2P: P2P-over-SIP and SIP-using-P2P. In both approaches, all hosts fairly participate in a single overlay. The DHT is "the heart" of P2P-SIP network. It is not a single entity but it is distributed overall nodes. Most of previous works [9, 13] use Chord to provide DHT functions. A P2P-SIP overlay is set of super-nodes organized in peer-to-peer manner in order to enable real-time communication between client nodes, using SIP. These super-nodes are use by client nodes as bootstrap.

The authors of [11] propose a fault-tolerant method for super-peers that compose the network. Super peers are organized into groups, and they select *k* super peers in each group to become virtual super peer. In other words, the virtual super peer acts as landmark for each group. Afterwards, the virtual super peer will be used, in a round robin fashion, by the remaining super peers in order to instantiate communication. Note that super-peers refer also to super-node.

Afterwards, the authors of [16] present an efficient fault-tolerant approach for the super-peers in P2P file sharing systems where peers are organized into multiple groups. In each group, we have a special peer called super peer to serve the regular peers within the group. In this hierarchical architecture, if the super peer fails, any file queries, from peers that are managed by this super peer, will not be delivered. To overcome this limitation a multiple publication techniques are proposed in [16] in order to ensure that each peer is served by more than one super peer.

Zhu et *al.* in [12] propose to organize super peer into two adjacent overlay networks in order to deal with the failures of super peers in P2P file sharing systems. By taking into account the different resources (e.g. network bandwidth, storage capability, and processing power) of participating super peers, the peers with large resources are selected as super peers for the new overlay. Furthermore, the super peers are self-organized as a secondary overlay in order to manage the super peers that have fewer resources and they are located in first overlay.

III. HP2P-SIP ARCHITECTURE

In this section, we present a new mechanism for super-nodes fault tolerance in the P2P-SIP telephony systems. As mentioned, the ONs are organized around the super-nodes. Each ON is attached to a super-node. Therefore, a failure of the super-nodes put temporally

offline all ONs that are related to it. HP2P-SIP, by considering a light super-node in addition, allows to reach any ON even if the failure of the attached super-node. We also propose an optimization of physical failures and introduce new timing failures management.

A. HP2P-SIP approach

We propose a hierarchical architecture with three levels in contrast to the classical P2P-SIP architecture which is based on two levels. Figure 1 shows the architecture of HP2P-SIP. As illustrated, it is based on three-tier approach where we have respectively ONs, super-nodes, and light super-node. The first two levels are the same compared to those described in the classical P2P-SIP architectures. We propose a third level for light super-nodes. A light super-node is a node that stores the registrations of some ONs. It participates in the routing localization process if a super-node is breakdown. It should be noted that the light super-nodes are chosen from super-nodes. The choice guided by the following parameters: life time node on the network, bandwidth, speed of CPU, memory size.
The main motivations of HP2P-SIP are:

1) The accessibility of ordinary nodes during the physical failure and the timing failures of super-nodes.
2) To mitigate the localization delay of Remote Ordinary Nodes (RON), during the call establishment, when its attached super-node is breakdown.

Since the number of refreshment messages for each ON will be multiplied by the number of attached super-nodes, we choose to consider only two levels of super-nodes. The motivations of the utilization of light super-nodes are twofold: (*i*) the light super-nodes enable to overcome the physical faults for attached super-node by reducing the localization delay of RON; (*ii*) they are used to manage the timing failures of super-nodes over the ring. In HP2P-SIP, each ON is attached to a super-node and a light super-node in order to add a redundancy (Figure 1) in the network. Note the number of light super node is less than the number of super-node.

In HP2P-SIP architecture, each ON is attached to two super-nodes. The first acts as the super-node which has the responsibility of the ON's key registration. Note that this key is obtained from a hash function. We consider the hash function used by Chord in [15]. However, this hash function uses node's SIP identifier in order to generate a key. The second, called the light super-node, is chosen according to the same key. We argue that this mechanism increases the availability of the overlay network with respect to results obtained in Section IV.

B. HP2P-SIP functionalities

During the starting phase, an ON tries to connect to a super-node. In so doing, the ON discovers a super-node that acts as bootstrap (gateway). After this discovering phase, the ON tries to register into the discovered super-node. Afterwards, the requested super-node replicates the ON's records towards a light super-node. The different phases can be described as follows:

1) *Super-node discovery:* Firstly, the ON tries to discover a super-node. It sends a multicast message using the address 224.0.1.75. It should be noted that

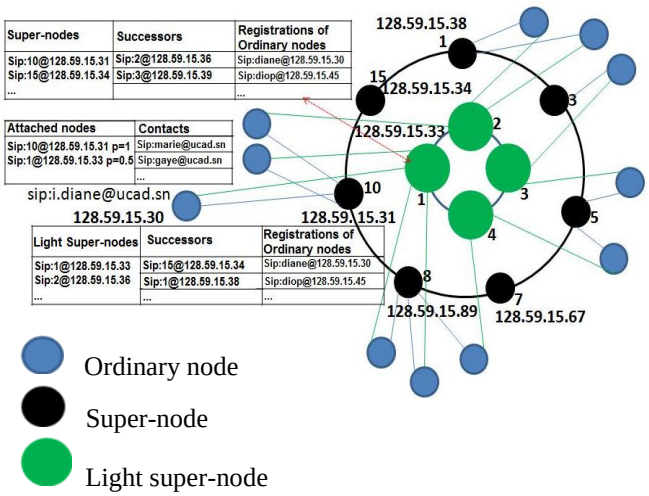

Figure 1. HP2P-SIP architecture.

this address is a well known SIP multicast address. If a super-node receives the multicast message, it responds with a unicast message where it adds its own address. If the ON receives multiple replies, it chooses only the super node that has firstly answered.

2) *Registration procedure:* After the discovering phase of a given super-node, the ON tries to register to the super-node that has the responsibility of its own key. Note that this given super-node can be either the super-node discovered at the first step or another super-node which is located around the ring (Figure 1). In such case, the super-node has the role to discover the appropriate super-node that is responsible to ON's key. Afterwards, the ON sends its registration to the super-node found previously. If this super-node is not the responsible of ON's key, it sends the registration to the appropriate super-node. Furthermore, the super-node sends the registration request to the appropriate light super-node. The light super-node is chosen based on the key value of the ON with respect to Chord DHT. An acknowledgment is sent by the light super-node to the super node that confirms the registration of the ON according to the light super-node. As well, an acknowledgment is sent by the super-node to the ON. This acknowledgment contains the address of the light super-node. It will be used by the ON to establish a communication.

After these steps, the super-node that is responsible to ON's key replicates the registration on its *m* successors in order to tolerate physical failures; *m* is also the size of keys in the considered network.

3) *Ordinary node location:* for instance, Figure 2 shows the different steps that follow the node having the SIP identifier "*sip:i.diane@ucad.sn"* in order to

establish a communication with "*sip:marie@ucad.sn"* which is the SIP identifier of the RON. To localize a RON ($N_{r\text{-}id}$), the local ordinary node ($N_{l\text{-}id}$) sends a localization request message to its attached super-node. This step is illustrated by the arrow labeled (1) (Figure 2). The message is relayed to the super-node that manages $N_{r\text{-}id}$'s key (step 2 in Figure 2). The super-node that is attached to $N_{r\text{-}id}$ processes the message and returns a response to $N_{l\text{-}id}$ through its attached super-node. This is illustrated by step 3 and 4 in Figure 2. It is worth noticing that this message contains the IP address of the RON. After the localization process of the RON, the communication can be established between both ONs.

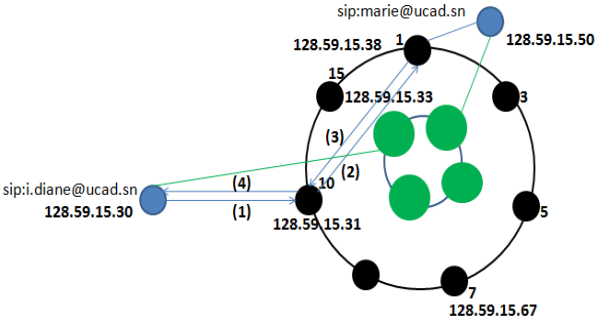

Figure 2. Localization of an ordinary node with HP2P-SIP.

C. Fault management of super-nodes

This section presents the used mechanisms to deal with physical and timing failures of super-nodes in HP2P-SIP architecture. We take into account two types of faults:

1) Physical fault management:

Before recovering a fault, we should be able to detect it. Therefore, we use as in the classical P2P-SIP [4, 7], an approach based on DHT to implement the detection of physical failures. The failures are detected using the refresh messages. An ON detects the failure of its attached super-node if it does not receive a refresh message reply during a fixed amount of time. However, we propose a fault recover mechanism in order to mitigate the delay which is induced by the search of new super-node. The recovering phase in HP2P-SIP is done according to the following scenarios:

- If the breakdown super-node has the responsibility of the local ON's key then the ON delegates its light super-node to establish the communication. Therefore, the light super-node initiates the localization of the remote ordinary node. In this case, the goal of the light super node is to find the light super node that manages the RON. Since the ON's super-node is breakdown, the communication between both nodes will be managed only by the light super-nodes even if the super-node where is attached the RON is available. This approach enables to obtain a gain during the localization routing process.
- Let's assume that the breakdown super-node is among the successors of the super-node that has the responsibility of the ON's key. In such case, the faulty super-node should be ignored in favor of the immediate known successors (super-node) along the ring. Therefore, the routing localization process is continued and the message is sent to the super-node where is attached the remote node. Note that an update of node's successors will be done overall the ring.
- Nevertheless, if the faulty super-node is the super-node that has the responsibility of RON's key, then its immediate successor handles the message and returns the localization information to the ON. It should be noted that each super-node replicates node's key over its *m* successors as in Chord [15].

2) Timing fault management:

Fault tolerant mechanisms include quite often two phases. The first phase (fault detection) where we detect the failures, and the second phase (fault recovery) where we should react in order to recover the crash.

In order to detect timing failures, we use the "*ping/pong*" technique. In fact, periodically, nodes send each other "ping" messages. Any node that receives a ping message should reply with a pong message. It should be noted that the ON sends ping messages to their attached super-node, and the super-nodes sends ping messages to their own successors. Therefore, any node that does not reply is considered as temporally down and recovery mechanisms are triggered as follows:

- For instance, Figure 3a illustrates the case where the super-node having the id 10 goes down. If the breakdown super-node has the responsibility of ON's key, the ON delegates the localization of the RON to its associated light super-node. Afterwards, the light super-node that manages RON's key returns the localization information of the RON.
- If the breakdown super-node is a successor of the super-node that has the responsibility of ON's key (Figure 3b), then the localization request message is sent to a light super-node which hosts ON's key. In contrast to a physical fault where we send the localization message to available successor, here, we send directly the request to the light super-node that host ON's key. The reasons are twofold: (*i*) this allows us to reduce the routing latency; (*ii*) since the breakdown is short-lived, and the node will become "alive" shortly, we do not want to update node's successor over the ring. If we do that, and the node becomes available, we will have incoherence on node's finger tables. Therefore, the localization message (*i.e.* IP address) of the RON is sent to the ON by the intermediate of the light super-node that hosts RON's key.

- For instance, Figure 3c illustrates a case where the super-node (id 3) which is responsible to RON's key is breakdown. In such case, the last successor, along the routing process (from ON to RON), before the super-node that is breakdown, sends the localization request to the light super-node that hosts RON's key.

IV. EXPERIMENTAL RESULTS

This section presents the experimental validation of HP2P-SIP by using the Oversim simulator [18]. We use also the P2P-SIP architecture which is available in the Oversim simulator as comparison tool.

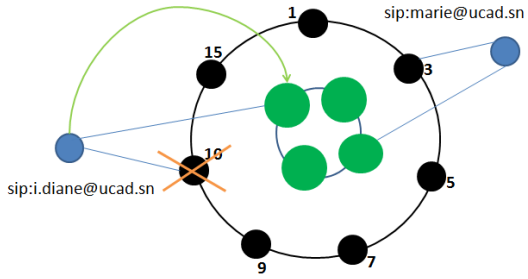

Figure 3a. Timing failure management: case where the attached super-node of an ordinary node is breakdown.

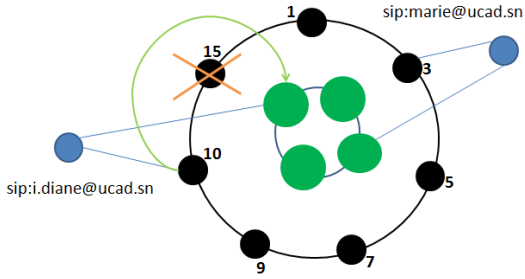

Figure 3b. Timing failure management: case where one of the successor of a given super-node is breakdown.

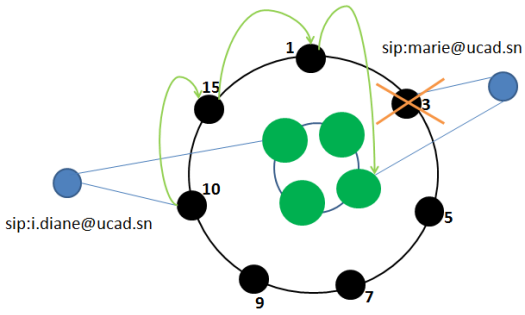

Figure 3c. Timing failure management: case where the remote ordinary node's super-node is breakdown.

In our simulations we consider a fixed number of super-nodes ranging from 100 to 2000 by step of 100. The number of successors for each super-node is equals to m = ln ("number of super-nodes") / ln (2) as stipulate in [15]. Each simulation is run during 1000 seconds. Note that we fix the percentage of super-nodes that can breakdown to 70% with respect to the number of super-nodes. These super-nodes are chosen randomly. It should be noted that the breakdown super-nodes can reappear in the network or leave definitely if a physical failure is simulated. Otherwise, if we simulate timing failures nodes always reappear when a fault happens. The number of light super node is fixed at 10% with respect to the number of super-nodes. The light super nodes are chosen at random among the super-nodes.

For each set of fixed nodes we evaluate the amount of time that we need to localize a given RON. We assume that the light super-nodes are trustworthy. Put simply, they are always available. The case where light super-nodes can breakdown is left for future work.

A. Comparison between HP2P-SIP an P2P-SIP

In this section, we compare HP2P-SIP to the classic P2P-SIP. We use as metric the amount of delay which is necessary to localize a RON.

The results illustrated in Figure 4 show an important gap between the standard P2P-SIP system and our HP2P-SIP approach with respect to the necessary amount of time to localize an ordinary node. In Figure 4 we take into account the timing failures. It should be noted that we observed the same trend for physical faults. By lack of space these figures are not shown.

The elapsed time to localize an RON in P2P-SIP is linear when the number of nodes in the network varies from 100 to 1200 and grows about 1.6 (Figure 4). In contrast to HP2P-SIP, we note a slight raise from 100 to 400 nodes (Figure 4) where the elapsed time grows to about 0.6, but then level off until 1200 nodes. In general, the elapsed time to localize a given RON for HP2P-SIP (resp. P2P-SIP) varies from 0.4 to 0.6 (resp. 1.2 to 1.6 seconds). These results show that HP2P-SIP is less sensitive with respect to the number of nodes in the system. Furthermore, if the number of nodes in the network varies from 1300 to 2000 the delay is almost constant for both architectures. However, around 1300 nodes, we note a slight augmentation. Figure 4 illustrates clearly that HP2P-SIP reduce considerably the routing process in order to localize a given RON.

Figure 5 shows the likelihood that we have to discover an RON for P2P-SIP and HP2P-SIP in presence of timing failures. Note that, this probabily is given by the Oversim simulator [18] for each simulation. When the network is formed by 100 nodes we obtain 90% (resp. 50%) of success to discover a RON for HP2P-SIP (resp. P2P-SIP). Nevertheless, the probabilty of succes decreases when the number of nodes in the network increases.

Note that if the number of nodes augment in the network the number of super-nodes used to localize a RON augment as well. So the possibilty to send a request to a super-node that is breakdown is high. It's the reason why the probabily of success in Figure 5 decreases. Nevertheless, in the case of 2000 nodes, we have 70% (resp. 35%) of chance to localize a RON im HP2P-SIP (resp. P2P-SIP) architecture.

The results illustrated in Figure 4 and 5 show the reliability of HP2P-SIP compared to conventional system like P2P-SIP.

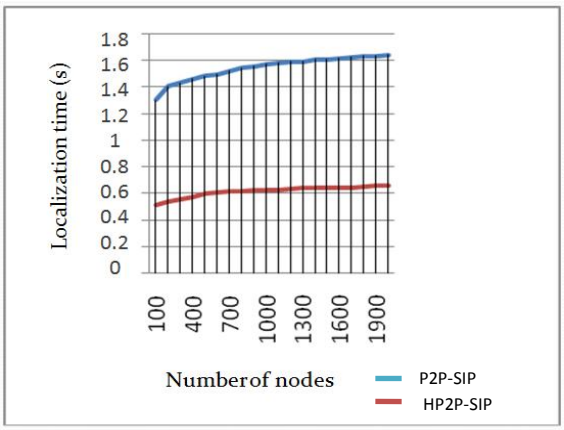

Figure 4. Comparison of localization delay in presence of timing failures between HP2P-SIP et P2P-SIP.

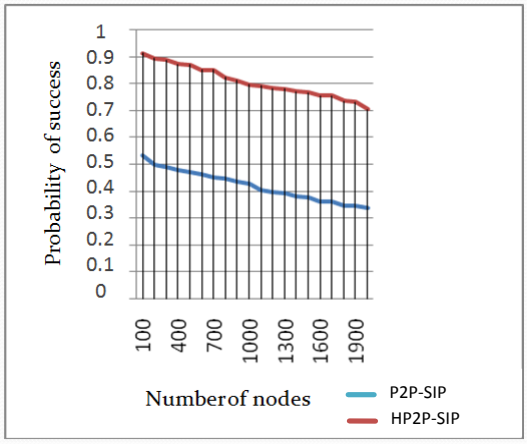

Figure 5. Probability to discover an ordinary node in presence of timing failures.

V. CONCLUSION

We proposed a three-tier approach for fault tolerant management in P2P-SIP networks which combines physical and timing failures management. The light super-nodes, located in the third level of HP2P-SIP architecture, increase the resilience as well the availability of overall network. The obtained results show that HP2P-SIP is able to manage the system during physical and timing failures of super-nodes. HP2P-SIP enables efficiency and robustness, in presence of failures, for P2P-SIP telephony compared to P2P-SIP architectures. The results illustrate that With HP2P-SIP (resp. P2P-SIP) we have at least 70% (resp. 35%) of success to localize a remote ordinary node. We plan as future work to use mathematical models with respect to the chosen super-nodes that should breakdown. We also plan to vary the number of failing nodes as well to take into account the breakdown of light super-nodes.

VI. REFERENCES

[1] Rudiger Schollmeier. A definition of peer-to-peer networking for the classification of peer-to-peer architectures and applications. In Proc. of the 1st IEEE International Conference on Peer-to-Peer Computing, Washington, DC, USA, August 2001.

[2] G.X. Yue, R.F. Li, and Z.D. Zhou, A P2P network model with multi-layer architecture based on region. Journal of Software, vol. 16, no. 6, pp. 1140-1150, June 2005.

[3] Q. Lv, P. Cao, E. Cohen, K. Li, and S. Shenker. Search and Replication in Unstructured Peer-to-Peer Networks. In Proc. of the 16th international conference on Supercomputing, New York, USA, pp. 84-95, 2002.

[4] W. Litwin, M.-A. Neimat, and D. A. Schneider. LH*- a scalable, distributed data structure. ACM Transactions on Database Systems, vol. 21, no. 4, pp. 480-525, 1996.

[5] S. Ratnasamy, P. Francis, M. Handley, R. Karp, and S. Shenker. A scalable content-addressable network, In Proc. of ACM SIGCOMM, San Diego, USA, pp. 161-172, August 2001.

[6] A. Rowstron and P. Druschel, Pastry: Scalable, distributed object location and routing for large-scale peer-to-peer systems, In Proc. of IFIP/ACM International Conference on Distributed Systems Platforms (Middleware), Heidelberg, Germany, pp. 329-350, November 2001.

[7] Skype Web site. http://www.**skype**.com.

[8] Bryan, D., B. Lowekamp, and C. Jennings. SOSIMPLE: A Serverless, Standard-based, P2P SIP Communication System. In Proc of International Workshop on Advanced Architectures and Algorithms for Internet Delivery and Applications, pp. 42-49, June 2005.

[9] K. Singh and H. Schulzrinne. Peer-to-Peer Internet Telephony using SIP. In Proc. of NOSSDAV, Stevenson, Washington, USA, pp. 63-68, 2005.

[10] Laban Mwansa, Jan Janeček. Ensuring fault-tolerance in generic network location service. In Proc. of on 22nd European conference on Modelling and simulation, Brussels, Belgium, pp. 254-260, 2008.

[11] B. Yang and H. Garcia-Molina. Designing a Super-Peer Network. In Proc. of 19th International Conference on Data Engineering, Bangalore, India, pp. 49-62, March 2003.

[12] Yingwu Zhu, Honghao Wang and Yiming Hu. A Super-Peer Based Lookup in Structured Peer-to-Peer Systems. In Proc. of the 16th International Conference on Parallel and Distributed Computing Systems, pp. 465-470, 2003.

[13] K. Singh and H. Schulzrinne. Using an External DHT as a SIP Location Service. . Columbia University Technical Report CUCS00706, New York, NY, February 2006.

[14] Singh, K., and H. Schulzrinne, "Peer-to-Peer Internet Technology Using SIP," Columbia University Technical Report CUCS-044-04, New York, October 2004.

[15] Ion Stoica, Robert Morris, David Karger, M. Frans Kaashoek, and Hari Balakrishnan, Chord: A scalable peer-to-peer lookup protocol for internet applications. In Prof of SIGCOMM, pp. 149-160, San Diego, California, USA, 2001.

[16] Jenn-Wei Lin, Ming-Feng Yang, and Jichiang Tsai. Fault Tolerance for Super-peers of P2P Systems. In Proc. of the 13th Pacific Rim International Symposium on Dependable Computing, Washington, DC, USA, Melbourne, Victoria, Australia, December 2007.

[17] J. Rosenberg et al., "SIP: Session Initiation Protocol," IETF RFC 3261, June 2002.

[18] Ingmar Baumgart, Bernhard Heep, and Stephan Krause. OverSim: A Flexible Overlay Network Simulation Framework. In IEEE Global Internet Symposium, pp. 79-94, 2007.