ORIGINAL RESEARCH

CrossMark

PCSM: an efficient multihop proximity aware clustering scheme for mobile peer-to-peer systems

Moufida Rahmani[1] · Mahfoud Benchaïba[1]

Abstract

A mobile peer-to-peer (MP2P) system results from overlaying a peer-to-peer (P2P) system over a mobile ad hoc network (MANET). A cluster/superpeer based architecture can be used as an efficient solution to reduce communication redundancy and network traffic induced by flooding. In this paper, we propose an efficient multihop Proximity aware Clustering Scheme for Mobile peer-to-peer systems (PCSM). PCSM is based on the physical proximity of peers and reduces the mismatch between the P2P overlay and the network layer. PCSM integrates three factors to allow the new peer to efficiently select the cluster to join, namely the number of physical hops, the cluster size and the availability of the clusterhead. Additionally, a maintenance process manages the mobility of peer. The simulation results show that our overlay topology fits the MANET underlay and that PCSM enhances the results of the search process in terms of the average file-discovery delay and false-negative ratio. Additionally, PCSM performs better than the existing cluster-based P2P overlay regarding of load balancing and routing overhead.

Keywords Mobile peer-to-peer · Clustering · Physical proximity · Overlay topology construction

1 Introduction

Peer-to-peer (P2P) systems are an alternative to traditional client/server systems: every node acts as both a client and a server for respectively requesting and sharing resources. P2P overlays were initially deployed over fixed wired underlays such as Internet. Nowadays, mobile devices owners need to exchange data everywhere, anytime, which leads to the deployment of P2P overlays over mobile environment in the both cellular network and mobile ad hoc network (MANET). In this paper, we target the P2P overlays over MANET (MP2P). MP2P constitute an interesting environment to exchange books, games, MP3 file, or even movies in public place like gardens, libraries and shopping malls or in temporary events like conferences and concerts.

P2P overlays can be classified as unstructured and structured. Structured P2P overlays (Stoica et al. 2001;

Ratnasamy et al. 2001; da Hora et al. 2009; Liang et al. 2011; Shah and Qian 2010b; Shah et al. 2014, 2016) impose a particular organization and linkage of peers and use a Distributed Hash Table (DHT) to map the identifier of a resource to the identifier of the peer holding it. While the DHT decreases the number of lookup messages and reduces search time, its maintenance cost is important. Overlaying a structured P2P system on a MANET induces a higher cost in messages because of the network partitioning in the physical network and the overlay layer caused by limited the radio range and the mobility of nodes (Shah et al. 2016). Unstructured P2P overlays (Ripeanu et al. 2002; Mawji et al. 2011; Papadakis et al. 2013; Shah and Kim 2014; Seddiki and Benchaïba 2015), on the other hand, impose few constraints on the overlay networks and the placement of data is completely unrelated to the overlay topology. Each peer indexes its own files and has no knowledge about the other shared files in the network and their locations. They are widely used because their design is simple. Furthermore, unstructured P2P overlays over MANET are very resilient to churn and topology changes (da Hora et al. 2009), while inducing little maintenance cost compared to P2P structured overlays. Unstructured P2P overlays can be classified as flat (a decentralized unstructured P2P overlays) and hierarchical

✉ Moufida Rahmani
morahmani@usthb.dz

Mahfoud Benchaïba
mbenchaiba@usthb.dz

1 Computer-Science Department, LSI, University of Science and Technology Houari Boumediene, Algiers, Algeria

(a cluster/superpeer based unstructured P2P overlays). A cluster/superpeer based unstructured P2P overlays performs resource search with less traffic because search only operates between CHs (Han et al. 2008b; Noor et al. 2012; Seddiki and Benchaïba 2016) and further increases scalability compared to a decentralized unstructured P2P system.

Clustering is a technique for organizing peers into different virtual groups called clusters. Each cluster is formed by a set of peers called members and managed by a clusterhead/super-peer. The clusterhead stores the index information of the shared content/files of its members. In the rest of this paper, we use the abbreviations "CH" for clusterhead. Clustering is used in many replication strategies (Mondal et al. 2006; Kumar and Kumar 2012; Rahmani and Benchaïba 2017a) and incentive mechanisms (Hu and Kuo 2012; Zhang and Antonopoulos 2013). It helps to efficiently get a global knowledge such as availability of shared resources (Rahmani and Benchaïba 2017b). In MP2P systems, clustering can be at the underlay network (Shah and Qian 2010a; Nakahara et al. 2014) and at the overlay layer (Han et al. 2008a; Li et al. 2011; Noor et al. 2012; Elgazzar et al. 2013; Saghiri and Meybodi 2017).

The challenge with clustering at the overlay layer is to match the underlay. This mismatch occurs when two physical neighbors are not logical neighbors and leads to inefficient paths between peers, i.e. a single-hop at the overlay usually incurs many hops at the underlay, increasing the traffic overhead. Another challenge with clustering is to make a careful choice regarding the clusterhead selection because choosing an appropriate CH can minimize the cost of cluster maintenance in terms of traffic. This latter becomes more important with the characteristics of MANETs such as their dynamic topology due to nodes mobility and limited energy.

The mismatch between the logical and the physical networks was addressed for structured (Abid et al. 2014; Shah et al. 2014, 2016) and decentralized unstructured P2P over MANETs (Mawji et al. 2011; Shah and Kim 2014; Shah et al. 2015; Seddiki and Benchaïba 2015) but was not clearly addressed for a cluster/superpeer based unstructured MP2P systems because they did not consider an overlay distinct from an underlay (Han et al. 2008a; Shah and Qian 2010a; Noor et al. 2012; Saghiri and Meybodi 2017). Additionally, existing approaches for clusterhead selection based on single criteria (e.g. energy (Noor et al. 2012), random number (Han et al. 2008a), degree (Han et al. 2008a), mobility (Kim et al. 2011)) can lead to poor performance because the selected CH may not be efficient.

In this paper, we propose an efficient multihop Proximity aware Clustering Scheme (PCSM) for MP2P. PCSM is an overlay clustering technique that confines the lookup process to only CHs which positively influences the lookup process: the traffic overhead and the response time are reduced, and the success rate is increased. PCSM bases on the physical proximity of peers (cross layer design) to construct an adaptive overlay topology that fits the MANET underlay. Peers that are physically close to each other are into a common overlay of clusters, therefore, the communication redundancy (inefficient paths between peers) is reduced for both the search and the establishment/maintenance of clusters. Additionally, PCSM integrates three factors to efficiently select the CH, namely the number of physical hops to choose the nearest cluster to join, the cluster size in order to balance the load of CHs and the availability of the CH. A CH with a larger availability value means that it will likely remain for a long time, leading to a reduced failure rate and a more stable system. To reduce traffic overhead, a new peer exploits the broadcast nature of MANET to discover the physically nearest peers, to establish connection with them and to join a cluster at the same time. In mobile environments, peers may move around, change connectivity and leave without notification, therefore, we propose an efficient and adaptive maintenance process in order to prevent peers isolation (absence of neighbors) or non-affiliation to CHs.

The rest of this paper is organized as follows. Related work is described in Sect. 2. Section 3 presents the problem statement. We provide detail our proposition in Sect. 4. Section 5 describes the environment and results. Finally, Sect. 6 concludes this paper.

2 Related work

Clustering and P2P overlay topology construction, considering the network underlay, are separately addressed in literature. The first subsection addresses the overlay topology construction, the second subsection addresses the clustering techniques and the last one addresses the limitations of the presented clustering techniques and how PCSM overcomes these limitations.

2.1 P2P overlay topology construction

To our knowledge, Shah and Qian (2010a) and Shah et al. (2011) are the first to construct an unstructured P2P overlay over MANET by considering the underlying physical network topology. The new joining peer first discovers the closest peers of the P2P network to connect to them. The process of discovering and establishing connection with the other peers are sequentially done, requires more than one message and increases the establishment delay. At the maintenance phase, authors use a root-peer to choose the appropriate peer responsible of maintaining the neighbor relationship. The closest peer to the root-peer is conventionally chosen as responsible to send messages to its neighbors. The limitation of a the root-peer approach is that it requires maintenance, which generates additional network traffic.

Seddiki and Benchaïba (2015) proposed an adaptive overlay over MANET to improve some drawbacks of Shah and Qian (2010a). In their proposition, authors decreased the generated redundant network traffic during the overlay building by using the same request to simultaneously discover and establish connection with the other peers. They also reduced the network traffic generated by the use of the root-peer in the maintenance process. The neighbor responsible of sending *hello* messages to others was the one who first sent the connection request. Our proposal is different than that of Seddiki and Benchaïba (2015) in that the TTL of the connection request is not fixed and adapts to physical network topology change, therefore, the search space for neighbors is extended or shortened according to actual underlying network topology.

The survey of Shah et al. (2017) compared the features and limitations of existing overlay construction approaches over MANETs.

2.2 Clustering schemes

Clustering schemes used at the overlay are classified according to the the number of parameters featured in the clusterhead selection algorithm into the single-parameter clusterhead selection and the multi-parameter clusterhead selection.

- Single-parameter clusterhead selection. Han et al. (2008a) proposed to select the CH based on the degree of a peer where the peer with the highest degree among its neighbors becomes CH and lets its adjacent peers become members. They also proposed an alternative where the degree criterion is replaced by a randomly generated number: the peer with a highest randomly generated number becomes CH. Kim et al. (2011) proposed to select a CH based on its mobility pattern. The peer with the lowest mobility among its neighbor peers is selected as CH and its neighbors become its members. In the proposal of Han et al. (2008b), the criteria to choose the CH is the residual energy. In the proposal of Kim et al. (2011) and Han et al. (2008b), the network is periodically reconstructed which means that CHs are periodically selected to reflect the peers movements and their energy levels in the system, generating traffic overhead and leading to long query delay on searching files. Noor et al. (2012) extended the proposition of Han et al. (2008b) in that a joining peer that does not receive any message from existing CHs checks whether other peers are trying to join a cluster and compares its energy level to theirs. If its own energy level is less than any of the energy levels of its neighboring peers, it waits so that the peer with the highest energy level becomes CH. Then the new peer joins this cluster and informs the other peers with a lower energy level of its affiliation in order that the remaining unaffiliated peers repeat the above CH selection process until affiliation. In their proposition, the authors also suggest the role of secondary CH undertaken by the member peer with the highest energy level that takes the primary CH role when this latter leaves the system or if its residual energy becomes lower than a threshold value. The *id* of the secondary CH is broadcasted to all members within the cluster. In the maintenance process, if the member does not receive any *alive* message from its primary CH after a period, it waits again the same amount of time to get an *alive* message from its secondary CH. When a member receives the *alive* message from its secondary CH, it affiliates to the secondary and affiliates to it. Otherwise, the member must find a new cluster and the same process repeats.

- Multi-parameter clusterhead selection. RobP2P, proposed by Elgazzar et al. (2013), integrates many factors to select CHs, including the current mobility, the energy level, the network capability, the mean uptime and the degree. The clusterhead selection algorithm is invoked every time a new peer joins a cluster. The most effective peer becomes CH, which on one hand accommodates the dynamic context change of MP2P networks but generates high traffic overhead and leads to long query delay on searching files. Amirazodi et al. (2018) also highlighted the re-selection of CH: each peer updates its role to CH or member after a comparison of its own capacity and its neighbors ones in each round. In the proposal of Li et al. (2011), the new joining peer selects the CH with the maximum connection time. The connection time between the connected peers can be determined by the location, velocity and communication range of each mobile peer. In the maintenance process, when a CH disappears with notification, members broadcast a message to each other to elect a new CH. The peer who can make the network longer and has the best connectivity is elected as the new CH. In case the CH disappears without notification, the member peer who first detects the departure of its CH becomes a CH and generates a new cluster. The other members join this new cluster.

2.3 Limitations

Han et al. (2008a) and Kim et al. (2011) elect CH do not consider residual energy as a clustering factor. Therefore, if a node with low residual energy is elected as CH, it may fail early enough due to energy depletion, leading to more traffic overhead to elect another CH, to the failure rate increase, to longer response delay in the search process and affecting the overall stability of the system. Our solution selects highly available peers as CHs in a distributed fashion to solve the precedent issues.

Balancing the load between CHs is crucial in clustering. Noor et al. (2012), Han et al. (2008a, b), Kim et al. (2011) and do not limit the number of cluster members. Therefore, some CHs exhaust their energy more quickly than others and traffic overhead increases due to the maintenance process. Our approach limits the members in each cluster at a reasonable number and considers this factor in the clusterhead selection.

The solution of Han et al. (2008a) does not include a maintenance process. Secondary CHs are important because they minimize the time a peer remains without a CH (Noor et al. 2012). However, MP2P where the topology changes frequently, maintaining up-to-date the information about secondary CH is very difficult and generates important traffic overhead. Using the broadcast as proposed by Li et al. (2011) to elect the new CH is not suitable because it generates traffic overhead. Our approach maintains the structure of clustering by selecting the new CH among the cluster members in case the CH quits the MP2P overlay or when its energy level becomes insufficient (beneath a threshold value).

In all the clustering methods presented above, the bootstrapping process and neighborhood establishment process are not clearly mentioned. Authors do not detail how these two processes proceed. Additionally, Han et al. (2008a, b),

Kim et al. (2011), Elgazzar et al. (2013) and Amirazodi et al. (2018) consider that the P2P system already exists and is divided into groups, and they apply the clusterhead selection algorithm within each group independently. The grouping is based on the physical proximity. Applying a clustering scheme after topology overlay construction can generate double traffic overhead whereas our approach functions in an incremental manner from the beginning of network deployment.

The mismatch between the logical and the physical networks is a challenge in a cluster/superpeer based unstructured MP2P; however, clustering methods presented above neither mention nor address this problem.

To decrease the traffic overhead and the query delay on searching files, PCSM activates the clusterhead selection algorithm when it is necessary (for example when CH leaves the P2P network) and not every time a new peer joins the P2P network as proposed by Elgazzar et al. (2013) and Amirazodi et al. (2018) or periodically as proposed by Kim et al. (2011) and Han et al. (2008b).

Table 1 summarizes the comparison of the presented schemes along with PCSM regarding some criteria: the clusterhead selection feature, load balancing (addressed or not addressed), the number of hops between a CH and its members (1-hop, multi-hop or not mentioned), the maintenance

Table 1 Clustering approaches comparison

Mechanism	Clusterhead selection feature	Load balancing	The number of hop	Maintenance process	Neighborhood establishment	Principle
Noor et al. (2012)	Energy level	Not addressed	1-hop	Addressed	Not addressed	From the beginning of network deployment
First method of Han et al. (2008a)	Degree of peer	Not addressed	1-hop	Not addressed	Not addressed	After topology construction
Second method of Han et al. (2008a)	Random number	Not addressed	1-hop	Not addressed	Not addressed	After topology construction
Li et al. (2011)	The maximum connection time, the hop count away from the superpeer, cluster size	Addressed	Multi-hop	Addressed	Not addressed	From the beginning of network deployment
Kim et al. (2011)	Mobility pattern	Not addressed	1-hop	Addressed	Not addressed	After topology construction
Han et al. (2008b)	Energy level	Not addressed	1-hop	Addressed	Not addressed	After topology construction
Elgazzar et al. (2013)	Current mobility, energy level, network capability, mean uptime, connectivity degree	Not addressed	Not mentioned	Addressed	Not addressed	After topology construction
Amirazodi et al. (2018)	Capacity	Not addressed	Not mentioned	Addressed	Not addressed	After topology construction
PCSM	Physical hop count, cluster size, CH's availability	Addressed	Multi-hop	Addressed	Addressed	From the beginning of network deployment

process (addressed or not addressed), the neighborhood establishment (addressed or not addressed), principle (after the topology construction or at network deployment).

3 Problem statement

We consider a MP2P system with two types of mobile nodes: peer and non-peer. A peer is a member of the P2P overlay network that accesses and/or shares a resource whereas a non-peer node does not, but still cooperates to forward messages to the other mobile nodes. In this section, we describe the functioning of PCSM and compare it to the approach proposed by Shah and Qian (2010a) regarding a common scenario.

Figure 1a–d illustrates the joining process scenario proposed by Shah and Qian (2010a). Authors use the clustering technique at the MANET underlay to construct a decentralized unstructured P2P overlay. Figure 1a presents a MANET underlay example with P1 as the root-peer and Fig. 1b presents its corresponding overlay. If node N5 wants to join the P2P overlay, it broadcasts a request message with a time-to-live (TTL) using expanding-ring-search algorithm to find the closest online peer in the P2P network. If the receiving node is a peer, it sends back the list of its direct neighbors. In this example, P2 and P3 send back their direct neighbors that are P5 and P1. Otherwise, the non-peer nodes rebroadcast the request message if the TTL has not expired. N5 sends connection requests to P2, P3, P1 and P5. These latter accept N5 as a neighbor. After that, each of P2, P3, P1, P5 and N5 executes the MST algorithm with themselves as sources vertexes

to identify redundant links. Redundant links are removed and an efficient overlay, closer to the physical network, is built. The weight of the links between two peers in the MST corresponds to the number of hops between them. The connected graph of N5 is shown in Fig. 1c. After executing the MST algorithm, the resulting MST of N5 is shown with bold lines in Fig. 1c. Notice that N5 drops connection with P1 and P5. The resulting final overlay topology is shown in Fig. 1d. In case the new joining peer does not receive any response message after a predefined time, it rebroadcasts the request message by incrementing the TTL value provided the TTL value has not reached a maximum threshold value.

Suppose now that N5 wants a file owned by P5. It sends a lookup message to its direct neighbors in the overlay. When P2 and P3 receive the lookup message, they forward it to their direct neighbors, excluding the one from which the request is received. P1 does the same as P2 and P3. Upon receiving a lookup message, P5 sends the response message to N5. This lookup message produces seven transmissions in the underling network as shown through thin arrows in Fig. 1a.

Shah and Qian (2010a) provide an overlay topology which is aware of the MANET underlay but the proposal has some drawbacks that can be improved as follows.

– Peers exchange a significant amount of messages when they join the P2P overlay because they must discover the physically nearest peers in a first step then try to establish a connection with the selected peers. This generates redundant network traffic and increases connection establishment delay.

Fig. 1 An overlay topology construction according to Shah and Qian (2010a)
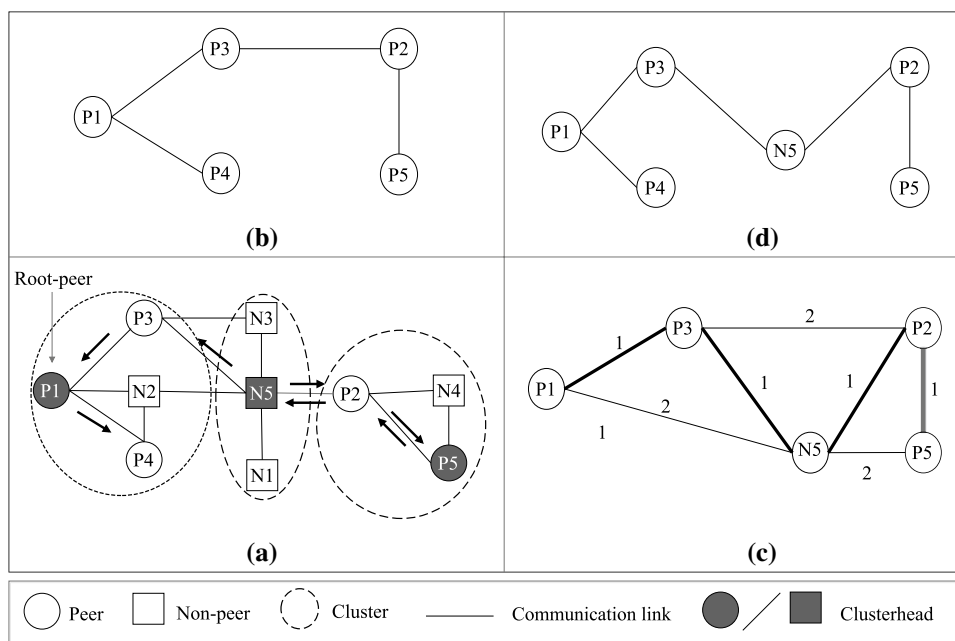
– The strategy uses a root-peer which is connected to all peers as a reference point to designate which neighbor is responsible for maintaining the neighbor relationship (send *hello* message). All the peers in the P2P network must maintain a connection to the root-peer which generates additional network traffic. Moreover, the root-peer can randomly join/leave the P2P network and the issue of the leaving root-peer is not addressed.

– During the search process, flooding is used at the overlay layer. This increases search delay and network traffic.

An example of PCSM is shown in Fig. 2. When N5 joins the system, it broadcasts a connection request message with a TTL value of 2. Suppose 2 is the number of maximum and minimum number of neighbor peers. N5 chooses to join the cluster 2 because the CH of cluster 2 is closer than the CH of cluster 1 and also because the cluster size of cluster 2 is less that cluster size of cluster 1. Since N5 is physically closer

to P2 and P3, these latter accept N5 as a neighbor and drop their connections to each other.

We consider that N5 wants a file owned by P5. N5 sends a lookup message to its CH P2. This latter sends a response message to N5 informing it that P5 has the target file. Therefore, only two transmissions in the underlay network are necessary as shown through the thin arrows in Fig. 2a. Clustering at the overlay layer can be used is as an efficient solution to reduce communication redundancy and network traffic induced by flooding.

PCSM fits the MANET underlay as the proposition of Shah and Qian (2010a). Indeed, two physical neighbors are also logical neighbors and peers that are physically close to each other are into common overlay clusters. However, PCSM achieves better network performance because it decreases the network traffic compared to the described technique. PCSM tries to address the listed issues by using the following efficient strategies which significantly decrease the network traffic.

– A new peer exploits the broadcast nature of MANETs to discover the physically nearest peers and establishes a connection with them simultaneously rather than doing it sequentially as proposed by Shah and Qian (2010a).

– The neighbor responsible of sending *hello* messages to others is the one who first sends the request connection. Therefore, it is not necessary to use the root-peer as proposed by Shah and Qian (2010a).

– PCSM restricts the search to CH rather to all peers (as with flooding).

4 Proposed system

In this section, we describe our proposed scheme, regarding the following features.

– The overlay topology construction: each joining peer discovers some existing peers in the overlay and becomes a neighbor of the physically closest nodes. The physical hops tolerated for searching neighbors is a dynamic value called *Rn*. This latter varies according to the physical network topology.

– The maintenance process for the overlay topology: each peer tries to maintain the connections with its overlay neighbors. When the number of its neighbors reaches a minimum threshold, it finds other neighbors in order to prevent the isolation from the MP2P overlay. In response to the mobility of peers, our solution also ensures that each peer maintains the physically closest neighbor peers.

– The clustering approach (PCSM): our topology is structured into clusters with a limited radius. The choice of
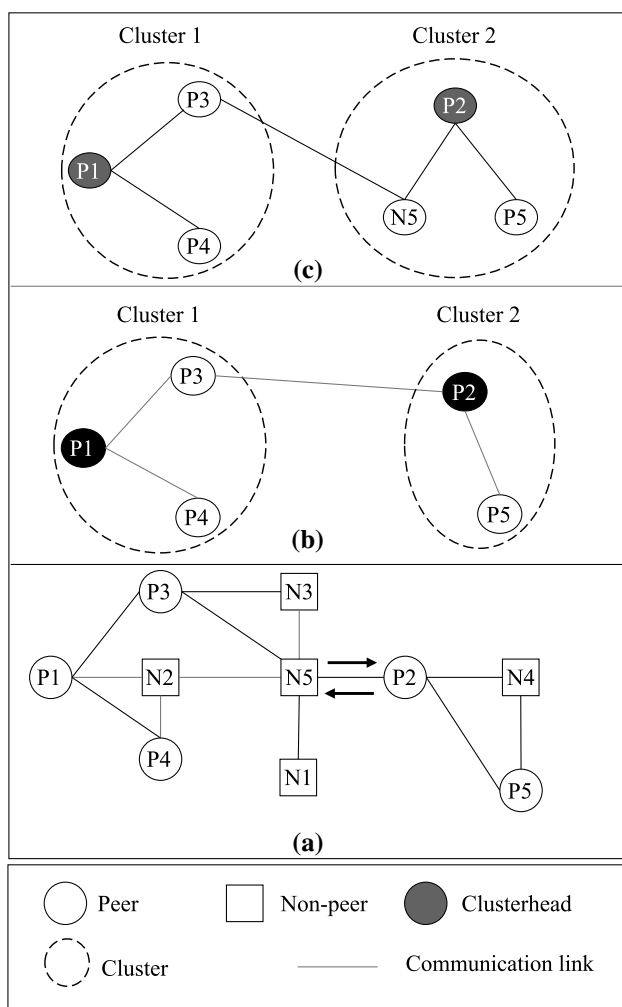

Fig. 2 An overlay topology construction according to PCSM

the CH is based on the availability, which is related to energy and online time.

- The maintenance process for PCSM: each cluster continuously gathers peers that are at a fixed number of physical hops. It also tries to reduce the overlay maintenance by keeping the structure of clustering and selecting the new CH from members in case the CH quits the MP2P overlay.
- The messages used for constructing topology and forming the clusters are the same, which reduces the traffic overhead.
- The overlay topology construction and the clustering approach are based on the physical proximity.

4.1 Structures and variables

Each peer P_i has its own information (IP_p, IP_c, *Nature*, Ot_{P_i}, E_{P_i}, $Avail_{P_i}$), where IP_p is IP address of peer P_i, IP_c is IP address of its CH, *Nature* is to distinguish whether the peer is a CH or member, Ot_{P_i} is the estimated online time of P_i, E_{P_i} is energy amount at the device of P_i and $Avail_{P_i}$ is the availability of P_i. When P_i joins the P2P overlay, estimates its energy amount for a session E_{P_i}. Ot_{P_i} is the average connection time during the last sessions and is calculated as follows.

$$Ot_{P_i} = \sum_{0}^{n-1} T_{P_i}/n \qquad (1)$$

$\sum_{0}^{n-1} T_{P_i}$ is the online lifetime of a peer during the n last sessions. Since a peer joins and leaves the network unexpectedly, it is impossible to predict when a peer will leave the network, but we assume that the longer a peer is in the system, the more likely it will remain.

$Avail_{P_i}$ is used during the clusterhead selection. The bigger the $Avail_{P_i}$ value, the more available the peer is and can perform tasks. $Avail_{P_i}$ is calculated as follows.

$$Avail_{P_i} = E_{P_i}/Ot_{P_i} \qquad (2)$$

$Avail_{P_i}$ is the energy that P_i can offer per time unit hoping to remain online for an Ot_{P_i} time interval.

Each peer (CH or member) maintains a *neighbor_peer* table at the application layer. Each entry of this table is associated with a neighbor, contains its IP address and its physical distance from the current peer. A peer also maintains a *neighbor_cache* list which contains the list of peers that have asked to be neighbors but have not confirmed their status yet. When a peer becomes a CH, it also maintains a *cluster_member* table which contains information about each peer of the cluster. Each entry of this table is associated with a member and contains its IP address, its physical distance to the CH and its availability. The physical distance between two peers is the number of physical hops and is the length in term of hops of the messages exchanged between these two peers. To route the exchanged messages, PCSM can use any routing protocol, proactive or reactive. A CH also maintains a *member_cache* list which contains the list of peers that have asked to be members but have not confirmed yet. Moreover, it uses the *neighbor_cluster* table in which it stores information about its neighbor CHs like their IP addresses. Table 2 summarizes the technical terms of our proposal.

4.2 *Rn* value calculation

Each joining peer has to discover the physically closest peers and the existing clusters in order to respectively establish a neighborhood relationship and a membership. The peer floods a request that is forwarded by peers and non-peer nodes until the TTL expires. In this initial phase, the TTL of the request is set to a fixed value. When the current number of neighbors (denoted *Num*) of a peer is lower than *Minn*, the peer must look for new neighbors to prevent its isolation. In this case, the TTL of the request is set to a value *Rn* calculated according to the Eq. 3. To compute *Rn*, the peer must know the number of its 1-hop physical neighbors (the

Table 2 Technical terms of our proposal

Notation	Significance
Rc	Radius of cluster. It is the number of physical hops tolerated to join a cluster
Rn	Radius of neighborhood search. It is the number of physical hops tolerated between neighbors
Maxn	Maximum number of neighbors for a peer
Minn	Minimum number of neighbors for a peer
Maxc	Maximum number of members in a cluster
Cluster_size	Current number of members in a cluster
Aver_dist	Average physical distance between two peers
Phy_neighbor	Number of 1-hop physical neighbors
Phy_hop	Number of physical hops between two peers

cross layer design allows retrieving this information from the network layer) and the average distance between two peers. Rn is computed as follows.

$$Rn = (Maxn/Phy\_neighbor) * Aver\_dist \qquad (3)$$

$Aver\_dist$ indicates the average distance of a peer to all others (average number of hops). It changes depending on the number of mobile nodes in the network, the rate of peers, the area and the transmission range. Since it is costly in messages to compute $Aver\_dist$ in a decentralized way, each peer estimates it based on owned or received information. The number of physical hops (Phy hop) between two peers is deduced from a received or passing messages. $Aver\_dist$ is calculated incrementally as follows.

$$Aver\_dist = (Aver\_dist + Phy\_hop)/2 \qquad (4)$$

Where $Aver\_dist$ initially equals 1.

The example in Fig. 3 illustrates the advantage of a dynamic Rn rather than a fixed one. The scenario in Fig. 3a shows a peer P1 looking for neighbors with 2 as $Maxn$. Since Rn is fixed to 2 hops, the peer P1 floods a message for searching neighbors with a TTL of 2. The message stops at the non-peer node N4 as the TTL equals 0. In this case, the peer P1 does not find neighbors and remains isolated from the P2P overlay. However, according to the Eq. 3, Rn will have the value 4 (where 1-hop physical neighbors number equals to 1 and $Aver\_dist$ equals to 2). In this case, the peer P1 can find potential neighbors (peers P2 and P4).

Another example is given in Fig. 3b, where Rn is fixed to 6. When the number of 1-hop physical neighbors equals to 2 and $Aver\_dist$ equals to 3, the calculated Rn will have the value 3. The calculated Rn value is lower than the fixed one and generates less traffic. When the peer has a small number of physical neighbors, the Rn value is large and the area to search overlay neighbors must be broad in order to find a maximum number of overlay neighbors. Otherwise, when

a peer has a great number of physical neighbors, a small Rn value avoids unnecessary traffic and finds the maximum neighbors in the nearby proximity.

4.3 Peer joining process

In this subsection, we describe our bootstapping process, neighborhood creation and cluster joining. When a new peer wishes to join a P2P overlay, it broadcasts a *discovery_request* (*hello* message) in the MANET to discover the closest peers in its area. It must indicate in the *discovery_request* that it is in the initialization phase. The initialization phase is meant to establish a neighborhood relationship but also the peer to join a cluster. Each non-peer node receiving the *discovery_request* decrements the TTL and forwards it to all its physical neighbors if the TTL is greater than 0. Each peer (member or CH) receiving this message checks the possibility of accepting the sender as a neighbor and/or as a member and sends a response message back to the requester. Messages are routed according to a routing protocol used in the MANET. Table 3 summarizes all different types of the exchanged messages during the construction and the maintenance of the overlay and the cluster.

4.3.1 Neighborhood establishment process

In our approach, the neighborhood optimization criterion is physical closeness. When a peer (member or CH) receives a *discovery_request* from a new joining peer, it checks if its current number of neighbors is lower than its capacity or if the joining peer is physically closer than one of its current neighbors. In both cases, the joining peer is accepted as a neighbor but it is not directly added to the *neighbor_peer* table. The joining peer is first added to the *neighbor_cache* list and acknowledged by a *discovery_reply* message. Otherwise, the peer rejects the joining peer by not sending
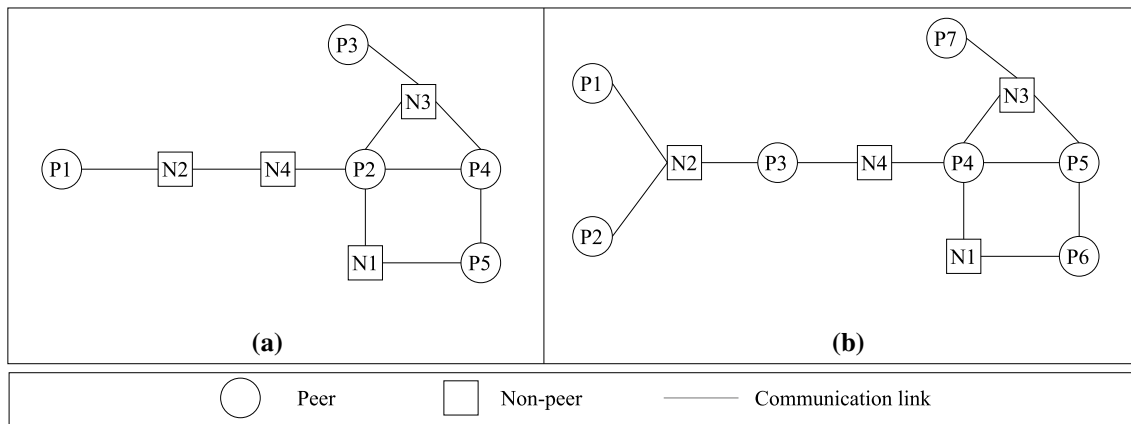

(a)   **(b)**

⃝ Peer   ☐ Non-peer   —— Communication link

Fig. 3 Example demonstrating the utility of calculating the *Rn* value

Table 3 List of used messages in our propsal.

Notation	Significance
Discovery_request	Neighborhood relationship request + lookup-cluster request
Discovery_reply	Neighborhood relationship reply + lookup-cluster reply
Neighbor_reply	Neighborhood relationship confirmation
Cluster_info	Broadcasts clusterhead affiliation
Hello (Heartbeat)	Maintains connectivity with neighbors and cluserhead
Hello_reply	Reply to *Hello* message
Join	Lookup-cluster confirmation

discovery_reply message. The *neighbor_cache* list is periodically emptied to manage other neighborhood requests. When the joining peer receives a *discovery_reply* and its current number of neighbors is higher than *Maxn*, the peer sending the *discovery_reply* message is rejected. Otherwise, the peer sending the *discovery_reply* message is directly accepted as a neighbor, added in the *neighbor_peer* table with its IP address and its physical distance, and acknowledged by a *neighbor_reply* message. When a peer receives a *neighbor_reply*, it checks in its *neighbor_cache* list if the joining peer can be accepted as a neighbor, without or with replacing another neighbor. In both cases, the joining peer is deleted from *neighbor_cache* list and added to *neighbor_peer* table. In the case that the joining peer is accepted with replacing another neighbor, the replaced peer (physically further than the requesting peer) is removed and informed.

Each peer also periodically sends a *hello* message to its neighbors to inform them that it is alive. In order to reduce the traffic overhead, we adopt the same idea used by Seddiki and Benchaïba ([2015](#)), that I only one of both neighbors (the one that sent the *discovery_request*) is responsible for

sending the *hello* message to the other. Figure [4](#) summarizes the neighborhood establishment process.

4.3.2 The cluster selection process

The cluster selection is initiated when the peer joins the network or looses its membership from a previous cluster. The *discovery_request* is used to both construct the overlay topology and join the cluster and its TTL is higher or equal to *Rc*. When a CH receives a *discovery_request* from the joining peer, it first checks if it is possible to accept the joining peer as a neighbor by executing the steps defined in Sect. [4.3.1](#). Then, it checks if it is possible to accept the sender peer as a member. The CH verifies if the joining peer is at *Rc* hops or lower. It adds the joining peer to *member_cache* list, sends a *discovery_reply* to the joining peer that is accepted as a member. If adding on the new member makes the *cluster_size* greater than *Maxc*, the joining peer is rejected.

In order to minimize the traffic overhead, the *discovery_reply* message is also used to inform the requester peer

Fig. 4 Neighborhood establishment process
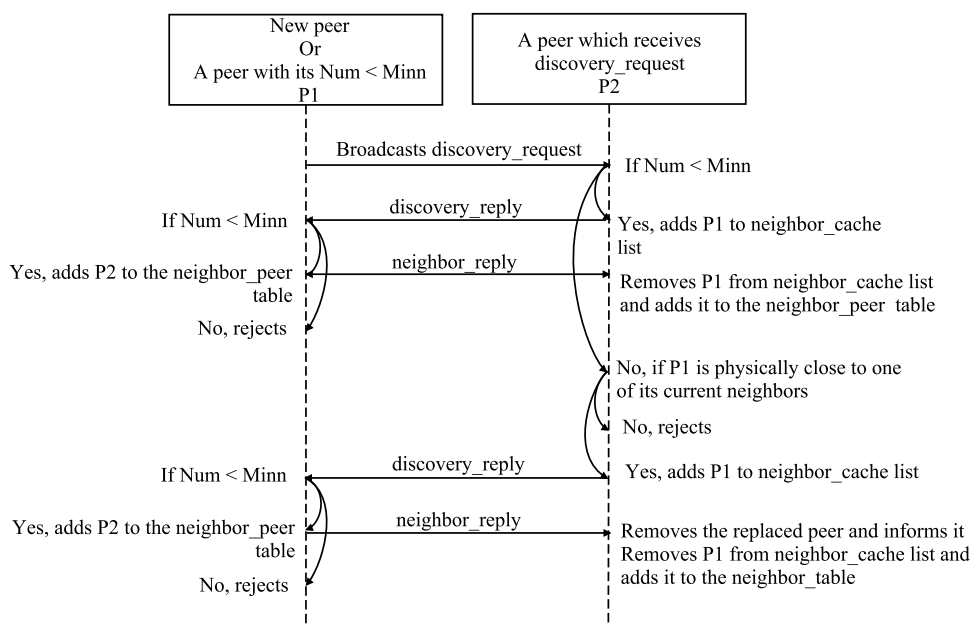
on both its neighborhood and new membership status. If the joining peer does not receive a *discovery_reply* message from a CH after a short period of time, then the joining peer becomes a CH and a new cluster is created. Upon receiving *discovery_reply* messages from multiple CH, the peer chooses the CH which has the maximum weight value noted W and sends a *a join* message. W is calculated as follows.

$$W = 1/Phy\_hop * p + (Avail_{P_i}/(cluster\_size + 1)) * q \tag{5}$$

Where $p + q = 1$ and are used to balance between the physical distance and the availability related to *cluster_size*. When receiving a *join* message, the CH stores the sending peer as a member in its *cluster_member* table. A *join* message contains the IP address, the physical distance and its availability $Avail_{P_i}$. Since our algorithm tries to gather peers that are physically close to each other into a common overlay cluster, W is based on the number of physical hops. W also takes into account the availability of the peer related to the *cluster_size* in order to balance the load between the CHs. Our algorithm reduces the traffic of re-clustering when the CH leaves the network and balances the number of CH and their members. When the new peer becomes CH, it broadcasts a *cluster_info* message with its IP address and its $Avail_{P_i}$ to notify normal peers about its existence. It must also find other CHs in the near vicinity.

In order to decrease the network traffic, members are responsible to maintain the connectivity with their CHs by periodically sending *hello* messages to their CHs to inform them that they are alive. A CH that receives the *hello*

message from a member confirms its presence with a *hello_reply*. The physical distance between each member and its CH is updated at each exchange of *hello* and *hello_reply* messages.

A simple example of a new joining peer is illustrated in Fig. 5. Let's set *p* and *q* to 0.5. The new peer P6 calculates the weight of each CH (Fig. 5a). The weight of cluster P4 is 1.5 and weight of cluster P7 is 2. As the weight of cluster P7 is greater than the weight of P4, the new peer P6 chooses cluster of P7 to join (Fig. 5b).

4.4 Update of the peer-neighbor table

When the current number of neighbors for a peer reaches *Minn*, a peer must look for new neighbors. It broadcasts a *discovery_request* with a TTL equal to *Rn* calculated according to the Eq. 3 and indicates in the *discovery_request* that it is in the update phase. Each peer receiving this message does the same steps defined in Sect. 4.3.1. When a peer receives a *hello* message from a neighbor, the physical distance of this neighbor is updated.

4.5 The peer leaving process

A peer may leave the network due to energy drain. In the best case, it informs its CH and its neighbors to update their tables. In case a peer leaves without informing, its CH and its neighbors will themselves detect its departure. If the CH (resp. neighbors) does not receive a *hello* message after a period (for example, equal to three times the transmission
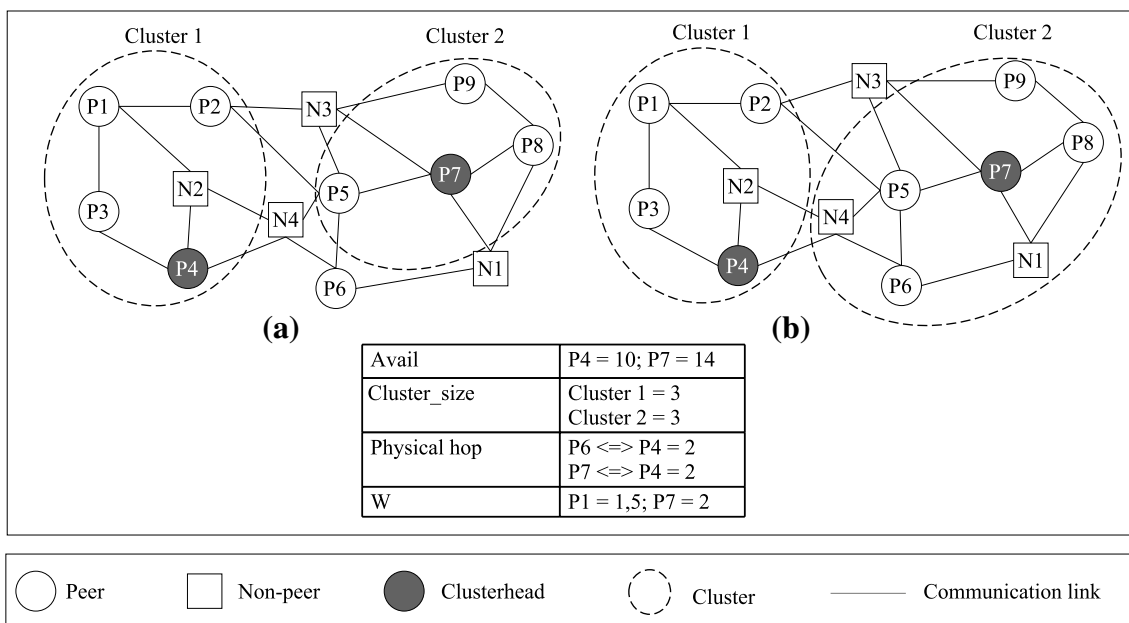

Avail	P4 = 10; P7 = 14
Cluster_size	Cluster 1 = 3 Cluster 2 = 3
Physical hop	P6 <=> P4 = 2 P7 <=> P4 = 2
W	P1 = 1,5; P7 = 2

Fig. 5 An example of a new joining peer

period), the CH (resp. neighbors) assumes that the member (resp. neighbors) is disconnected and removes it from its *cluster_member* table (resp. *neighbor_peer* table).

4.6 Clusterhead leaving process

The clusterhead either voluntarily or involuntarily

– *Voluntarily*: The CH that leaves the network voluntarily selects its successor from its members. The successor is the physically closest member with the maximum value of $Avail_{P_i}$. The selected member is informed that it is elected to be the new CH. The leaving CH also informs its members, its successor and its neighboring CHs about its departure. The new CH sends *cluster_info* messages to notify normal peers about its existence. Each peer receiving this message sends back a *join* message. Members that do not receive this message are at a physical distance to the new CH greater than *Rc*. In this case, each member detecting its disconnection to its CH broadcasts a *discovery_request* with a TTL equal to *Rc* and indicates that it is in the search phase. In this phase, only the CHs reply and the same process as in Sect. 4.3.2 is repeated. A simple example of CH departure is illustrated in Fig. 6. When the CH P2 decides to leave the network, it chooses the the closest member P1 to become CH because P1 has the largest value of $Avail_{P_i}$ compared to peer P5 (Fig. 6a). The peer P1 broadcasts a *cluster_info* message. The peer P6 does not receive this message because it is at 3 physical hops away from the new CH P1. The peer P5 must

find a new cluster to join. In this example, it joins the cluster 2 (Fig. 6b).

– **involuntarily:** if members do not receive a *hello_reply* message from their CH after a number of retries of sending *hello* messages (for example 3 times), the member assumes that the CH is disconnected and looks for a new CH.

4.7 Change in affiliation of peer

The peer changes its affiliation to its CH in the two following cases.

– When the peer receives a *cluster_info* message from a new CH in the network, it calculates the weight of this CH. If *W* is greater than that of its current CH, the peer checks if the *cluster_size* of its CH minus 1 (considering that it quits) is lower than the *Maxc* of its CH divided by two, if so, the peer keeps its current CH. Otherwise, the peer immediately leaves its actual CH. We want to minimize the traffic overhead of re-affiliation because when more than one member from the same cluster receive the *cluster_info* message, every one executes the same procedure at the same time and they all quit their actual cluster to go to the other clusters. Therefore, the new cluster becomes overloaded while some others are lightly loaded and traffic overhead of affiliation is augmented. We reduce the rate of re-affiliation and achieve the load
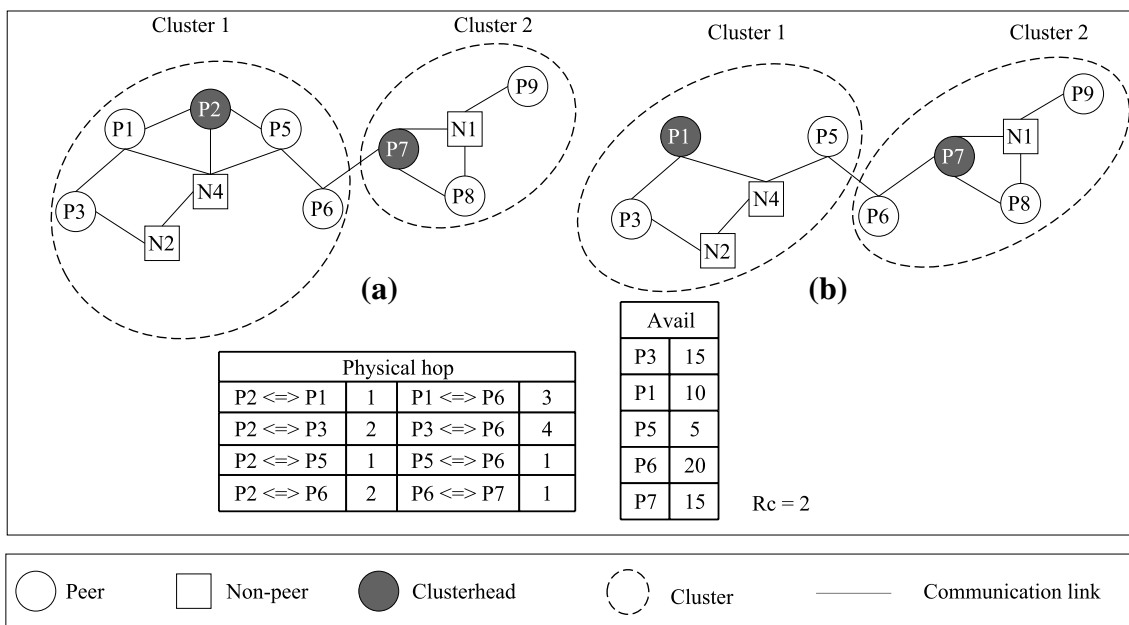

Fig. 6 An example of clusterhead departure

balancing in the case that *cluster_size* is lower than *Maxc* divided by two.

– If the peer receives a *hello_reply* message from its CH and it is at a physical distance greater than *Rc*, it leaves this cluster and searches for a new one.

5 Simulation

We implement our proposed approach (PCSM) in OMNeT++ (Varga and Hornig 2008), a discrete event simulation environment which provides both a P2P model and an ad-hoc 802.11 model. The network area is set to 1500m*1500m with 300 mobile nodes. A random peer quits and a new one joins the P2P overlay every 20s. We define the peer ratio parameter as the number of peers to number of all mobile nodes in the ad-hoc network. The transmission range of a mobile node is 100 m. The mobile nodes move according to the Random Waypoint model that is commonly used to model node mobility in MANET. The maximum moving speed of a peer is set to 2 m/s (speed of a person walking which is reasonable for some types of applications deployed over MANET). The maximum amount of energy of a mobile node is 100 joules. The MANET routing protocol used in this simulation is AODV, proposed by Perkins and Royer (1999), but any other routing protocol can be used. Table 4 shows the most important experimental parameters.

5.1 Performance metrics

In order to evaluate our proposal, we do the following comparisons.

Table 4 Simulation parameters.

Parameter	Value
Simulation area	1500×1500
Transmission range	100 m
Number of mobile nodes	300
MANET routing protocol	AODV
MAC layer	IEEE 802.11
Mobility model	Random Waypoint
Maximum moving speed of a mobile node	2 meter/second
Maximum energy of a peer	100 j
Maximum number of cluster members	10
Minimum number of neighbors	2
Maximum number of neighbors	5
TTL	7
Rc	1–5
p	0.5
q	0.5

1. Performance metrics for neighborhood relationship establishment process and the cluster selection process.

   – The average physical distance between two neighbors and the average number of neighbors per peer versus the peer ratio.
   – The average number of neighbors per peer versus the transmission range.
   – The number of clusters versus the peer ratio and the *Rc* value.
   – The number of clusters versus the transmission range.
   – The average number of members in each cluster versus the peer ratio and the transmission range.

2. Performance metrics to compare PCSM with the proposal of (Shah and Qian 2010a) and the proposal of (Shah et al. 2011).

   To emphasize the advantages of PCSM in the search process, we compare it with O-Cluster proposed by Shah and Qian (2010a) and O-Reactive proposed by Shah et al. (2011). These latter are two important techniques for overlay topology construction in MP2P networks. O-Cluster uses a cluster-based routing protocol (Jiang et al. 1998) and O-Reactive uses the reactive routing protocol AODV as PCSM. We use the same simulations parameters used in O-Cluster and O-Reactive and we fix *Rc* to 1. The lookup process is randomly initiated for a total of 100 random files owned by the peers in the network. We vary the peers ratio and the maximum speed and we use the following metrics for comparison.

   – The average file-discovery delay: it is the average time elapsed from the moment when a file-lookup query is sent to the moment when the first reply is received.
   – The false-negative (FN) ratio: it is the ratio between the numbers of unresolved file-lookup queries for the files that exist in the P2P network and the total number of initiated file-lookup queries.

3. Performance metrics to compare PCSM the proposal of (Noor et al. 2012).

   In order to have a consistent comparison, we implemented the approach proposed by Noor et al. (2012) that we refer to as the energy-based approach, where all the mobile nodes are peers and members are at a single hop from their CHs. We fix *Rc* to 1 and the peer ratio to 100%. We compare PCSM to the energy-based approach, which is a competitive clustering approach for the P2P overlays over MANETs. We analyze the following metrics.

- The number of clusters versus the number of peers.
- Load Balancing: this metric relates to the number of members handled by each cluster. In the best case, this number is the same for all clusters. In order to measure how well balanced the clusters are, we use the same parameter introduced by Mainak et al. (2002), called load balancing factor (LBF):

$$LBF = n_c / \sum_i (x_i - \mu)^2 \qquad (6)$$

where $\mu = (N - n_c)/n_c$, n_c is the number of CHs, μ is the average number of neighbors of a CH, x_i is the cardinality of the cluster i and N the total number of nodes in the system. A higher value of LBF signifies a better load distribution: it tends to infinity for a perfectly balanced system.

- The stability rate: it is the rate of members which remain in their clusters. The stability is decreased when a node moves out from its current cluster and affiliates to another cluster. This metric is calculated by dividing the total count of stable members by the total number of members.
- The re-affiliation rate: it increases when a node gets dissociated from its CH and becomes a member of another CH or becomes CH. This metric is calculated by dividing the total count of re-affiliations by the total number of peers.
- The communication overhead: it measures the total received routing traffic in kilobytes/s and is equal to the total kilobytes received by all peers divided by the simulation time. This overhead includes both cluster construction and maintenance.

5.2 Simulation results and discussion

5.2.1 Results of neighborhood relationship establishment process and PCSM

Figure 7 describes the relationship between the average number of neighbors, the average physical distance between two neighbors and the peer ratio. The average number of neighbors per peer increases and stabilizes when the peer ratio is high. Indeed, the peer receives many replies for its neighborhood requests when the number of peers is high. Additionally, this proves that the calculation of the Rn value according to Eq. 3 is good and adapts to the value of peer ratio. Figure 7 also shows that the average physical distance between two neighbors decreases when the peer ratio increases, because the probability to find the peer in the nearest vicinity increases when there is more peers in the network. In our approach, the peers choose only nearest peers as neighbors.
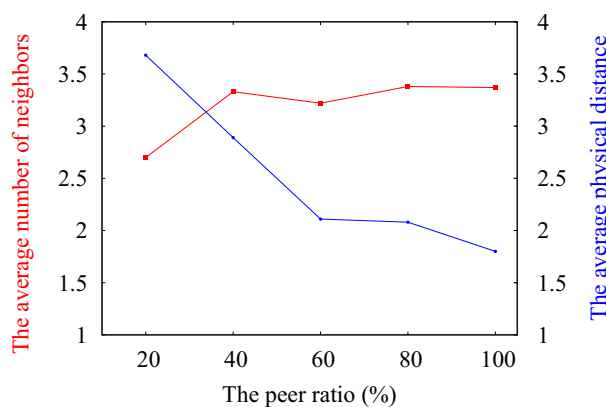

Fig. 7 The average physical distance between two neighbors, the average number of neighbors per peer versus the peer ratio
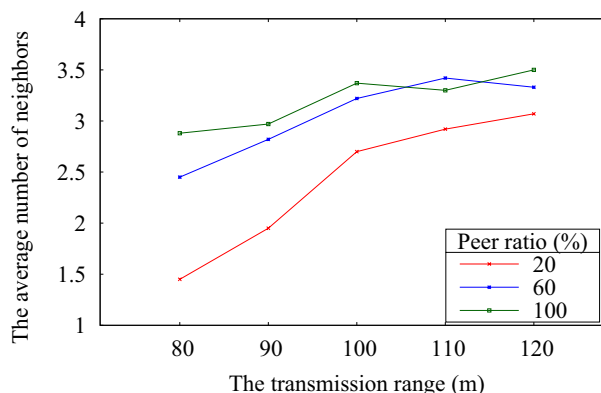

Fig. 8 The average number of neighbors per peer versus transmission range

The variation of the number of neighbors with the transmission range is presented in Fig. 8 with peers ratio of 20, 60, and 100%. Figure 8 shows that the average number of neighbors increases when both the transmission range and the peer ratio increase. The peer with a high transmission range covers a larger area and finds neighbors that are physically closer.

Figure 9 shows the number of created clusters versus the peer ratio and Rc. It shows that the number of created clusters increases when the peer ratio increases for each Rc value. This is reasonable because the maximum number of members in cluster is limited (in this case, is equal to 10): when the peer ratio increases, the number of clusters must increase in order to gather all the peers. However, for each peer ratio value, when the Rc value increases, the number of clusters decreases. This is because when the Rc value increases, the CH covers a larger area. For example, for a peer ratio equal to 20%, when Rc is equal to 1, the number of clusters equals 50. When Rc is equal to 2, the number
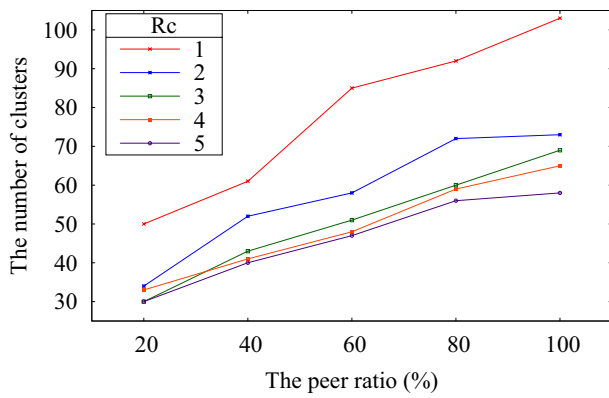
Fig. 9 Number of clusters versus peer ratio and *Rc*

Fig. 11 The average number of members versus peer ratio and transmission range

of clusters is equal to 34 but When *Rc* is equal to 5, the number of clusters is equal to 30.

Figure 10 describes the relationship between the number of created clusters versus the transmission range. Let us consider the case with a value of *Rc* that equals 3. As the transmission range increases, the average number of clusters decreases for each peer ratio. With a higher transmission range, the probability that a peer finds more CHs increases. When the peer chooses one of the existing clusters to join, it prevents the creation of a new cluster. Consequently, the number of formed clusters decreases and the clusters become larger in size as shown in Fig. 11. When the transmission range increases (Fig. 11), the number of clusters decreases and the average number of members increases. Additionally, for each transmission range, when the peer ratio increases the number of clusters decreases because the search area for clusters is bigger and gives the new peers a higher probability to find an existing clusters instead of creating a new one.

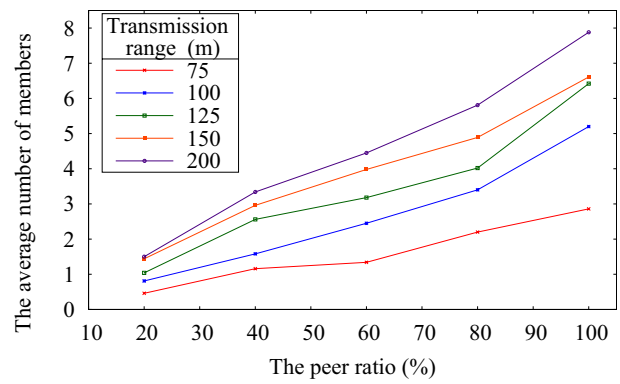**Fig. 10** Number of clusters versus transmission range

5.2.2 Results of comparison of PCSM with O-Cluster (Shah and Qian 2010a) and O-Reactive (Shah et al. 2011)

Figure 12 presents the average file-discovery delay according to the peer ratio and the peer moving speed. We notice that the average file-discovery delay increases for all approaches with the increase of the moving speed. This is because when the peer moving speed increases, the routes break and this leads to several route discovery processes. We also notice that PCSM has the best performance in terms of average file-discovery delay. To perform the search process, PCSM only involves CHs, which have the file index information and quickly finds the searched file, unlike in O-Cluster and O-Reactive where the search process involves all peers. It is also shown that by increasing peer ratio in the network, the average file-discovery delay of O-Cluster and O-Reactive increases. In fact, both O-Cluster and O-Reactive use flooding that causes more traffic overhead with the increase of peer ratio in the network. This leads to big contention delay to access the medium and more packet collisions. In PCSM the delay decreases by increasing peer ratio in the network. This is because the PCSM architecture takes advantages from new peers to enrich the index information of the shared content/files and this increases the probability to find the searched file quickly.

We measure the ratio between sent and lost packets (called FN ratio). Figure 13 shows that the FN ratio increases with the increase of peer moving speed for all approaches. This is because packet loss increases when routes break due to a higher moving speed of peers. We can see that PCSM performs better compared to other approaches for same reasons as previously. In fact, PCSM only involves CHs, which manage the search process, resulting in a decreased packet loss unlike O-Cluster and O-reactive which involve all peers. Regarding the peer ratio, we can see also that PCSM has the best performance in terms of FN ratio while the peer ratio increases compared to the other approaches. This is because
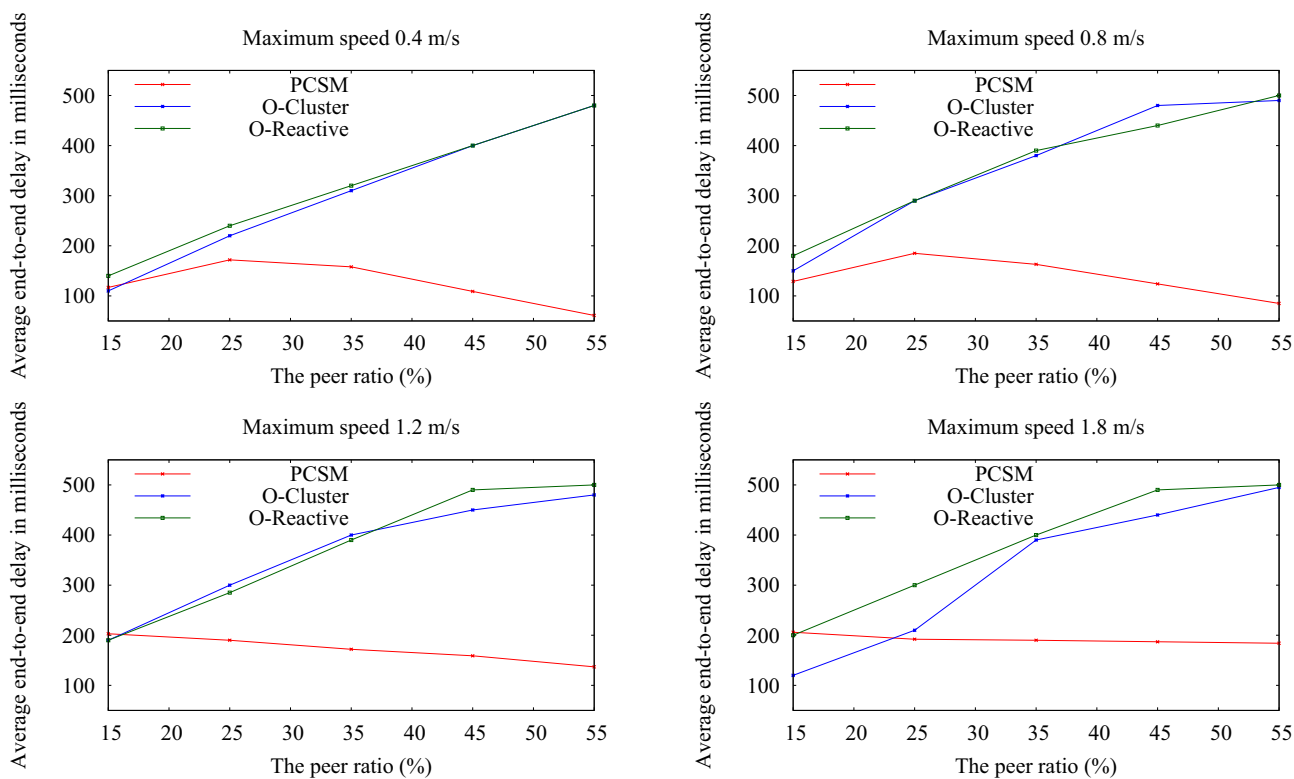
Fig. 12 The comparison of average-file discovery delay between PCSM, O-Cluster and O-Reactive
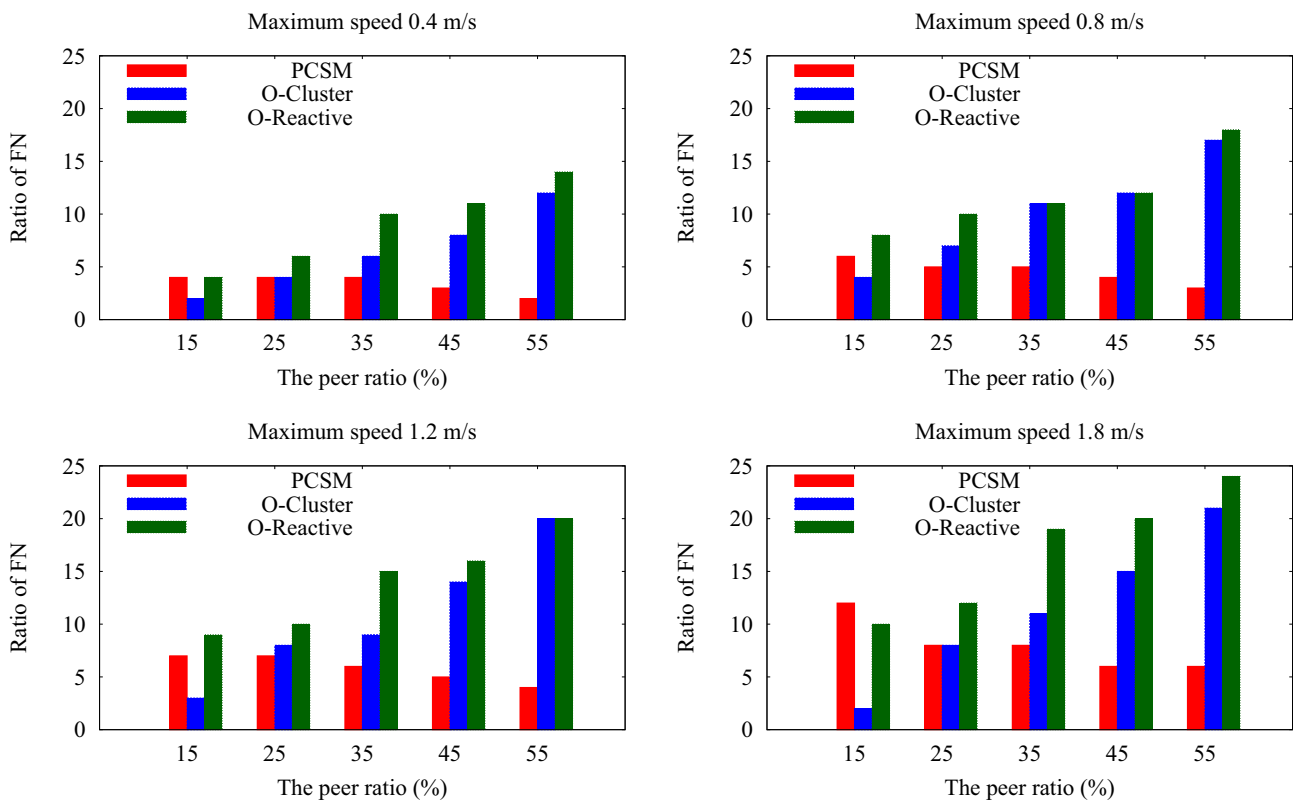

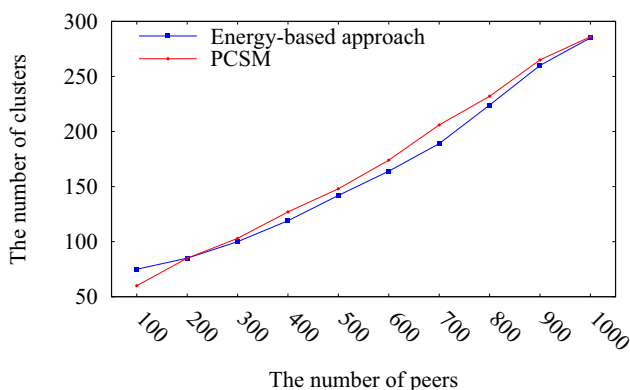**Fig. 13** The comparison of FN ratio between between PCSM, O-Cluster and O-Reactive
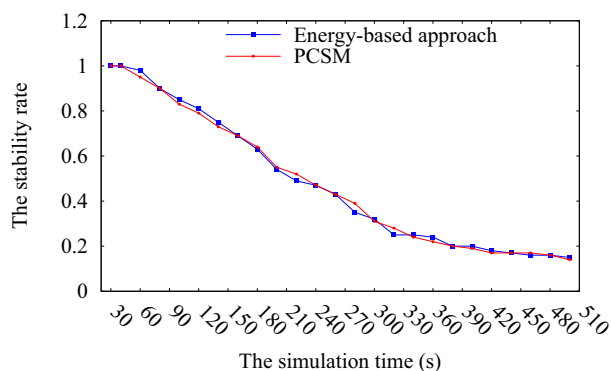
Fig. 14 Number of clusters versus number of peers

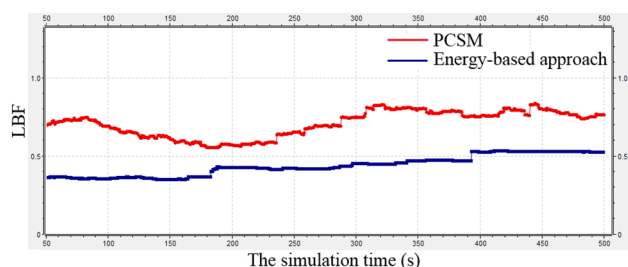Fig. 15 The load distribution in PCSM and the energy-based approach
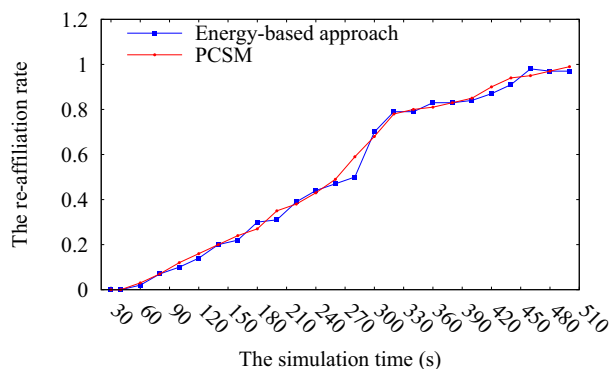

Fig. 16 Stability rate

Fig. 17 Re-affiliation rate

CHs in PCSM use efficient index information which allows finding the searched file at near neighboring peers and this generates less traffic and decreases the packet loss.

5.2.3 Results of comparison of PCSM with the energy-based approach (Noor et al. 2012)

Figure 14 shows the number of clusters for different numbers of peers in the network. Our proposal creates somewhat more clusters than the energy-based approach because PCSM limits the number of members in each cluster at a reasonable number to balance the load between clusters (in this case, the number of members is 10), while in the energy-based approach, each cluster can gather more than 10 peers.

From Fig. 15, the LBF value of PCSM varies between 0.54 and 0.84, whereas that of the energy-based approach varies between 0.34 and 0.54. Our algorithm achieves better load balancing compared to the energy-based approach because in PCSM, we limit the number of members in each cluster and when a new cluster is created, some members quit their actual clusters and join the new ones in order to balance the load.

The stability rate with a transmission range equal to 100 is shown in Fig. 16. From the results, we can see that for

both proposals, the stability values rate decreases and both proposals are close to each others. This is due to the node mobility and the high link failure rates in the network.

Figure 17 shows the re-affiliation rate with a transmission range equal to 100. For both proposals, the re-affiliation rate increases through time because whenever a peer moves, it may disconnect from one cluster and join another cluster, ensuring that each member is always at one hop from its CH.

Figure 16 and Fig. 17 show that either PCSM gives better results than the energy-based approach or inversely. The main reason of this difference is that in PCSM, when a new cluster is created, some members may disconnect from their clusters and join this new one which results in less stability and more re-affiliations. This behavior aims at balancing the load between clusters.

The communication overhead is shown in Fig. 18. The overhead generated for both proposals increases when the speed increases because peers disconnect from their current clusters and join other clusters. Consequently, there are more exchanged messages when the speed increases. However, the generated overhead in the energy-based approach is significant compared to that of PCSM. This is because in the energy-based approach: (1) A CH does not
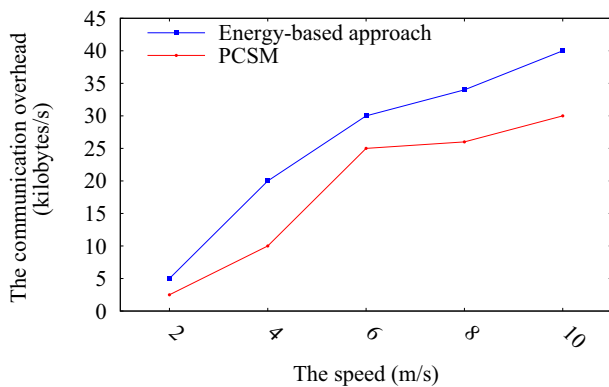
Fig. 18 The communication overhead

send an alive message but broadcasts it. (2) The additional traffic generated by each CH to maintain a secondary CH with a maximum value of energy compared to the other members and to maintain this information up-to-date at all the members. (3) The overhead generated in initializing phase i.e. when the peer receives information about the creation of a new cluster, it must inform the remaining unaffiliated peers about the existence of a new cluster and when these peers receive this information, they broadcast again a *hello* message. In turn, our proposal generates a low overhead because we do not broadcast but rather send the *alive* and *hello* messages. Additionally, our proposal does not use a secondary CH, but maintains the network structure by selecting a new CH among members when necessary.

6 Conclusion and future work

In this paper, we proposed an adaptive overlay topology construction for mobile peer-to-peer (MP2P) systems by considering the physical proximity between peers and structuring them in clusters. Our multihop Proximity aware Clustering Scheme (PCSM) integrates three factors to allow a new peer to efficiently select a cluster to join, including the number of physical hops, the cluster size and the availability of the clusterhead. We also proposed a simple and adaptive maintenance process for clusters and the overlay topology. PCSM achieves better performance than the existing techniques in terms of the average file-discovery delay and false-negative ratio because it confines the lookup process to only CHs. It also gives better results than existing cluster-based P2P overlay in terms of load balancing and routing overhead.

In our future work, we will propose a two complementary solution composed of resource search algorithm and replication strategy to efficiency reduce the traffic overhead.

References

Abid S, Othman M, Shah N (2014) 3D P2P overlay over MANETs. Comput Netw 64:89–111. https://doi.org/10.1016/j.comnet.2014.02.006

Amirazodi N, Saghiri AM, Meybodi M (2018) An adaptive algorithm for super-peer selection considering peer's capacity in mobile peer-to-peer networks based on learning automata. Peer-to-Peer Netw Appl 11:74–89. https://doi.org/10.1007/s12083-016-0503-y

Elgazzar K, Ibrahim W, Oteafy S, Hassanein HS (2013) RobP2P: a robust architecture for resource sharing in mobile peer-to-peer networks. In: Proc of the 4th Int Conf on ambient systems, networks and technologies, Elsevier, Halifax Nova Scotia, Canada, pp 356–363. https://doi.org/10.1016/j.procs.2013.06.049

Han JS, Lee KJ, Song JW, Yang SB (2008a) Mobile peer-to-peer systems using super peers for mobile environments. In: Proc of 2008 Int Conf on information networking, IEEE, Busan, South Korea, pp 1–4. https://doi.org/10.1109/ICOIN.2008.4472831

Han JS, Song JW, Kim TH, Yang SB (2008b) Double-layered mobile p2p systems using energy-efficient routing schemes. In: Proc of the telecommunication networks and applications Conf, IEEE, Adelaide, SA, Australia, pp 122–127. https://doi.org/10.1109/ATNAC.2008.4783308

da Hora DN, Macedo DF, Oliveira LB, Siqueira IG, Loureiro AA, Nogueira JM, Pujolle G (2009) Enhancing peer-to-peer content discovery techniques over mobile ad hoc networks. Comput Commun 32:1445–1459. https://doi.org/10.1016/j.comcom.2009.04.005

Hu CL, Kuo TH (2012) A hierarchical overlay with cluster-based reputation tree for dynamic peer-to-peer systems. Netw Comput Appl 35:1990–2002. https://doi.org/10.1016/j.jnca.2012.07.022

Jiang M, Li J, Tay YC (1998) Cluster based routing protocol (CBRP) Functional specification. Tech. rep., https://www.ietf.org/proceedings/43/I-D/draft-ietf-manet-cbrp-spec-00.txt

Kim JH, Song JW, Kim TH, Yang SB (2011) An enhanced double-layered P2P system for the reliability in dynamic mobile environments. Comput Inform 30:467–489

Kumar R, Kumar P (2012) Replica allocation techniques based on clusters for MANETs. In: Proc of the Int Conf on emerging trends in computer and electronics engineering, Dubai, pp 24–25

Li H, Bok K, Park Y, Yoo J (2011) An efficient mobile peer to peer architecture in wireless Ad Hoc network. In: Proc of the Int Conf on hybrid information technology, Springer, New York, pp 1–8. https://doi.org/10.1007/978-3-642-24082-9-1

Liang JC, Chen JC, Zhang T (2011) An adaptive low-overhead resource discovery protocol for mobile ad-hoc networks. Wirel Netw 17:437–452. https://doi.org/10.1007/s11276-010-0289-4

Mainak C, Das SK, Turgut D (2002) WCA: a weighted clustering algorithm for mobile Ad Hoc networks. Clust Comput 5:193–204. https://doi.org/10.1023/A:1013941929408

Mawji A, Hassaneina H, Zhang X (2011) Peer-to-peer overlay topology control for mobile ad hoc networks. Pervasive Mobile Comput 7:467–478. https://doi.org/10.1016/j.pmcj.2011.04.003

Mondal A, Madria SK, Kitsuregawa M (2006) EcoRep: an economic model for efficient dynamic replication in mobile-P2P networks. In: Proc of the 13th Int Conf on management of data, New Delhi, India, pp 116–127

Nakahara S, Ohta T, Kakuda Y (2014) A routing scheme based on autonomous clustering and P2P overlay network in MANETs. In: Proc of 2014 IEEE 11th Intl Conf on ubiquitous intelligence and computing and 2014 IEEE 11th Intl Conf on Autonomic and Trusted Computing and 2014 IEEE 14th Intl Conf on scalable computing and communications and its associated workshops, IEEE, pp 518–524. https://doi.org/10.1109/UIC-ATC-ScalCom.2014.18

Noor TB, Salehin MR, Islam SR (2012) A clustering scheme for peer-to-peer file searching in mobile Ad Hoc networks. Int J Adv Res Comput Commun Eng 1:487–493

Papadakis H, Fragopoulou P, Markatos EP, Roussopoulos M (2013) ITA: innocuous topology awareness for unstructured P2P networks. IEEE Trans Parallel Distrib Syst 24:1589–1601. https://doi.org/10.1109/TPDS.2012.137

Perkins CE, Royer EM (1999) Ad-hoc on-demand distance vector routing. In: Proc of 2nd workshop on mobile computing systems and applications, IEEE, New Orleans, LA, USA, pp 90–100. https://doi.org/10.1109/MCSA.1999.749281

Rahmani M, Benchaïba M (2017a) An efficient replication scheme to increase file availability in mobile peer to peer systems. In: Proc of the 2017 Int symposium on networks, computers and communications, IEEE, pp 1–6. https://doi.org/10.1109/ISNCC.2017.8072021

Rahmani M, Benchaïba M (2017b) Toward a resource availability measurement in peer to peer systems. In: Proc of the 5th Int Conf on networked systems, Springer, New York, pp 248–252. https://doi.org/10.1007/978-3-319-59647-1-19

Ratnasamy S, Francis P, Handley M, Karp R, Shenker S (2001) A scalable content-addressable network. In: Proc of the 2001 Conf on applications, technologies, architectures, and protocols for computer communications, ACM, San Diego, California, USA, pp 161–172. https://doi.org/10.1145/383059.383072

Ripeanu M, Iamnitchi A, Foster I (2002) Mapping the gnutella network. IEEE Internet Comput 6:50–57. https://doi.org/10.1109/4236.978369

Saghiri AM, Meybodi MR (2017) A distributed adaptive landmark clustering algorithm based on mOverlay and learning automata for topology mismatch problem in unstructured peer-to-peer networks. Int J Commun Syst. https://doi.org/10.1002/dac.2977.

Seddiki M, Benchaïba M (2015) An adaptive P2P overlay for MANETs. In: Proc of the Int Conf on distributed computing and networking, ACM, pp 1–8. https://doi.org/10.1145/2684464.2684496

Seddiki M, Benchaïba M (2016) 2P-lookup: popularity and proximity based P2P lookup mechanism over MANETs. J Netw Comput Appl 71:181–193. https://doi.org/10.1016/j.jnca.2016.03.017

Shah B, Kim KI (2014) Towards enhanced searching architecture for unstructured peer-to-peer over mobile Ad Hoc networks. Wirel Pers Commun 77:1167–1189. https://doi.org/10.1007/s11277-013-1560-7

Shah N, Qian D (2010a) An Efficient overlay for unstructured P2P file sharing over MANET using underlying cluster-based routing. KSII Trans Internet Inform Syst 4:799–818. https://doi.org/10.3837/tiis.2010.006

Shah N, Qian D (2010b) An efficient structured P2P overlay over MANET. In: Proc of Ninth ACM int workshop on data engineering for wireless and mobile access, ACM, Indianapolis, Indiana, pp 57–64. https://doi.org/10.1145/1850822.1850832

Shah N, Qian DP, Wang R (2011) An efficient unstructured P2P overlay for file sharing over MANETs using underlying reactive routing. J Internet Technol 12:517–530

Shah N, Ahmad A, Mehmood W, Qian D, Wang R (2014) An efficient and scalable routing for MANETs. Wirel Pers Commun 75:987–1004. https://doi.org/10.1007/s11277-013-1403-6

Shah N, Qian D, Wang R (2015) Merging of P2P overlays over mobile ad hoc network: evaluation of three approaches. Ad Hoc Sens Wirel Netw 25:199–222

Shah N, Ahmad A, Nazir B, Qian D (2016) A cross-layer approach for partition detection at overlay layer for structured P2P in MANETs. Peer-to-Peer Netw Appl 9:356–371. https://doi.org/10.1007/s12083-015-0341-3

Shah N, Abida S, Qian D, Mehmood W (2017) A survey of P2P content sharing in MANETs. Comput Electr Eng 57:55–68. https://doi.org/10.1016/j.compeleceng.2016.12.013

Stoica I, Morris R, Karger D, Kaashoek MF, Balakrishnan H (2001) Chord: scalable peer-to-peer lookup service for internet applications. In: Proc of the 2001 Conf on applications, technologies, architectures, and protocols for computer communications, IEEE/ACM, San Diego, California, USA, pp 149–160. https://doi.org/10.1145/383059.383071

Varga A, Hornig R (2008) An overview of the omnet++ simulation environment. In: Proc of the 1st Int Conf on Simulation tools and techniques for communications, networks and systems & workshops, ICST/ACM, Marseille, France, pp 1–10

Zhang K, Antonopoulos N (2013) A novel bartering exchange ring based incentive mechanism for peer-to-peer systems. Future Gener Comput Syst 29:361–369. https://doi.org/10.1016/j.future.2011.06.005