# Design of Very Efficient Lookup Algorithms for a Low Diameter Hierarchical Structured Peer-to-Peer Network


Bidyut Gupta
*Department of Computer Science*
*Southern Illinois University*
Carbondale, IL 62901, USA
bidyut@cs.siu.edu

Nick Rahimi
*Department of Computer Science*
*Southern Illinois University*
Carbondale, IL 62901, USA
nick@cs.siu.edu

Henry Hexmoor
*Department of Computer Science*
*Southern Illinois University*
Carbondale, IL 62901, USA
hexmoor@cs.siu.edu

Shahram Rahimi
*Department of Computer Science*
*Southern Illinois University*
Carbondale, IL 62901, USA
rahimi@cs.siu.edu

Koushik Maddali
*Department of Computer Science*
*Southern Illinois University*
Carbondale, IL 62901, USA
*koushik@siu.edu*

Gongzhu Hu
Department of Computer Science
Central Michigan University
Mount Pleasant, MI 48859, USA
(hu1g@cmich.edu)



***Abstract*— In this work, we have considered a new hierarchical non-DHT based architecture for Peer-to-Peer (P2P) networks in which at each level of the hierarchy existing networks are all structured and each such network has the diameter of 1 overlay hop. Such low diameters have immense importance in designing very efficient data lookup algorithms. The mathematical model based on the Chinese Remainder Theorem has been used to define the neighborhood relations among peers to obtain the above-mentioned diameters. In this work, we present very efficient intra-group as well as inter-group data look up algorithms with *O(1)* time complexity exploiting the above mentioned low diameter features of the P2P system. Besides, we present very efficient algorithms related to peers joining with new and existing resource types.**



***Keywords: Structured P2P networks, Chinese Remainder Theorem, Network Diameter, Data Lookup***


## 1 INTRODUCTION

Peer-to-Peer (P2P) overlay networks are prevalent in distributed systems. Unstructured and structured constitute two classes of P2P networks. In unstructured systems [1], [2] peers are organized into arbitrary topology. Flooding is commonly used for data look up. *Churn* is the problem where peers frequently join and leave the system. *Churn* is handled effectively in unstructured systems. However, it compromises with the efficiency of data query and the coveted flexibility. On the other hand, properly designed structured architectures can offer efficient, flexible, and robust service [3] - [5], [7], [8]. Such overlay networks use distributed hash tables (DHTs) to achieve efficient data insertion, lookup etc. However, maintaining DHTs is a complex task and it needs huge effort to manage the problem of churn. Therefore, the major challenge for architecture design is easing churn management while still providing an efficient data query service. There are notable approaches that have considered hybrid systems [6] with the aim of incorporating the advantages of both structured and unstructured architectures. However, these works have their own pros and cons.

### Our Contribution

In this work, we have considered interest-based P2P systems [6], [9], [10], [11]. We have conceived a non-DHT based hierarchical P2P architecture in which each level of the network hierarchy is structured and diameter of each network is 1 overlay hop. Preliminary findings have been reported in [12]. To the best of our knowledge, it is the first time that a successful attempt has been made to design structured hierarchical P2P networks with its entire constituent subnetworks possessing the diameter of 1 overlay hop only. Note that low diameters play significant role in designing very efficient data lookup algorithms. The proposed architecture uses a mathematical model based on the Chinese Remainder Theorem (CRT) to define the neighborhood relations among the peers to obtain small diameters.

In this work, we present very efficient intra-group as well as inter-group data look up algorithms with $O(1)$ time complexity exploiting the above mentioned low diameter features of the P2P system. In addition, we present algorithms related to peers joining with new and existing resource types.

The paper is organized as follows. In Section II, we state in detail the proposed architecture and the mathematical foundation used in the design phase. In Section III, we present the data lookup algorithms.

In addition, an analytical comparison with some notable works has been presented. In Section IV, we present the algorithms for peer joining.

## II. PROPOSED ARCHITECTURE

In this section, we present a structured architecture for interest-based peer-to-peer system [6], [10], [11], [14] and the required mathematical basis supporting the architecture. As mentioned earlier that some related preliminary findings have been reported in [12]. We use the following notations along with their interpretations while we define the architecture.

We define a resource as a tuple $<R_i, V>$, where $R_i$ denotes the type of a resource and V is the value of the resource. A resource can have many values. For example, let $R_i$ denote the resource type 'songs' and V' denote a particular singer. Thus $<R_i, V'>$ represents songs (some or all) sung by a particular singer V'. In the proposed model for interest-based P2P systems, we assume that no two peers with the same resource type $R_i$ can have the same tuple; that is, two peers with the same resource type $R_i$ must have tuples $<R_i, V'>$ and $<R_i, V''>$ such that $V' \neq V''$.

We define the following. Let S be the set of all peers in a peer-to-peer system. Then $S = \{P^{Ri}\}$, $0 \leq i \leq r-1$. Here $P^{Ri}$ denotes the subset consisting of all peers with the same resource type $R_i$ and no two peers in $P^{Ri}$ have the same value for $R_i$ and the number of distinct resource types present in the system is r. Also for each subset $P^{Ri}$, $P_i$ is the first peer among the peers in $P^{Ri}$ to join the system. However, this constraint can easily be relaxed.

We now propose the following architecture suitable for interest-based peer-to-peer system. We assume that no peer can have more than one resource type.

### A. Two Level Hierarchy

We propose a two level overlay architecture and at each level, structured networks of peers exist. It is explained in detail below.

At level 1, we have a network of peers such that peers are directly connected (logically) to each other. In graph theoretic term, the network at level 1 is a complete graph. Hence, the network diameter is 1 overlay hop. The periphery of this network appears as a ring network and we name it as transit ring network. This network consists of the peers $P_i$ ($0 \leq i \leq r-1$).

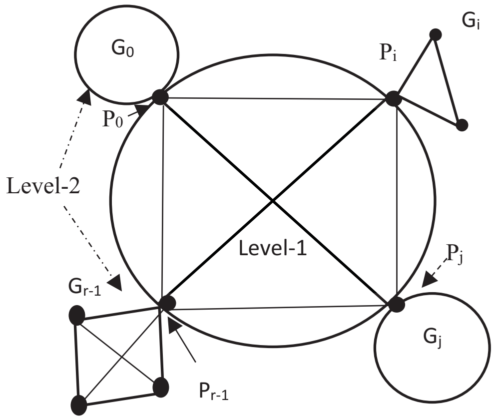


Fig. 1 A two-level structured P2P architecture with r distinct resource types

Therefore, number of peers on the ring is r and we have assumed that this number represents the number of distinct resource types of the P2P system. Each of these r peers will be termed as a group head. The periphery of this network as well as the direct links connecting any two peers in this network can be used for efficient data lookup.

At level-2, there are r numbers of completely connected networks of peers. Each such network, say $N_i$ is formed by the peers of the subset $P^{Ri}$, ($0 \leq i \leq r-1$), such that all peers ($\epsilon$ $P^{Ri}$) are directly connected (logically) to each other, resulting in the network diameter of 1 overlay hop.

Each such $N_i$ is connected to the transit ring network via the peer $P_i$, the group-head of network $N_i$.From now on network $N_i$ will be called as group$_i$ (in short as $G_i$) with $P_i$ as its group-head. Sometimes $N_i$ will be referred to as the i$^{th}$ cluster as well. The proposed architecture is shown in Fig. 1.

In any structured P2P system, the mathematical model used to build the architecture defines neighborhood relations among peers. The mathematical model is intimately related to the efficiency of different data lookup schemes used in a given structured P2P system. We now state a brief sketch of the mathematical model used in our approach to realize the architecture. We shall determine a simultaneous solution (a positive integer) of a given system of linear congruencies and then determine some more solutions as needed to form the architecture, which are congruent to the simultaneous solution. For this, we shall take help of the Chinese Remainder Theorem (CRT). Each such solution will become the logical address of a group head uniquely. At the same time, we will determine separately the solutions of each linear congruence as needed and these solutions will represent the logical addresses of the peers present in a group. The following interesting structural facts will be revealed.

(a) The neighborhood relationships among the group heads based on the logical addresses assigned to them will be shown to form the network at level 1 with diameter 1 overlay hop. In graph theoretic term, each $G_i$ is a complete graph.
(b) Assignment of logical addresses to peers in a subnet $P^{Ri}$ (i.e. group $G_i$) based on the solutions of an individual linear congruence used in the CRT will guarantee that all peers in $G_i$ are directly connected to each other (logically) forming an overlay network of diameter 1.
(c) Assignment of logical code to a distinct resource type will be the same as the corresponding group head's logical address. That is, the resource $R_i$ will have the code that is identical to the logical address of the group head $P_i$ of group $G_i$.

Below we give a short overview of Chinese Remainder Theorem.

*B. Chinese Remainder Theorem*

Chinese Remainder Theorem (CRT) states the following:
Suppose $m_0, m_1, m_2\ldots, m_{r-1}$ are r integers such that no two of which have a common factor other than 1.
Let $M = m_0\, m_1\, m_2 \ldots m_{r-1}$. Also suppose that $a_0, a_1, a_2, a_{r-1}$ are integers such that gcd $(a_i, m_i) = 1$ for each i.
Then the r congruencies given as
$a_i x \equiv b_i \pmod{m_i}$, $0 \le i \le r-1$ have a simultaneous solution X that is unique modulo M and is given as
$X = c_o\, n_1\, n_1^{\sim} + c_2\, n_2\, n_2^{\sim} + \ldots + c_{r-1}\, n_{r-1}\, n_{r-1}^{\sim}$
where $n_i = M/m_i$ and gcd $(n_i, m_i) = 1$; that is, $n_i\, n_i^{\sim} \equiv 1 \pmod{m_i}$; and $c_i$ is the least positive integer such that
$a_i c_i \equiv b_i \pmod{m_i}$, for $0 \le i \le r-1$
Also, note that each congruence, $a_i x \equiv b_i \pmod{m_i}$ has a solution $x_i$, which is unique modulo $m_i$ because any solution $x_i'$ of it is congruent to $x_i$ modulo $m_i$. The reason for this is that gcd $(a_i, m_i) = 1$ for each i.

*An example*:
$x \equiv 2 \pmod 3$ (1)
$x \equiv 3 \pmod 5$ (2)
$x \equiv 2 \pmod 7$ (3)

These three congruencies satisfy all restrictions of CRT.
We see that $a_1 = a_2 = a_3 = 1$ and $c_1 = 2$, $c_2 = 3$, $c_3 = 2$; $M = 105$; also $n_1^{\sim} = 2$, $n_2^{\sim} = 1$, $n_3^{\sim} = 1$
Therefore, the simultaneous solution satisfying all these three congruencies is 23 (the least positive solution) and all solutions are congruent to 23 (mod 105). That is, all solutions of the form 23+k.105 (k is an integer) are mutually congruent as well since '<u>congruence</u>' is an '<u>equivalence</u>' relation.

*C. Construction of the Architecture*

Let us consider the construction of the architecture for r distinct resource types. We consider r linear congruencies,
$a_i x \equiv b_i \pmod{m_i}$, $0 \le i \le r-1$, which satisfy the CRT's requirements as mentioned earlier. Each such congruence is also known as linear Diophantine equation (LDE) and note that each congruence, $a_i x \equiv b_i \pmod{m_i}$ has a solution $x_i$, which is unique modulo $m_i$. As mentioned earlier, the P2P architecture will be a 2-level structured one such that at each level, network(s) will have a diameter of one overlay hop. It will greatly enhance the speed of data lookup algorithms.

To start with, we first construct the network at Level-1, so we need to define the neighborhood relations among the peers at this level, which in effect will create the necessary logical links among the peers. These neighborhood relations will be defined via the logical address assignments and for this purpose, we use the CRT as the mathematical base. We determine a simultaneous solution X of the r different LDEs; the simultaneous solution X is unique modulo M; M is the product of all r moduli used in the r LDEs. The address assignment goes as follows.

*<u>Level-1 address assignment and neighborhood relations</u>*: Suppose that $P_0$ is the first peer to join the system with resource type $R_0$. Therefore, $P_0$ is considered as the group head of group $G_0$ that has currently only one peer, namely, $P_0$. We assign the logical address X (the simultaneous solution X). Later when other peers with the same resource type join, they are placed in the group $G_0$. We will explain a little later how to assign their logical addresses and how they are linked. Next, suppose that peer $P_1$ is the first to join the system with resource type $R_1$ among all other peers that have the same resource type.

Therefore, $P_1$ becomes the group head of group $G_1$. It will get the logical address (X + M). Observe that the solutions X and (X + M) are mutually congruent. The process of assigning logical addresses to group heads implies that group head $P_i$ with resource type $R_i$ joins the system before the group head $P_{i+1}$ with resource type $R_{i+1}$. Hence, for assigning logical addresses to group heads, the sequence of their arrivals is used; therefore, in general, the logical address of a group head $P_i$ becomes (X + iM). Observe that all these addresses are mutually congruent. Besides, the resource type $R_i$ of $P_i$ gets the code [X + iM]. In this architecture, we use the relation 'congruence' to create links between any two peers. That is, two peers $P_i$ and $P_j$ are linked together if their logical addresses are mutually congruent. Therefore, because of the 'equivalence' property of 'congruence' relation, all logical addresses are mutually congruent. Hence, we link each $P_i$ directly with any other $P_j$, $j \neq i$. Observe that such overlay links among the peers have created a network (a complete graph) of r peers at Level 1. That is, the diameter is 1 overlay hop. Observe that the periphery of this network appears as a ring network *(transit ring network)* consisting of the peers $P_i$ ($0 \leq i \leq r-1$). The above addressing scheme leads to the following observation.

**Observation 1.** *Any insertion of a group head $P_l$ always takes place between the current last group head $P_{l-1}$ and the first group head $P_0$ along the transit ring network.*

Note that a recently inserted group head $P_l$ has direct logical connections to each existing $P_i$; however according to the *transit ring network*, its predecessor is $P_{l-1}$ and successor is $P_0$. This idea will help in designing inter-group data look up algorithm with *anonymity*. In addition, use of (CRT) allows a group-head address to be identical to the code given to a resource type owned by the group; for example, logical code for resource type $r_i$ owned by group head $P_i$ is (X + iM). It will have a positive impact on the efficiency of data lookup algorithms.

**Lemma 1.** *Diameter of the Level-1 network is 1 overlay hop.*

*Proof: A logical link exists between two group heads $P_i$ and $P_j$ if their addresses are mutually congruent. Since all addresses assigned to Level-1 peers are congruent to each other because congruence relation is an equivalence relation, hence, every $P_i$ is directly connected to every other $P_j$. Hence, the network is a complete network of diameter 1 overlay hop.* ●

In this architecture, each $P_i$ maintains a table of 'information tuples', called as *Global Resource Table (GRT).* The table grows dynamically in size with the arrivals of new group heads. Therefore, the size of the table can be at most r. In other words, maximum number of the tuples in the table can be r, the number of distinct resource types. In this context, note that number of distinct resource types is usually very small compared to the number of peers in a P2P system [13]. The $i^{th}$ tuple appears as *<resource name $R_i$, resource code/group head $P_i$'s address code, $P_i$'s IP address>*. A new tuple is created in the table when a recent group head arrives; therefore, the $i^{th}$ tuple is created before the $(i+1)^{th}$ tuple and hence, based on the resource code/group head's address (i.e. based on the arrival sequence of the group heads) the table remains always sorted. Observe that every $P_i$ in the GRT is an immediate neighbor of every other $P_j$ in the GRT because all group heads are logically linked to each other directly.

*Level-2 address assignment and neighborhood relations*: We have to use the r congruencies to assign addresses to the members of the different groups. Without any loss of generality, let us consider an already existing group head, say $G_i$ headed by $P_i$ that has the resource type $R_i$.

We consider the $i^{th}$ congruence, viz., $a_i x \equiv b_i \pmod{m_i}$ for address assignments for peers in $G_i$. Note that this congruence has a solution $x_i$, which is unique modulo $m_i$. any solution $x_i'$ of it is congruent to $x_i$ modulo $m_i$. In addition, all solutions of the form $[x_i + k.m_i]$, where k is an integer, are mutually congruent since 'congruence relation' is an 'equivalence relation'. We assign $x_i$ as the Level-2 address for the group head $P_i$. Later based on the sequence of arrivals of peers possessing the same resource type, peer $p_j$ ($\in G_i$) will get the address $(x_i + j.m_i)$ and $p_{j+1}$ ($\in G_i$) will get address $[x_i + (j+1).m_i]$ assuming that $p_{j+1}$ joins after $p_j$, and so on. As before, based on our mathematical model two peers are neighbors if their logical addresses are mutually congruent; meaning thereby that two peers in a group are directly linked if their logical addresses are mutually congruent. Therefore, because of the equivalence relation, each peer in $G_i$ is logically connected to every other; hence, the diameter of the group $G_i$ (i.e. the cluster $N_i$) is 1 overlay hop, making the cluster a complete graph. In a group each member maintains a list of the other peers which are also its immediate neighbors.

**Lemma 2.** *Diameter of a Level-2 group (cluster) is 1 overlay hop.*

*Proof. Two peers in a group (cluster) are directly linked together if their logical addresses are mutually congruent. Address assignments of peers in a Level-2 cluster are based on the mutually congruent solutions of an LDE. Since all these solutions (addresses) form an equivalence class, hence, each peer in a cluster is directly linked to each other. Therefore, the diameter of any cluster at Level-2 is 1 overlay hop.* ●

**Theorem 1.** *Diameter of the hierarchical two-level structured architecture is 3 overlay hops.*

*Proof.* Level 2 clusters have diameter 1 each and the Level-1 network has the diameter 1. Therefore, the maximum distance between any two peers belonging to two different clusters is 3. Hence, the proof follows. ●

**Remark 1.** *There are infinite number of solutions which are congruent to the one mutually congruent solution of any LDE considered in CRT, hence, size of a cluster at Level-2 can be made very large (theoretically unlimited), yet the diameter remains 1.*

**Observation 2.** *Each group head has two different logical addresses; one from Level-1 assignment and one from Level 2 assignment.*

It has important implication related to intra-group and inter-group data lookup queries. Level 1 assigned addresses are used to move the query from one group (cluster) to another for inter-group lookup and also for data lookup with anonymity (not considered in this paper); Level 2 assigned addresses are used to answer intra-group query.

**Observation 3.** *Different group heads may get identical Level 2 assigned addresses.*

It will not affect any intra-cluster lookup query in a cluster, as this address is local to this cluster only.

```
node p_a (∈ G_i) broadcasts in G_i for < R_i, V_b >
   // one-hop communication since G_i is a complete
      graph
if p_b with < R_i, V_b > then
      p_b unicasts < R_i, V_b > to node p_a
else
      search for <R_i, V_b > fails
end
```

Fig. 2 Algorithm 1: Intra-Group-Lookup

## III. DATA LOOKUP

In this section, we show how the structural properties of CRT-based P2P architecture help in designing very efficient resource query algorithms - both for intra-group as well as inter-group resource lookups.

### *A. Intra-Group Data Lookup*

Without any loss of generality, let us consider data lookup in group $G_i$ by a peer $p_a$ possessing $< R_i, V_a >$ and requesting for resource $< R_i, V_b >$. The algorithm for intra-group data lookup appears in Fig. 2. The time complexity is $O(1)$.

### *B. Inter-Group Data Lookup*

In the proposed architecture, any communication between a node $p_i \in G_i$ and $p_j \in G_j$ takes place only via the respective group-heads $P_i$ and $P_j$. Without any loss of generality let a peer $p_i \in G_i$ request for a resource $< R_j, V^* >$; where $R_j$ denotes a resource type and $V^*$ denotes an instance of $R_j$. As is evident from the architecture that peer $p_i$ knows that $R_j \notin G_i$. We denote the IP address of a node $P_m$ as $IP(P_m)$. The algorithm appears in Fig. 3. The time complexity of the algorithm is $O(1)$. It needs a maximum of six overlay hops irrespective of in which groups the requesters and the corresponding responders reside.

### *C. Data Lookup Complexity Comparison*

In Table 1, we have stated the complexities of the presented data lookup approaches along with those of some other noteworthy structured approaches.

Table 1 Data Lookup Complexity Comparison

| | CAN[15] | Chord [5] | Pastry [4] | Our Work |
|---|---|---|---|---|
| **Architecture** | DHT-based | DHT-based | DHT-based | CRT-based |
| **Lookup Protocol** | {Key, value} pairs to map a point P in the coordinate space using uniform hash function. | Matching key and NodeID. | Matching key and prefix in NodeID. | Inter-Group: Routing through Group-heads; Intra-group: Inside a Graph. |
| **Parameters** | *N*-number of peers in network; *d*-number of dimensions. | *N*-number of peers in network. | N-number of peers in network; *b*-number of bits ($B = 2^b$) used for the base of the chosen identifier. | *r* - Number of distinct resource types; N-number of peers in network. $r << N$ |
| **Lookup Performance** | $O(d N^{1/d})$ | $O(\log N)$ | $O(\log_B N)$ | Intra-group: $O(1)$ Inter-group: $O(1)$ |

```
Node pa (ϵ Gi) unicasts request for < Rj, V*> to
group-head Pi
Pi determines IP(Pj) from GRT
Pi unicasts the query to Pj    // Pi is directly linked
with Pj
if Pj possesses < Rj, V*> then
      Pj unicasts < Rj, V*> to Pi
      Pi unicasts < Rj, V*> to pa
else
     Pj broadcasts the request for <Rj, V*> in group Gj
        // one-hop communication since Gj is a
         complete graph
     if pb (ϵ Gj) with <Rj, V*> then
          pb unicasts < Rj, V*> to Pj
          Pj unicasts < Rj, V*> to Pi
          Pi unicasts <Rj, V*> to pa
     else
```

Fig. 3 Algorithm 2: Inter-Group-Lookup

## IV. PEERS JOINING THE SYSTEM

We consider the following two situations in this paper:

1 A new node possessing an existing resource type wishes to join.

2 A new node with a new resource type wishes to join.

We assume that any new node $p$ wishing to join the system contacts a well-known server that sends the IP address of the group-head $P_0$ of the first created group $G_0$ in the P2P network. In fact, this IP address is also the address of the first peer to join the system. The IP address of the server can be obtained by a DNS-like public service. New node $p$ then sends a join request to P0. All join requests are processed sequentially by P0 by putting arriving requests in a queue. After a requesting new node p joins the P2P network successfully, it sends an ACK to P0. P0 then starts processing the next request from its queue.

```
New peer pi with resource type Rk unicasts its join
request to P0
P0 determines the group Gk for pi from its GRT
P0 unicasts IP(pi) to Pk
          // Group head of Gk
Pk assigns pi with the next available address
  [xk + i.mk]
  // xk is the Level-2 address for the group head Pk
Pk includes pi in its list of neighbors in Gk
Pk asks all members of Gk to include pi in their lists
Pk sends the updated list of neighbors in Gk to pi
pi establishes direct logical links to all members of
Gk
          // Level-2 network is a complete graph
```

Fig. 4 Algorithm 3: Join-Existing

New peer $p$ with resource type $R_l$ unicasts its join request to $P_0$
$P_0$ assigns its logical address as $[X + l.M]$ // It is the Level-1 address for $p$
$p$ becomes the group head $P_l$ of group $G_l$ and *code* $(R_l)$ = *logical address of* $P_l$
$P_0$ includes the tuple for $P_l$ in its GRT
// Tuple for $P_l \rightarrow$ < resource name $R_l$, resource code/group head $P_l$'s address code: $[X + l.M]$, *IP* $(P_l)$ >
$P_0$ sends a copy of the updated GRT to $P_l$
$P_0$ asks all group heads $P_i$, $0 \le i \le l-1$ to include the tuple for $P_l$ in their GRTs
$p$ $(P_l)$ establishes direct logical links to all group heads
// Level-1 network is a complete graph

Fig. 5 Algorithm 4: Join-New

In the proposed scheme, for the case of new peers joining with existing resource type, every group member adds the new peer in its list of neighbors in the group. In case of peers joining with new resource type, a new tuple is created in the GRT when a new group head arrives and every group head updates its GRT with the new entry. Note that the $i^{th}$ tuple is created before the $(i+1)^{th}$ tuple and hence, based on the arrival sequence of the group heads the table remains always sorted.

### A. New Peer with Existing Resource Type

The method of joining for a new node with an existing resource type is quite simple and is described in Algorithm *Join-Existing* (Fig. 4). In this algorithm, $p_i$ is a new node ($i^{th}$ arrival in group $G_k$) having an existing resource type $R_k$. Peer $p_i$ is assumed to have already obtained the IP address of $P_0$.

### B. New Peer with New Resource Type

Let $p$ be a new peer, which wishes to join the overlay P2P network with a new resource type $R_l$. Let $S_R = R_i$, $0 \le i \le l-1$, be the set of the existing resource types in the system. By following the logical address assignment process the code for the resource type $R_l$ will be [X + l.M] and this code will also be the new logical address of the joining node $p$, also the group-head $(P_l)$ of a new group $G_i$ in the system. The join process is described in detail in the algorithm *Join-New* (Fig. 5).

## V. CONCLUSION

In this work, we have presented a novel two level hierarchical P2P network architecture in which diameter of each cluster at Level- 2 is just 1 overlay hop and diameter of the network at Level-1 is also 1 overlay hop. This is a significant improvement over our recently designed highly efficient two level hierarchical architecture [9], [10], [11]. The noteworthy features of the architecture are: (1) the hierarchical P2P structure's diameter is only 3 overlay hops and (2) the multiple logical addresses assigned to some particular peers. Because of such low diameters the time complexity of each of the data lookup algorithms is $O(1)$. Such an architecture has been possible because of the use of a mathematical model based on the Chinese Remainder Theorem. To the best of our knowledge it is the first time that neighborhood relations among peers in an overlay network has been defined using such a mathematical model. 'Work in progress' includes anonymity and security considerations in communication protocols, fault-tolerance, and design of effective P2P Federation.

## REFERENCES


[1] P. Ganesan, Q.Sun, and H. Garcia-Molina, "Yappers: A peer-to-peer lookup service over arbitrary topology," in Proceedings of the IEEE Infocom 2003, San Francisco, USA, March 30 - April 1 2003.

[2] Y. Chawathe, S. Ratnasamy, L. Breslau, N. Lanham, and S. Shenker, "Making gnutella-like p2p systems scalable," in Proceedings of the ACM SIGCOMM, Karlsruhe, Germany, August 25-29 2003.

[3] B. Y. Zhao,L. Huang, S. C. Rhea, J. Stribling, A. Zoseph, and J. D. Kubiatowicz, "Tapestry: A Global-Scale Overlay for Rapid Service Deployment", IEEE J-SAC, vol. 22, no. 1, pp. 41-53, Jan. 2004.

[4] A. Rowstron and P. Druschel, "Pastry: Scalable, Distributed Object Location and Routing for Large Scale Peer-to-Peer Systems", Proc. IFIP/ACM Intl. Conf. Distributed Systems Platforms (Middleware), pp. 329-350, 2001.

[5] I. Stocia, R. Morris, D. Liben-Nowell, D. R. Karger, M. Kaashoek, F. Dabek, and H. Balakrishnan, "Chord: A Scalable Peer-to-Peer Lookup Protocol For Internet Applications", IEEE/ACM Tran. Networking, vol. 11, No. 1, pp. 17-32, Feb. 2003.

[6] M. Yang and Y. Yang, "An Efficient Hybrid Peer-to-Peer System for Distributed Data Sharing", IEEE Trans. Computers, vol. 59, no. 9, pp. 1158-1171, Sep. 2010.

[7] M. Xu, S. Zhou, and J. Guan, "A New and Effective Hierarchical Overlay Structure for Peer-to-Peer Networks", Computer Communications,Elsevier, vol. 34, pp. 862-874, 2011.

[8] D. Korzun and A. Gurtov,"Hierarchical Architectures in Structured Peer-to-Peer Overlay Networks", Peer-to-Peer Networking and Applications, Springer, pp. 1-37, March 2013.

[9] Bidyut Gupta, Shahram Rahimi, Ziping Liu, and Sindoora Koneru,

"Design of Structured Peer-to-Peer Networks Using Linear Diophantine Equation", Proc. CAINE 2014, pp. 147-151, New Orleans, Oct., 2014.
[10] N. Rahimi, K. Sinha, B. Gupta, and S. Rahimi, "LDEPTH: A low diameter hierarchical p2p network architecture," Proc. 2016 IEEE Int. Conf. on Industrial Informatics (IEEE INDIN'16), Poitiers, France, July, 2016.
[11] Bidyut Gupta, Nick Rahimi, Shahram Rahimi, and Ashraf Alyanbaawi, "Efficient Data Lookup in Non-DHT Based Low Diameter Structured P2P Network", Proc. IEEE 15th Int. Conf. Industrial Informatics (IEEE INDIN'17), July 2017, Emden, Germany.
[12] Bidyut Gupta, Nick Rahimi, Henry Hexmoor, and Koushik Maddali, "Design of a New Hierarchical Structured Peer-to-Peer Network Based On Chinese Remainder Theorem", to appear in the Proc. The 33rd International Conference on Computers and Their Applications (CATA), March 2018, Las Vegas, USA.
[13] J. Cheng and R. Donahue, "The Pirate Bay Torrent Analysis and Visualization", Int. J. Science, Engineering, and Computer Technology, 3(2), 38, 2013.
[14] Z. Peng, Z. Duan, J.J. Qi, Y. Cao, and E. Lv, "HP2P: a Hybrid Hierarchical P2P Network", Proc. Intl. Conf. Digital Society, 2007.
[15] Ratnasamy, S., Francis, P., Handley, M., Karp, R., & Shenker, S. (2001). *A scalable content-addressable network* (Vol. 31, No. 4, pp. 161-172). ACM.