An Intelligent Technique for Implementing a New Platform for Merging Some of Protocols Package Using P2P Delayed Networks

Hatim Ghazi Zaini , Mohamed T. Faheem

Abstract — Recently, exploring effective ways for distributed resources that allow sharing large amounts of data among users are presenting new challenges to computer networks. The Peer-to-Peer (P2P) networks have emerged over the past several years as new and effective ways for distributed resources to communicate and cooperate security and privacy in a number of ways. This paper focused on a new platform for merging some of Protocols package for P2P networks based on an inelegant technique. The Merging Protocol based on Fuzzy Decision for Trust Evaluation raised by P2P networks have been introduced to help in studying some protocols to support efficiently of the transferred encrypted data. The efficiency of these protocols such as used in routing information raised the need to merge some of them to gain both the high speed and security of the significant valuable transmitted data. In order to reach this goal, we have utilized a dynamical model of trusted P2P transactions with Fuzzy reputation aggregation with delays transmission which is considered as an intelligent learning machine based on Fuzzy Logic technique in the design process of merging protocol of P2P network. To support the usefulness of this technique, a discrete model that showed an improvement of a distinctive protocol that combines more than one protocol in one side to achieve common standard behaviors has been constructed which have led to improve in the efficiency of exchange of data across P2P networks.

Index Terms — Merging protocols, P2P delayed networks, time delays factors, NS-2 simulation package

1 INTRODUCTION

Since a few years and due to a lack of standards and toolkits, early peer-to-peer(P2P) applications like KaZaA, Napster, and SETI have come into attention in the literature and demonstrated the true power of the Internet where millions of information stores in common PCs and sitting idle on desks around the world[1-7]. It is well known that the P2P technology has the ability to establish virtual overlay networks where there is no central authority or infrastructure that could coordinate the behaviours of the peers. We note that, a peer can act both as a server and a client since it can provide services to other peers as well as request services from other peers. The functionality of P2P networks is structured in two phases. In the first phase, a host is allowed to find other P2P hosts and connect to the network, while in the second phase, this connected host is enabled to search for files by broadcasting queries and test them for reputation based on some security rules to allow them to be downloaded. Any peer can arbitrarily join or leave the network at any time and each peer itself is responsible for making local autonomous decisions based on information received from other peers in the network[8-16]. The P2P technologies exploit the CPUs and storage devices of these PCs to produce and exchange huge data stores, communications systems, and processing engines. Therefore, an open P2P network is highly dynamic and autonomous. The protocols that are used with this overlay network designed for resource sharing across the global Internet. Recently, [7], the P2P network is being treated as a network that is used to present a new service and function that are built completely at the application layer where its nodes interact via client programs running on their local machines irrespective of the underlying physical network. The resource searching, connectivity, routing, and other real applications are handled in a complete distributed way where every node nominally equal to

- *Hatim Ghazi Zaini is currently working as the head of EE Dept. at College of Engineering, Taif University, (KSA). Emails*: rhgz@hotmail.com

•*Mohamed T. Faheem is currently working in EE Dpt. at College of Engineering, Taif University, (KSA) & on leave from College of Engineering, Dept of Computers Eng.& Auto. Cont.l, Tanta University, Egypt. Emails:* m_t_faheem@yahoo.com.

every other recent client. The P2P is not more than just the universal file-sharing model (such as Napster…etc), but it is alsohas a complete self organizing which requires no need for central instances to manage the network. Accordingly, the computation that occurred among P2Pworking groups, business applications fall into a number of scenarios and more details can be found in [5]. It should also be noted that, since peers are heterogeneous in their natural construction specially in both network and system capacity, then all other peers can be subsided with all their needs through the transactions that take place among themselves.

The most distinct characteristic of P2Pcomputing is its symmetric communication between the peers where each peer has both a client and a server role. The most advantages of the P2Psystems are their construction as a multi-dimensional layers. This construction helped in improving some propertied of the P2P network such as:

1. Scalability by enabling direct real-time sharing of services and information.
2. Enabling knowledge sharing by aggregating information and resources from nodes that are located on geographically networks which enable networked hosts to share resources in a distributed manner.

Another an important issue in such P2P networks is their ability to efficiently search the contents of the other peers successfully, where most existing search techniques are based on either the idea of flooding the network with queries or with some form of global knowledge[3].

In recent years, most common definitions are given throughout researches where often certain terminology have distributed entities and are not identified in all systems in the same way[1],[8],[14-16]. The definitions presented here for some terminology do represent what are given in most of references, however, often number of terms that are used to define a device or capability on a P2P network, e.g., node, resource, peer, agent, service, server, etc will be given in a short sentences as much as it could be. In this section, common definitions are given which are used throughout this research. Therefore, wherever appropriate, the terminology provided here is given within the context of the system they are described within and some of these terms [3],[6].

2 Review of efficient and distributed merging protocols

Based on the preceding section, It is our intension to review some of our new efficient and distributed merging protocols introduced in [19]. The main idea behind these protocols were to search correct fingers along the ring without establishing or reconstructing them. To accomplish these merging protocols, we have assumed that the three nodes systems namely, q with dimension n, q_1 with dimension n_1, and q_2 with dimension n_2 were used with the same size of identifier space, where q_1 nodes chosen uniformly at random from q. According to these assumptions and without loss of generality our merging protocol in [19] stated that when the number of peers become below the lower bound of the network, then the present system will be consisted of merging q_2 with q_1 to generate a new system denoted by q and is referred to as the next generated system. That is to say when q_2 merged into q_1 we ended up with a new merging system called q. Finally, the following notation has been used in our work in [19] as they are needed in our study.

1-- finger table denotes y as an active node that maintains a finger table for routing called a finger table.

2-- The i-th entry in the finger table contains y node that succeeds active node by at least $2i-1$ in the identifier ring where the i-th is entry element of a finger table.

3—The successor of active node y denotes the first node that immediately succeeds y in the p2p network.

4—A predecessor of node y represents the node that immediately precedes y in the the ring.

5-- nextsys(x) is the name deserved for the next system that node y belongs to.

3 Merging Protocol based on Fuzzy Decision for Trust Evaluation

To proceed in obtaining a dynamical trusted P2P transactions with fuzzy reputation aggregation, we need to introduced some of our work in [17-19] for merging protocol including delays factors based on Fuzzy decision to be used in trust evaluation. We note that the assumption of active nodes in q_2 joined in with q_1 was introduced to help in obtaining the required protocol without any loss of generality. Our work in[19] regarding the merging protocol was constructed by making each node in P_2 calls Pseudocode join( ) and Pseudocode search merging finger(i), while each node in q_1 just invokes search merging finger(i). These join( ) and search merging finger(i) codes introduced put some modification to compensate for the delay effect due to sleeping nodes. The following algorithms that include the effect of delayed data communication can be summarized as follows:

i. Pseudocode join( ) code algorithm for active peers with delayed time factor due to searching
.

```
1. active y.join(y)
2. asuccessor = y_.find _asuccessor(y) with delayed time factor
```

ii. Pseudocode for active y.find_asuccessor(z) with delayed time factor

```
1. active y.find_asuccessor(z)
2. if (z ε (y, asuccessor)
3. return asuccessor ;
4. else r = closet _preceding node(z);
5. return r.find_asuccessor(z);
```

iii. Pseudocode for active search_amerging _finger(i)

```
1. y. active search amerging finger(i)_delayed
2. z = finger[i].apredecessor;
2. while ( z < y + 2i−1)
3. z = z.apredecessor;
4. returnz.asucessor;
```

As we have shown in our work in [19], the Pseudocode for active peers introduced above has been used in the design process of merging protocols where its output decision was so important and required no additional construction of all fingers for the next P2P systems. This Pseudocode has led also to save some costs as compared with those covered by other protocols [5]

It should be also remembered as shown in our work in [ ] that the results of merging protocol that improved the performance of data access in p2p network has based on using a measure for evaluating the performance of data access in P2P network due to merging process by the ratio n_1/ n_2 as

$$r = n_1/ n_2$$

where the evaluation of normal (Gaussian) distribution for number of hops required in merging some of protocol packages in the P2P process has been constructed using the relation

$$\eta = \sigma e^{-\theta t} \quad , \text{ with } \quad \theta = r^{2,} \text{ and } \sigma = 1/(\xi\sqrt{2\pi}), \qquad (1)$$

where t denotes the number of required hops in the merging protocol for active peers, ξ is a weighing factor chosen as 2.5, 1.5, 1.25 for r_1, r_2, and r_3, respectively. This probability density function η given in Eq. (1) was used to plot the normal (Gaussian) distribution against the number of hops required for the execution of Pseudocode active search_ amerging _finger(i). The resulted plot showed that maximum number of hops for the merged system requires no more than 10 hops for different cases of r i.e. r1 =0.8, r2= 0.6, and r3=0.5 in the experiments as shown in Fig.10 and sharing almost 2 hops for $\eta = 0.15$

4 The implementation of a New Inelegant Technique for Improving Data Access Through Merging Some of Protocols package using P2P Networks

Based on the above review sections, we are utilizing an inelegant technique of merging Protocols package to give us a new concept of secure routing stack to be implemented in Linux operating system with NS-2 discrete simulator as well as the AgentJ [13] which is used to get a simplified algorithm to give a pure real P2P environment system. To complete our proposed work to implement this Inelegant Merging Protocols package, we augment another optimal algorithm named AntNet [27],[30] in our process to determine some optimal paths for improving data access of the real P2P environment. After completing building of this intelligent system , we ended up with a new modified system that uses routing protocol based on reputation based on Fuzzy decision which takes a delay time reputation values of the nodes into the computation processes. The augmentation of these new factors in the process gave a new platform for merging some of Protocols package for P2P Networks that can be shown in the following subsection.

4.1 Simulation Results of a new platform for merging some of Protocols package for P2P Networks

To gauge the behavior of the proposed routing algorithm for P2P network, the construction of the network has been implemented using a directed graph as shown in Fig. 1. with N nodes. All the links in the network are considered bidirectional with both the transmission capacity and the transmission delay being augmented in the design process. In this platform each node is treated as a double agent either a communication end-point (host) or a forwarding unit (router) and these node in the network is assumed to maintain an input buffer composed of a single queue and an output buffer composed of a high priority queue and a low priority queue for each neighbor or outgoing link where the high priority queue is served first before handling the low priority queue. Both data packets and mobile agents that taken place along the network are specified in the regular sense as in the literature [2]. Next, we showed the reputation trusted results that are obtained , where the data is delivered from source to destination with its arguments includes the open port and deliver functions. Consequently, the open port function is used first in creating a connection between the invoking host and server application as shown in Fig.5. and returns the port's name that allows this connection being implemented through it.

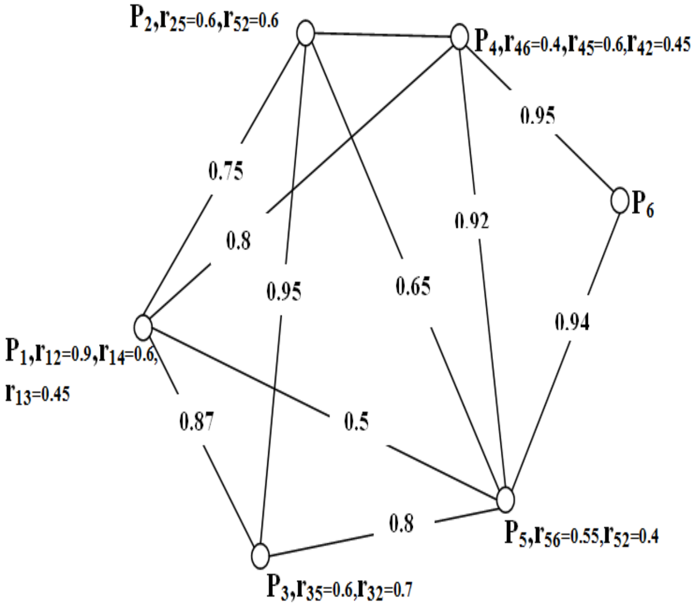
$P_2, r_{25}=0.6, r_{52}=0.6$
$P_4, r_{46}=0.4, r_{45}=0.6, r_{42}=0.45$
0.95
0.75
P_6
0.8
0.92
0.95
0.65
0.94
$P_1, r_{12}=0.9, r_{14}=0.6,$
$r_{13}=0.45$
0.87
0.5
0.8
$P_5, r_{56}=0.55, r_{52}=0.4$
$P_3, r_{35}=0.6, r_{32}=0.7$

Fig.1. The concept of GFTG, LFTG and PFTP

Secondly, the delivered function injects the data though a specific port that converts the data to a string and structurally marshals the data before sending it across the connection where the host application listens to the specified port through this connection and then prints any values received on the port. The simulation model based on discrete program where its argument uses open port as an input while its output functioning as deliver functions. The function of the open port is designed to create a connection between the invoking host and the server application during the successful transaction process and define the required port's name for this connection. On the other hand, the deliver function injects the data through the specific port converting it to a string and then structurally guides the carrying data to the assigned destination before sending them across the connection. That is to say, the host that contains an application received a query from a specified port through the connection, and then prints out all values received through this port.

5 Dynamical Trusted P2P Transactions with Fuzzy Reputation Aggregation

To evaluate the Global Reputation Aggregation of peers connected through P2P system , a simple formula reported in [8] can be used which is defined as:

$$q_j = \sum_{j \varepsilon H} u_i \, r_{ij} \Big/ \sum_{i \varepsilon H} u_i \qquad (2)$$

where q_j is the global reputation of peer i, H is the set of peers that are connected in P2P and has transactions with peer j, r_{ij} is the local trust score of peer j rated by peer i, and u_i is the aggregation weight of r_{ij}. We also note that the dynamical process characterized by the general time delay system define in our work in [19] is used to obtain a stable global reputation for all peers in the P2P net.

For the sake of comparison , we take similar topology as noted in [2]. Fig.1. shows six connected peers where the edge's value between peers denotes comprehensive evaluation value with each peer keeps LFTG within itself. Let us, for instance, consider a PFTP as noted in [2] so the system at P_2 queries for peer1 and peer3, while Peers3 and peer4 aren't qualified and consequently, aren't queried In this case , we have two options to choose PFTP. This two options can be seen using the well known rule that is reported in [8]. This rule defined each peer to maintain two records: one for a transaction record that takes place through the P2P net the global aggregation weights through the Fuzzy inference system that can be detected. The second one reserved for a local score table that maintains the remote peers' evaluated trust scores. To apply this rule for the topology shown in Fig.1., It is an easy task [2],[8] to compute global reputation aggregation based on this rule where each peer queries the trust scores from remote peers. That means, the system partially queries qualified peers that meet an aggregation threshold to be evaluated based on formula given in Eq.(2) . To see how to use formula in Eq.(2) to compute Peer 2's global reputation as shown in Fig.1. , We have

$$q_2 = \frac{u_1 \times r_{12}}{u_1 + u_3} + \frac{u_{13} \times r_{32}}{u_1 + u_3}$$

$$q_2 = \frac{0.75 \times 0.9}{0.75 + 0.8} + \frac{0.8 \times 0.7}{0.75 + 0.8} = 0.7968$$

Since it is assumed that the system set a high threshold for the Peer 2 = 0.7, so by applying Eq.(2), the system at node 2 queries the path from peer1 and 3, while Peers 3 and 4 aren't qualified and consequently, aren't queried thus reducing the bandwidth consumption. That is to say, Peers 1 and 3 are qualified to be queried, whereas Peers 4 and 5 aren't qualified which means Peers 1 and 3 transmit their local scores r_{12} and r_{32} to Peer 2 while Peers 3 and 4 aren't qualified and thus aren't queried. In this case, the system queries only a few peers which leads to reduce the bandwidth consumption. To see this, let the system sets Peer 2 in Fig.1. with an aggregation threshold of 0.7., then it is clear from Fig.1. that global reputation aggregation of p_2 is aggregated from two local trust scores $r_{1,2}$ and $r_{3,2}$ that are generated by Peer1 and Peer3, respectively while Peer4 and Peer5 are disqualified from participating in the evaluation process because the global reputation aggregation evaluation from 1 and 3 is equal 0.7968 while its counterpart from 4 and 5 is 0.4870. Then the system will restore the global reputation aggregation for peer 2 to become equal 0.7968. Similarly, all the rest of the peers in Fig.1. can be easily evaluated using Eq.(2). On the other hand as in Fig.1., if peer6 is acted as a server that receives a query from a specified port for connection and prints any values received through this port, then the delivered application is sent to this specified port through a secure routing path that is assigned through the global reputation aggregation protocol.. The global reputation aggregation information shown in table-6 is concluded from Fig.1. and table-1 summarizes the trust score and weight values to be used to get the global reputation aggregation of peer 6 in this application.

Table-1 The aggregation weight and the trust score of peer 2 as generated from Fig.1

Node	The aggregation weight of $u_{i,}$ i = (1, 3 ,4, 5)	r_{i2} : the local trust score of peer 2 rated by peer i =(1, 3 ,4, 5)	Trust Score
p_1	0.75	r_{12}	0.9
p_3	0.8	r_{32}	0.7
p_4	0.5	r_{42}	0.45
p_5	0.65	r_{52}	0.4

Table-2 Possible routing path from peer1 to peer 6

Routing Paths	Trusted Reputation
1-2-4-6	0.5818
1-4-6	0.4914
1-3-5-6	0.5320
1-2-5-6	0.6761
1-2-3-5-6	0.8059
1-5-6	0.5326
1-4-5-6	0.5823

From table-2, one can easily see that although both routing paths 1-4-6, 1-5-6, are shown to have the shortest path in the network which is consider to be of great advantage in the design process but they still have a less secure path than other paths especially the prominent path 1-2-3-5-6 with a secured reputation 0.8059. We note also that, the Unicast function described in the above lines is enforced dynamically by the interpreter and is used to send data from peer 1 to peer 6 in a network as shown in Fig.1. In this application, peer 6 acts as server which reply to the specified port that needs query and prints any values received through this port.

13. CONCLUSION

This paper introduced the problem of merging protocols based on Fuzzy decision for trust evaluation and obtained for both active and sleeping peers. It is shown that both of them are treated using some innovative technique that has led to improve protocols performance which has shown to be faster than previous protocols in terms of the number of hops . It should be mentioned also that, some experiments have been studied and implemented to support these claims. Based on these dynamical models, the development of a platform that would be suitable for its intended application in the P2P network has been implemented. The qualitative performance analysis was easily studied using this dynamical model for the implementation of a new inelegant routing algorithm that has improved data access through merging some of protocol packages using P2P networks. These packages have been achieved based on the NS-2 simulation package which is considered as the best popular simulation package. Using this package has helped us in obtaining the best secured reputation shortest path including the effect of transmission delays, where the performance of transactions among P2P nodes have been greatly improved. This paper demonstrated also that the use of individual computational intelligence techniques has helped us in deriving an intelligent technique with Fuzzy trust aggregation in the right way towards real P2P networks. Based on the obtained results from table1 and table2 indicated that the performance of the dynamical models of P2P network is much better compared to those individual models reported in [2] and [8] in terms of the P2P protocol accuracy.

REFERENCES

[1] Yu-Kwong R. K., Peer - to- Peer Computing Applications, Architecture, Protocols, and Challenges, a Chapman & Hall Book, Taylor & Francis Group, LLC, 2012

[2] C.Gui, et al, “ Towards Trustworthy Resource Selection: A Fuzzy Reputation Aggregation Approach”, B. Xiao et al. (Eds.): ATC 2007, LNCS 4610, pp. 239–248, c_Sprin.-Verlag Berlin Heidelberg 2007.

[3] Xu Xiang, “Merging and Splitting Structured Peer-to-Peer Systems,” International Symposium on Information Science and Engineering, 2008.

[4] Bong-Jun Ko, “ Distributed Self-Stabilizing Placement of Replicated Resources in Emerging Networks”, IEEE/ACM Trans. on network., Vol. 13, no. 3, June 2005

[5] T. T. DINH, ‘ Trustworthy Peer-to-Peer Infrastructure Using Hardware Based Security’, Ph.D School of Computer Science, College of Engineering and Physical Sciences, Univ. of Birmingham, Sept. 2010.

[6] S. Song, K.Hwang , et al,” Trusted P2P Transactions with Fuzzy Reputation Aggregation”, IEEE Computer Society, Internet Computing, pp 1089-7801, NOV. –DEC. 2005.

[7] B. Chord et al ,” a scalable peer-to-peer lookup service for Internet applications”, In Proc. of SIGCOMM, pp. 149-160, August 2001.

[8] Chunqi Tian, et al,”A New Trust Framework Based on Reputation for Unstructured P2P Networks”, IEEE, 2008. Applications (0975 – 8887) Volume 37– No.12, Jan. 2012.

[9] Wei Wang , et al , “ Ant-based Reputation Evidence Distribution in P2P Networks”,Proc. of the Fifth Inter.Conf. on Grid and Coop. Comp. (GCC'06), IEEE, 2006.

[10] I. Lekhal, et al, “ Intelligent framework for Resource Discovery in peer to peer systems “, IEEE 2012

[11] L. CAO, “ Intelligent Security System Based on Self-Organizing Wireless Sensor Networks,” the automation college of Beijing Union University, Beijing, . China.

[12] Ming-Chang Huang, et al, “Introduction to Two P2P Network Based Reputation Systems,” International Symposium on Biometrics and Security Technologies, 2012.

[13] Ian Taylor, et al,” AgentJ: Enabling Java NS-2 Simulations for Large Scale Distributed Multimedia Applications”, email: Ian.J.Taylor@cs.cardiff.ac.uk

[14] Wei Wang,et al, “Ant-based Reputation Evidence Distribution in P2P Networks”, Proc.of the Fifth Intern. Conf. on Grid and Cooperative Computing (GCC'06), 2006.

[15] Ian Taylor, AgentJ " Java Network Simulations in NS-2 An Installation and User Manual", A PROTEAN Research Group Project, Naval Research Laboratory, Code 5522 ,March, 2010.

[16] K. M. Sim and W. H. Sun, " Ant Colony Optimization for Routing and Load-Balancing: Survey and New Directions," IEEE Trans. Sys, Man, and Cybernetics-Part A: sys. and Humans, vol. 33, no. 5, Sept. 2003, pp. 560-572.

[17] Saidahmed, M. T. F., "A New Approach for Designing a feedback Controller for a Wind Tunnel Model Involving a Delay", Proc. of the 35th Mid. Symp., GWU, IEEE Circuits and Systems Society, Aug. 9-12, 1992.

[18] Saidahmed, M.T.F.,"A new approach for designing a feedback controller of a wind tunnel model involving a delay"; Proc.of the 35th Mid. Symp., Aug. 9-12, 1992.

[19] M. M. Al-Harthi, M. T. F Saidahmed, A A Al-Ahmady," An Intelligent Technique for Improving Data Access Based on Merging Some of Protocols Package Using P2P Delayed Networks," to be published in International Journal of Scientific & Technology Research, VOL 2, ISSUE 2, 2013