Chapter 6

Mobile P2P: Peer-to-Peer Systems over Delay Tolerant Networks

Angela Sara Cacciapuoti, Marcello Caleffi, and Luigi Paura

6.1 Introduction

P2P systems are distributed systems able to form self-organizing overlay networks to provide efficient search/distribution of data items [1]. By introducing the concept of *peer*, i.e., an entity that provides and, at the same time, consumes resources/services offered by others entities, P2P systems go beyond the traditional client/server paradigm thanks to its features of self-organization, fault-tolerance, and high scalability. In the last years, the P2P paradigm has gained popularity as a consequence of the diffusion of Internet file sharing applications like Napster [2], Gnutella [3], and Emule [4], which have allowed millions of users to share files in a decentralized manner.

On the other hand, DTNs represent a novel paradigm for wireless multi-hop networks that aims to provide connectivity also when links on an end-to-end path may not exist contemporaneously and therefore intermediate nodes may need to store data waiting for communication opportunities [5].

As pointed out in [6], DTNs and P2P systems share the same key concepts of self-organization and distributing computing, and both aim to work in a completely decentralized environment. Both lack central entities to which to delegate the management and the coordination of the network, and both rely on a time-variant topology. In fact, in P2P networks the time-variability is due to joining/leaving peers, while in DTN ones it is due to both node mobility and wireless propagation condition instability. Finally, both adopt a *store-and-forward* like paradigm: DTN nodes store packets waiting for a chance to deliver them to the destinations, while peers store data items waiting for requests from other peers.

Despite these similarities, the adoption of the P2P paradigm to disseminate and discover information in a DTN raises to new and challenging problems [7, 8]. One of the main issues concerns the layer where they operate. P2P systems build and maintain overlay networks at the application-layer, assuming the presence of an underlying network layer which assures connectivity among nodes. DTNs focus on providing a multi-hop wireless connectivity among nodes in scenarios where frequent and numerous network partitions would prevent packets from being delivered in a timely fashion. In addition, traditional P2P systems rely on wired infrastructures, characterized by reliable and bandwidth-supplied links. On the other hand, DTNs rely on unreliable and bandwidth-limited wireless links. Finally, DTN nodes usually have hard constraints on the available resources like energy, memory, and computation, while peers are commonly assumed to be resourceful.

For these reasons, trying to couple a P2P overlay network over a DTN is still an open problem. Neverthless, since the ad hoc network paradigm, which assumes that most of the time an end-to-end connectivity between each pair of nodes exists, can be considered as a special case of DTN [5], we start looking at the P2P solutions proposed for MANETs. However, since DTNs share with P2P systems a store-and-forward like paradigm which requires a unitary approach able to assure the effectiveness of integrated solutions, we describe some interesting examples of such an integrated approach.

The remaining part of the chapter is organized as follows. In Sec. 9.2 an overview of P2P overlay networks is provided, by distinguishing the unstructured P2Ps from the structured ones. Sec. 9.3 describes the main features and applications of DTNs. The store-carry-forward paradigm is presented and it is highlighted as MANETs can be considered as a sub-class of DTNs. In Sec. 9.4 we provide the main challenging issues for the design of P2P overlay networks in a DTN, and the main features of some representative Mobile P2P (MP2P) systems are discussed. This section is concluded with the open problems. Finally, conclusions are drawn in Sec. 9.4.

6.2 Peer-to-Peer Overlay Networks

Peer-to-Peer (P2P) systems represent a recently proposed scalable and fault-tolerant paradigm to disseminate and discover information in a communication network. In this section, we provide an overview of such a paradigm in terms of the main characteristics, some interesting applications, and finally some illustrative examples.

6.2.1 Overview

The Peer-to-Peer (P2P) paradigm is an application-level paradigm that aims to share both resources and services and in which the involved entities, namely the *peers*, behave both as resource/service consumers and providers. Such a paradigm assumes that the peers collaborate spontaneously by means of distribute procedures without the necessity of establishing a hierarchy and/or relying on a pre-existent infrastructure. Differently from the traditional client/server paradigm, the lack of hierarchy guarantees the absence of bottlenecks and of single point of failures, allowing the P2P paradigm to exhibit properties of fault-tolerance and high scalability.

The P2P paradigm does not deal with the communication issues, since it assumes the presence of an underlying layer which assures connectivity among nodes. In this sense, it defines an overlay network, i.e. a logical network built on top of the physical one. Fig. 6.1 shows an example of an overlay network built upon a physical one.

We note that usually the logical proximity, i.e. the proximity in the overlay network among peers, is not related to the physical one, namely the proximity in the physical topology. Therefore two neighboring peers in the overlay network are likely not to be neighbors in the physical one and so one logical hop usually involves multiple physical hops. Moreover, although traditional

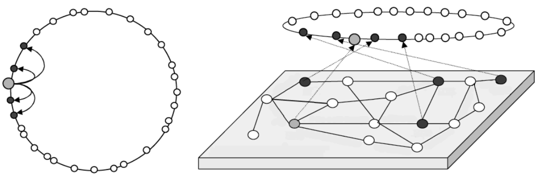

Figure 6.1: Overlay network

P2P systems are commonly developed over static networks, the related overlay networks are characterized by time-variant topologies due to the peer joining/leaving.

P2P systems can be grouped in two classes, namely *unstructured* and *structured* systems, according to the solution adopted for the content dissemination/discovery. More specifically, in unstructured P2Ps, peers are unaware of the resources that neighboring peers in the overlay network maintain. So, they typically resolve search requests by means of flooding techniques and they rely on resource replication to improve the lookup performance and reliability. Differently, in structured P2P networks peers have knowledge about the resources offered by overlay neighbors, usually by resorting to the Distributed Hash Table (DHT) paradigm, and, therefore, the search requests are forwarded by means of unicast communications. We note that in both the approaches the locations where data items have to be stored are selected regardless of the physical topology of the network.

In the following paragraphs we present the main features of each approach, along with some illustrative examples.

6.2.2 Structured Peer-to-Peer Overlay Networks

The adoption of a structured approach for content dissemination/discovery imposes that data items be placed at specific peers according to a globally known rule. In this way, the items can be efficiently retrieved with unicast communications, thus avoiding the inefficiency of flooding techniques adopted in unstructured systems.

Usually, structured P2P systems utilize as data structures the Distributed Hash Tables (DHTs), which allow retrieval of data items without the need of a-priori knowledge about the locations where the items are stored. More specifically, each peer has a unique identifier, namely a *peer id*, belonging to the identifier space I and each data item is univocally identified by a *key* belonging to the key space K.

The core of a DHT is a globally known hash function $h : K \rightarrow I$ able to map data items on live peers. By means of the hash function, a P2P system

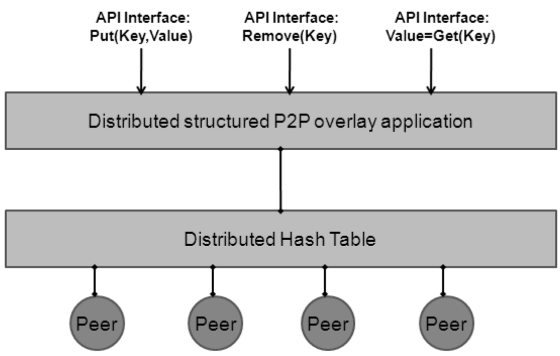

Figure 6.2: Application interface for structured DHT-based P2P systems

is able to provide a scalable storage and retrieval of data items in the overlay network by means of three common interfaces: *put*, *remove*, and *get*, as shown in Fig. 6.2 derived from [1]. Given the data item and the corresponding key, the put operation *put(key,value)* stores the data item *value* at the peer whose identifier is equal to $h(key)$. The remove operation *remove(key)* simply removes from the hash table the data item corresponding to the key. Finally, the lookup operation *get(key)* retrieves the data item corresponding to the key.

Structured P2P systems require that each peer maintains a table which stores, for each logical neighbor peer, both its identifier and its Internet Protocol (IP) address. The communication among peers exploits the overlay neighborhood: when a peer has to send a message to another one, it forwards the message to the neighbor peer whose identifier is the closest to the destination one according to a certain metric (e.g., numerically closest, shortest Euclidean distance, etc.). In such a way, structured P2P systems impose a *structure* on the overlay network topology, and the defined structure depends on the particular P2P protocol. Typical structures are the ring, the tree, and the butterfly.

In theory, DHT-based systems can guarantee that each data item can be located/retrieved in $O(\log n)$ overlay hops, where n is the number of peers in the system. Since the underlying network path between two neighbor peers can be composed by several physical links, the latency times for data items, disseminate/discovery can be quite long and can affect the overall performances. Moreover, the table maintenance can introduce a considerable overhead.

Widely-known examples of file-sharing applications based on structured P2P overlay networks are Content Addressable Network (CAN) [9], Tapestry [10], Chord [11], Pastry [12], Kademlia [13], and Viceroy [14].

6.2.3 Unstructured Peer-to-Peer Overlay Networks

Unstructured P2P systems do not impose that content has to be placed at a pre-defined peer. In other words, they do not impose a pre-defined structure on the overlay network and, therefore, the peers have to resort to strategies like flooding, random walks, or expanding-ring search to discover the data items.

From an operational point of view, when a peer receives a resource query it first locally evaluates the query on its own data items. Then, it replies to the requesting peer with a list of the owned data items corresponding to the query. Such a strategy is easy to implement and, moreover, it natively supports complex keyword-based queries.

Nevertheless, the lack of relation between a data item and its location implies scalability issues. The strategy, in fact, is effective in case of widely replicated data items, while in case of rare contents the queries have to be sent to a large set of peers, thus incurring a considerable overhead. On the other hand, structured P2P systems are able to efficiently locate rare items, but they incur significant overhead in discovering popular content.

Unstructured P2P systems can be classified in three main groups according to the adopted architecture: centralized, de-centralized, and hybrid, as shown in Fig. 6.3. In centralized P2P systems some functions are provided by a central entity, which coordinates and provides auxiliary information to peers. Nevertheless, peers communicate directly without any intermediate entity. The advantages of P2P centralized systems are easy management and implementation of security policies. On the other hand, the presence of the central entity limits the scalability and introduces single points of failures. Examples of file-sharing applications based on centralized P2P architectures are Napster [2] and SETI@home project [15]. Decentralized P2P systems, like Freenet [16] and Gnutella [3], are based on a flat peer hierarchy where all

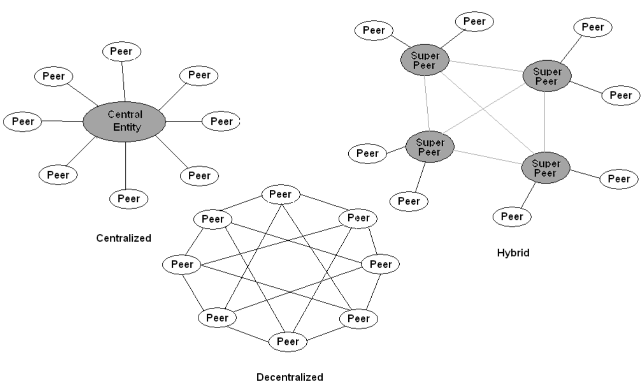

Figure 6.3: Unstructured P2P architectures

Table 6.1: Characteristics of Different Unstructured P2P Architectures

Characteristics	Architectures: Centralized	Decentralized	Hybrid
Manageable	Yes	No	No
Extensible	No	Yes	Yes
Fault-tolerance	No	Yes	Yes
Secure	Yes	No	No
Scalable	No	Maybe	Maybe

peers share the same role. As advantages, the decentralized systems are scalable and they exhibit fault tolerance properties. Finally, hybrid P2P systems try to conjugate both the advantages of centralized and decentralized architectures. In such systems, peers are organized in clusters, and the cluster-heads, namely the *super-peers*, are responsible for forwarding queries received by the peers. The communications among super-peers are decentralized since hybrid P2P systems adopt a two-level hierarchy. Examples of hybrid architecture are KazaA [17] and Morpheus [18]. Table 6.1, derived from [19], summarizes the main characteristics for each architecture.

6.3 Delay Tolerant Networks

Disruption or Delay Tolerant Networks (DTNs) are an emerging class of networks in which the assumption of a persistent end-to-end path between each pair of nodes is relaxed. DTNs are characterized by the following features [20]:

- intermittent connectivity: a DTN exhibits a weak, episodic connectivity as a consequence of unstable end-to-end paths;
- unpredictable end-to-end delays: as a consequence of intermittent connectivity, the end-to-end delays can exceed the requirements of real-time applications or protocols that rely on quick return of acknowledgments or data;
- asymmetric communications: communications exhibit asymmetric characteristics (data rates, loss rates, delays, etc.);
- unreliable communications: DTN routes are characterized by unreliable communications, and end-to-end Automatic Repeat Request (ARQ) strategies cannot be adopted in presence of long delays.

In addition, DTN nodes can have strong resource limitation (power, storage and computation), especially in case of mobile networks.

Several military and civilian applications can benefit from the DTN paradigm. Typical examples are the deep-space networks such as the NASA

JPL's Deep Impact Networking (DINET) [21] (where the delay/disruption tolerance is required due to long delays and high packet loss of the interplanetary communications), networks for satellite communications [22], and networks for rural areas such as Kiosknet[23].

6.3.1 The Store-Carry-Forward Paradigm

A typical networking paradigm for DTNs characterized by sparse topologies is the *store-carry-forward* one, which assumes that a message is stored and carried by the nodes, until an opportunity to deliver the message arises. An example of the process is shown in Fig. 6.4 through Fig. 6.6. More specifically, in Fig. 6.4 node S has to communicate with D, but there is not any connected path between the two nodes. Therefore, S has to store the message, waiting for a communication opportunity. The node R acts as a relay for S by carrying the message as shown in Fig. 6.5, and forwarding it to D as depicted in Fig. 6.6.

The routing protocols which adopt the store-carry-forward paradigm can be classified in two main groups, according to the assumptions made about the available knowledge of the network topology [24].

The first class of protocols requires a minimal knowledge about the topology. In such a case, the simplest delivery strategy is to replicate the messages in the network. In more detail, the source forwards a copy of the message each time that another node comes into its communication range. The same procedure is followed by the receiving node, by forwarding copies of the same message to nodes which in turn come in contact with it. Clearly, this implies that several copies of the same message are present in the network, wasting the resources. This strategy is the basic idea behind *Epidemic Routing* protocols [25, 26, 27, 28], which try to solve the scalability issues by adopting some limitations on the message replication, i.e., by limiting the number of copies for each message or by using historical encounter-based metrics.

The second class of protocols, namely the *message ferrying ones*, assumes the presence of well-connected islands of nodes that intermittently communicate each with other thanks to node mobility [29, 30, 31, 32]. The mobile nodes responsible for carrying the messages among the islands are called *ferries*. Differently from epidemic routing-like protocols, the message ferrying ones adopt single copy forwarding strategies.

6.3.2 MANETs as a Special Case of DTNs

As mentioned before, DTNs are a class of wireless networks that also aim to provide connectivity in the absence of a persistent end-to-end path between nodes, usually by requiring that relays store the messages waiting for connectivity. On the other hand, ad hoc networks are wireless networks in which a persistent end-to-end connectivity between each pair of nodes exists. In these networks it is assumed that if a path fails due to node mobility and/or

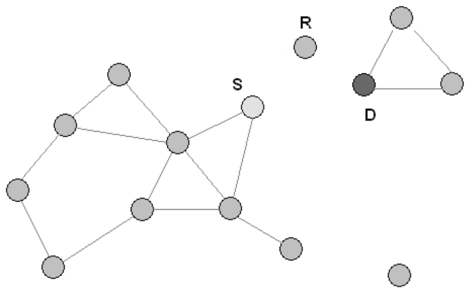

Figure 6.4: S stores the message waiting for a communication opportunity.

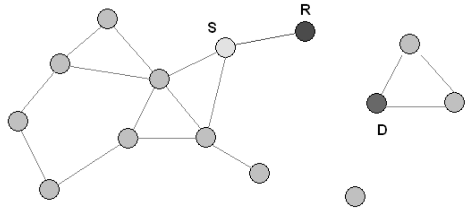

Figure 6.5: S forwards the packet to R.

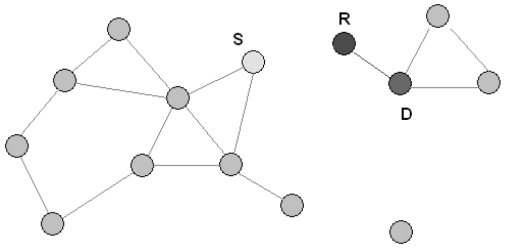

Figure 6.6: R carries the message, and forwards it to D.

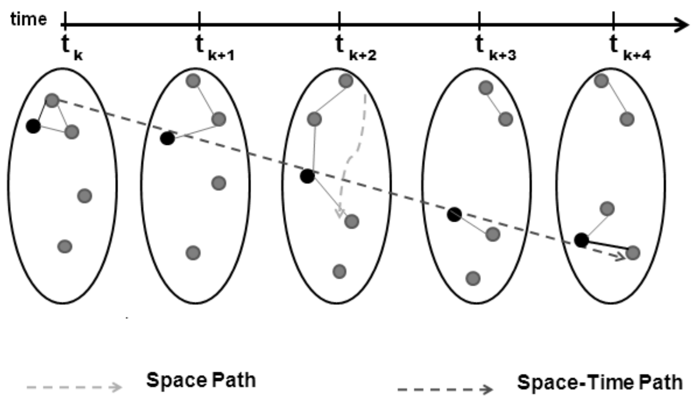

Figure 6.7: Example of a DTN space-time path.

wireless propagation conditions, instability, such a failure is temporary since alternative routes soon become available.

According to [5], we define the un-persistent paths of DTNs as *space-time paths* to underline that not all the links belonging to the path exist simultaneously, while ad hoc paths are referred to as *space paths*. Fig. 6.7 derived from [5] shows an example of a path in which the links appear at different temporal intervals.

According to this classification, space paths are a special case of space-time paths, and therefore ad hoc networks can be considered as a sub-class of DTNs. Clearly, since Mobile Ad hoc NETworks (MANETs) are a special case of ad hoc networks in which nodes are mobile, they are also a special case of DTNs. This classification is depicted in Fig. 6.8.

In the following, we will use such a classification to distinguish P2P systems proposed for DTNs from those proposed for MANETs and to analyze the limits of the latter ones when they are applied on mobile DTNs.

6.4 Mobile Peer-to-Peer Overlay Networks for Delay Tolerant Networks

In this section we first present the main challenging issues for the design of P2P overlay networks in a DTN, and then we describe the main features of some representative Mobile P2P (MP2P) systems, that is P2P systems for mobile multi-hop wireless networks. The considered proposals have been selected according to one or more of the following motivations: i) they may be popular choices among the research community; ii) they may be illustrative examples of interesting approaches; iii) they may have unique features that

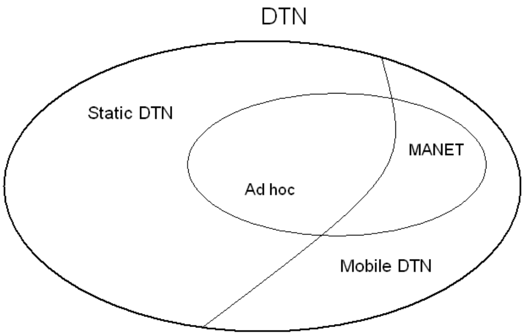

Figure 6.8: Classification of wireless multi-hop networks

make them appealing. In the following no comparison is carried out among the considered overlay networks, since their citations often provide performance evaluations of the systems.

6.4.1 Challenges

Providing an efficient architecture for information dissemination/discovery in DTNs is an open problem, since there are several challenging issues related to the sparse topologies and the intermittent connectivity.

DTNs operate with a smaller bandwidth than MANETs since in space-time paths links are available at different temporal intervals. Clearly, this implies that it is necessary to adopt solutions that avoid high overlay maintenance traffic (common in structured P2P overlay networks) or inefficient flooding-based searches (common in unstructured ones) to make them suitable for DTNs.

Moreover, the typical unpredictable delays of DTNs affect the information dissemination/discovery procedures. Some redundancy in queries/content forwarding is necessary to compensate the unreliability of wireless communications. Nevertheless, congestion due to excessive query/content messaging has to be avoided.

As mentioned in Sec. 6.2.1, traditional P2P systems exploit a logical proximity among peers that is not related to the physical one. In more detail, messages are routed among peers which are neighbors in the overlay network, but, since two logically neighboring peers are likely not to be neighbors in the physical one, each logical hop usually involves multiple physical hops, thus introducing a considerable overlay route stretch effect. Mobile P2P systems for DTNs should implement a kind of relation between the overlay and the physical topology to avoid such a route stretch effect.

Finally, implementing security policies in decentralized, self-organizing,

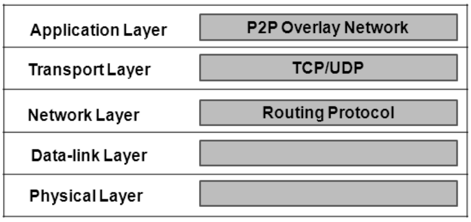

Figure 6.9: Traditional layered approach

and anonymous systems like the P2P ones is a complex task, which becomes harder in MP2P due to the node mobility and broadcast characteristics of wireless communications.

These challenging issues require new approaches to provide scalable MP2P systems [33]. It has been proved that the traditional layered approach (Fig. 6.9), namely simply deploying a P2P overlay network over an unreliable network substrate, causes significant message overhead and redundancy due to the lack of cooperation and communication between the two layers. For these reasons, several proposals exploiting the cross-layer approach (Fig. 6.10) have been presented in the last years. In these systems an inter-layer communication between the network and the application layers is introduced, thus allowing a weak interaction between the routing and the P2P functionalities. However, very recent solutions [34, 6, 35] which integrate the P2P services at the network layer have been proposed (Fig. 6.11), thus allowing a more strong interaction between the two layers.

6.4.2 Unstructured Mobile Peer-to-Peer Overlay Networks

Unstructured MP2P overlay networks generally provide flooding-based content discovery using reactive routing protocols as network substrate.

In the following, we describe five unstructured MP2P systems: the Optimized Routing Independent Overlay Network (ORION) [36], the Mobile

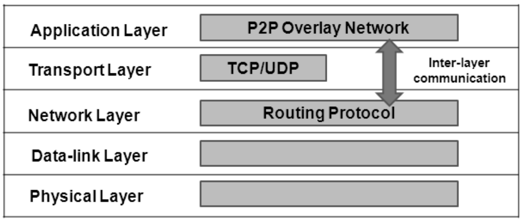

Figure 6.10: Cross-layered approach

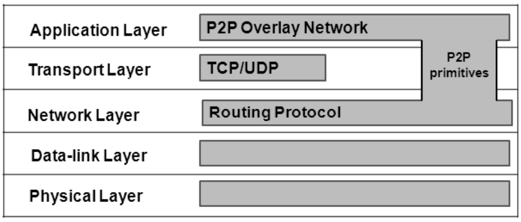

Figure 6.11: Integrated approach

Peer-to-Peer (MPP) [37], the Ad-hoc Storage Overlay System (ASOS) [38], the Peer-to-Peer file sharing system based on Swarm Intelligence (P2PSI) [39], and a Prophet-based information retrieval system [40]. A brief discussion about the limitations of each system is also provided.

6.4.2.1 Optimized Routing Independent Overlay Network

The Optimized Routing Independent Overlay Network (ORION) [36] offers file-sharing services over MANET scenarios with a layered approach. ORION provides advanced keyword-based content discovery using as network substrate the Ad hoc On demand Distance Vector (AODV) [41] and a Gnutella-like overlay network.

Each node maintains a list of the data items locally stored and an AODV-like routing table for the reverse paths. When a node needs a data item, it floods the network with a query message. Each node that receives the query first stores in the routing table the reverse path towards the source node, i.e. it stores as next hop toward the source the node that has forwarded the query. Then, it broadcasts the query to the (physical) neighbor nodes. Finally, it looks in the local data item list for content matching the query and replies with a query reply message in the event of success.

To reduce the overhead of the discovery process, ORION adopts reduced query replies. The intermediate nodes belonging to the path of a query reply avoid forwarding messages for already discovered data items.

Once the query source acquires the knowledge about the nodes that store the data item, it splits the data item in equal length blocks and sends a data request message for each block toward one of the storing nodes. When the data request is received by the storing node, it replies with a data reply message which contains the requested block and which follows the same reverse path discovered during the query process.

The adoption of ORION in DTNs poses several issues. The main problem is due to the assumption about the persistence of reverse paths, which is clearly unrealistic in DTNs. Moreover, the flooding-like strategy adopted for content discovery is not suitable in bandwidth-limited environments.

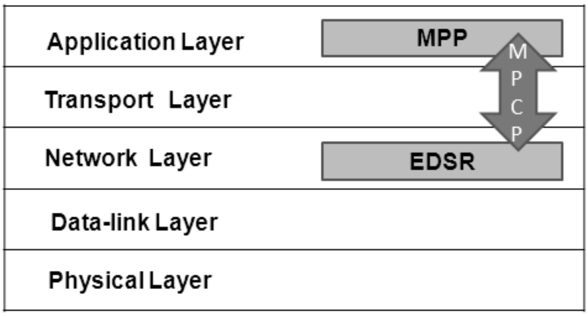

Figure 6.12: MPP cross-layer architecture.

6.4.2.2 Mobile Peer-to-Peer

The Mobile Peer-to-Peer (MPP) [37] offers MANET file-sharing functionalities with a cross-layered approach, by combining MANET routing with flooding-based content discovery. In more detail, MPP uses as network substrate a modified Dynamic Source Routing (DSR) protocol [42], namely the Enhanced Dynamic Source Routing (EDSR). The inter-layer communication between network and application layer is provided by Mobile Peer Control Protocol (MPCP), which allows the P2P application to register itself in the EDSR layer, as shown in Fig. 6.12. In such a way, the application can initialize search requests and it can process incoming requests from other nodes.

On startup, the P2P application on the mobile node announces itself to the EDSR layer via MPCP. When a node has to access to a data item, MPCP forwards the P2P application request to EDSR, which in turn transforms it into a search request. Similar to DSR route requests, EDSR floods the search request through the network and when a node receives the request via the EDSR substrate, it forwards such a request to the P2P application via MPCP. Thus the application layer can determine if any locally stored data item satisfies the request criteria. If so, the application layer initializes an EDSR data reply, which is sent back to the originating node and contains all necessary information for the data item transfer. Similar to DSR route replies, a data reply includes the complete path between source and destination.

MPP adopts the Hyper Text Transfer Protocol (HTTP) for the data item transfers between peers. Moreover, MPP specifies additional features to overcome the connection break events by allowing peers to continue the transfer from the last received byte.

Besides the adoption of a cross-layer approach, MPP shares the same limitations of ORION when applied to DTNs: flooding inefficiency and persistent path assumption. In particular, the inefficiency of MPP in case of un-persistent paths is made worse by its source routing nature. In fact, the complete ordered list of nodes through which the packets have to pass is singled out at the source side.

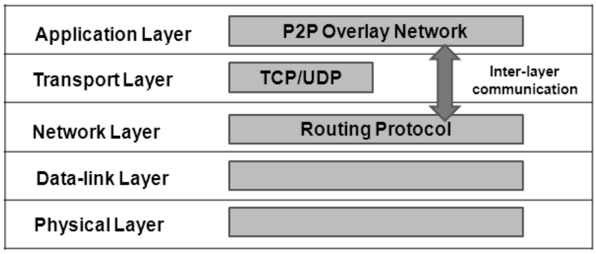

Figure 6.13: P2PSI cross-layer architecture.

6.4.2.3 Ad-Hoc Storage Overlay System

The Ad-hoc Storage Overlay System (ASOS) [38] is a self-organized P2P system specifically designed for MANETs. Nevertheless, the proposed approach is suitable for DTN scenarios, since it tolerates disruption-prone communications.

ASOS assumes the existence of nodes with high memory capabilities, namely ASOS agents, which are exploited to provide reliable communications over unreliable paths. In case of link failures, the ASOS agents cache the data items and deliver them to the original destinations when the connectivity is restored.

From an operational point of view, after the source node submits a data item to its ASOS agent, it becomes the first ASOS peer to hold a copy of such an item. To increase storage reliability, the item is also replicated to other ASOS agents, by selecting among the reachable neighbors of the agent $K-1$ locations to replicate the data to (where K is a configurable parameter). With the assumption that pairwise distances between nodes can be measured, storage locations are selected based on three guidelines: distance from the destination, distance from other ASOS agents, and load of the agent.

ASOS supports both implicit and explicit data deletion and replacement. In the explicit scheme, the original source or destination deletes the item from the system when the data is successfully delivered or it is not useful anymore. In the implicit scheme, instead, the system can accommodate storage scarcity with prioritized storage management, such as the Least Recently Used (LRU) and First-in-First-out (FIFO) algorithms.

Although ASOS is a promising P2P overlay, the authors underline that it is necessary to further investigate the ASOS performances in a DTN environment. Moreover the authors assume the availability of pre-configured agents, which is clearly unrealistic in DTN scenarios.

6.4.2.4 Peer-to-Peer Swarm Intelligence

In [39], a Peer-to-Peer file sharing system based on Swarm Intelligence (P2PSI) has been presented. Swarm intelligence [43] is an artificial intelligence technique based on the study of biological swarms, such as ants or bees. P2PSI applies it to the problem of implementing P2P services in MANETs with a cross-layer approach.

P2PSI adopts as network layer the Ant-colony based Routing Algorithm (ARA) [44] as shown in Fig. 6.13, and it assumes that a large fraction of users are *free-riders*, i.e., they consume resources without providing them. In other words, there are only a small fraction of collaborative nodes, namely *HotSpots*, that store and share files.

P2PSI relies on two processes: advertisement (push) and discovery (pull). In the advertisement process each HotSpot periodically advertises to neighbor nodes the available data items within a limited area. The amount of information about the available items is limited by means of Bloom filtering techniques [45].

Query messages are forwarded at intermediate nodes based on the pheromone tables stored at these nodes. A pheromone table records the pheromone intensity for each neighbor. Intuitively, the pheromone intensity of a neighbor denotes the probability that a query message reached the destination via that neighbor.

When a node receives a resource query, it looks for the requested data item in the cached advertise messages and it eventually replies to the originating node with the identity of the node storing the item. The reply message reinforces the pheromone information along the way. In such a way, subsequent query messages that look for the same data items can follow previously laid pheromone information, without the need of a further route discovery process. Although the swarm intelligence is a very promising research area, more investigations are required to understand how such an approach performs over un-persistent routes. Moreover, the proposed solution requires a careful setting of several parameters, which is not as easy in dynamic environments as in DTNs ones.

6.4.2.5 Prophet-Based Information Retrieval

In a very recent work [40], the authors propose an unstructured content-based information retrieval system for DTNs, by focusing on the aspects related to data caching, query disseminations, and message routing.

For data caching, two schemes have been proposed, namely *random caching* and *intelligent caching*. In the first scheme each node storing a data item creates K tokens, where each token represents the right to make a copy of the data item. Then, the node spreads half of the owned tokens, along with a copy of the data item, to the nodes that it meets. Instead, the intelligent caching scheme requires that the K tokens are spread to nodes selected according to a friendliness metric, which represents a measure of the average number of nodes met during an observation period.

As regards to query dissemination, the authors propose two strategies: *W-copy selective query spraying* (WSS) and *L-hop neighborhood query spraying*

(LNS). The WSS strategy replicates a query to nodes selected according to the friendliness metric, while in the LNS one, each query is replicated to L-hop neighborhood, that is the nodes which are distant L hops from the originating node.

The authors adopt for message routing an enhanced version of the Prophet protocol [26], although different routing strategies can be used. According to the adopted strategy, messages are forwarded to neighbors based on the estimate of the delivery probability to the destination.

The numerical performance analysis conducted by the authors reveals that the best solution is the combined use of intelligent caching with the WSS scheme.

Although the proposal deals with DTNs and therefore it does not suffer from the issues underlined for the above discussed solutions, there are some open problems pointed out by the authors. One problem is related to the design of indices for the cached data items that allow nodes to determine if the newly encountered ones carry any data items that match the queries stored locally. Another problem is the assumption of fixed expiration times for the data items, since realistic applications require strategies to invalidate expired data.

6.4.3 Structured Mobile Peer-to-Peer Overlay Networks

Structured MP2P overlay networks generally adopt the Distributed Hash Table (DHT) paradigm as substrate to provide scalable content management. Although the use of DHTs simplifies the discovery process thanks to the a-priori knowledge of the data items, locations, the management of DHT tables is still an open problem in disconnected networks like DTNs ones.

In the following, we describe four proposals: the Mobile AD-hoc Pastry (MADPastry) [46], the Indirect Tree-based Routing (ITR) [6], the Virtual Ring Routing (VRR) [34], and the Opportunistic DHT-based Routing (ODR) [35]. For each system, we underline the main features along with the main limitations.

6.4.3.1 Mobile Ad Hoc Pastry

Mobile AD-hoc Pastry (MADPastry) [46] is a structured cross-layered P2P overlay network for MANETs in which a Pastry-like [12] application layer is combined with the Ad-hoc On demand Distance Vector (AODV) [41] routing protocol.

In standard DHTs, there is no relation between the overlay distance and the physical one, thus causing large overlay route stretch. To solve this issue, MADPastry utilizes the concept of Random Landmarking [47] to clustering nodes according to the overlay identifiers. Thus, two nodes that are physically close in the physical topology are also likely to be close in the overlay network.

To couple with node mobility, MADPastry does not rely on fixed landmark nodes. Instead, it uses a set of landmark keys chosen so that they divide the overlay id space into equal-sized segments. Nodes whose overlay identifiers are currently closest to one of the landmark keys become temporary landmark nodes. Clusters are formed by imposing that nodes have to associate themselves with the temporary landmark node that is currently closest to them. The association consists of adopting its overlay id as identifier prefix. Since broadcast messages introduce excessive overhead in resource-constrained networks, landmark beacons are only propagated within the landmark's own cluster.

MADPastry routes packets based on the overlay identifiers, i.e., by means of indirect routing. Each route is composed of several overlay hops, and each overlay hop corresponds to a physical path composed of several physical hops. At each overlay hop, the node will consult its Pastry routing table to find the node whose overlay identifier is numerically closest to the key. On the other hand, at each physical hop the node looks for the next physical hop in its AODV routing table.

To avoid unnecessary overhead in case of absence of valid information about the overlay next hop, a node belonging to the target's cluster broadcasts the data item within the confines of its cluster. Otherwise, if the node does not belong to the target cluster, it queues the data item and starts a regular AODV expanding ring broadcast to discover a route to the item's destination.

As pointed out by [48], the main issue with MADPastry is that queries are very sensitive to changes in AODV (physical) routes. In addition, the indirect routing based on overlay identifiers is a form of source routing, which is unsuitable in the absence of persistent paths.

6.4.3.2 Indirect Tree-Based Routing

The authors in [6] propose the Indirect Tree-based Routing (ITR), which extends the Multi-path Dynamic Address RouTing (M-DART)[49, 50], a DHT-based routing protocol for MANETs, by providing fully functional P2P services.

ITR assigns location-dependent identifiers, namely strings of l bits, to peers by means of a distribute procedure and locally broadcasted hello packets. The peer identifier space can be represented as a complete binary tree of $l+1$ levels as shown in Fig. 6.14-a, that is, a binary tree in which every vertex has zero or two children and all leaves are at the same level. In the tree structure, each leaf is associated with a peer identifier, and an inner vertex of level k, namely a level-k subtree, represents a set of leaves (that is a set of peer identifiers) sharing a prefix of $l-k$ bits. A level-k sibling of a leaf is the level-k subtree which shares the same parent with the level-k subtree the leaf belongs to.

Indirect Tree-based Routing performs the whole routing, resorting to an iterative procedure which explores the topological meaning of the node identi-

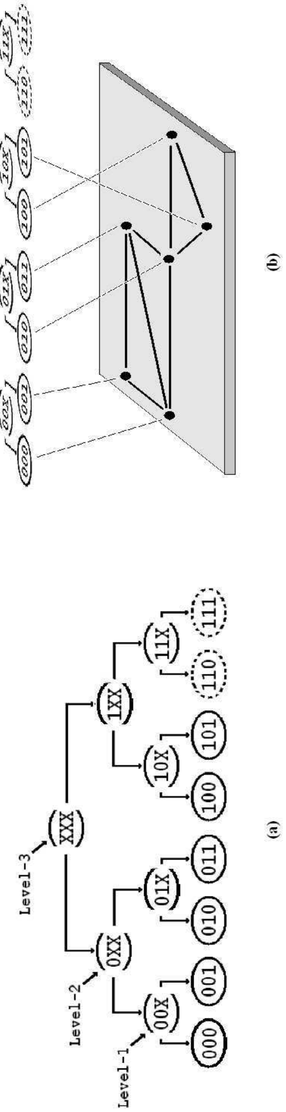

Figure 6.14: Address space and overlay network of ITR.

fiers with a hierarchical form of multi-path proactive distance-vector routing [51]. Each node stores a routing table with l sections, one for each sibling, and the k-th section stores the physical 1-hop neighbor peers which can forward a packet towards peers whose location-dependent identifiers belong to the level-k sibling.

From an operational point of view, ITR performs like traditional P2P systems: namely, when a node stores a resource, it periodically sends a pointer composed of a resource identifier and a storing peer identifier to the rendezvous-point, i.e., the node responsible (according to the hash function) for that resource. When a node has to retrieve a resource, it sends a resource query to such a rendezvous-point. Similarly, for MANET communications, each node periodically sends its current identifier to the rendezvous-point. When a node has to communicate with a node, it will send an identifier query to its rendezvous-point. After the reception of the query reply, the node can start a MANET communication.

The key-feature of ITR is the ability to forward both resource and identifier queries without introducing overlay path stretch, since the overlay distance is strictly related to the physical one, as shown by Fig. 6.14-b. Although ITR is one of the first proposals that tries to couple location-dependent identifiers with P2P overlay networks, the address space overlay management introduces a considerable overhead which could not be suitable in DTNs. Moreover, the ITR performances in DTNs have not been evaluated.

6.4.3.3 Virtual Ring Routing

In Virtual Ring Routing (VRR) [34], the authors adopt an integrated approach to provide connectivity in MANETs by exploiting the DHT paradigm. Like ITR, VRR integrates the DHT functionalities directly at the network layer, by providing both direct and indirect routing.

Nodes are identified by means of random location-independent unsigned integers, organized into a virtual ordered ring. Each node maintains a small number of routing paths, say r, to its logical neighbors, namely neighbors in the virtual ring. In more detail, it proactively stores the paths towards the $r/2$ closest neighbors clockwise in the virtual ring and the $r/2$ closest neighbors counter clockwise. Since node identifiers are random and location independent, the virtual neighbors of a node will be randomly distributed across the physical network and each virtual path is composed of several physical hops. Each node also stores a physical neighbor set by means of locally broadcasted hello packets.

The virtual neighborhood is used to route a packet in the network, by forwarding it to the node whose identifier is numerically closest to the destination in the overlay network. In addition, physical neighbors are exploited for packet forwarding to limit the overlay path stretch.

Since VRR routes messages sent to numerical keys to the node whose identifier is numerically closest to the key, it also supports DHT functionalities when the keys identify data items instead of VRR nodes.

Although VRR adopts an integrated approach, the management of the routing tables poses a strong issue about its application in DTNs since the node identifiers are randomly assigned to peers, i.e., there is no relation between logical and physical topology. In addition, like ITR, performances in DTNs have not been evaluated.

6.4.3.4 Opportunistic DHT-Based Routing

The Opportunistic DHT-based Routing (ODR) [35] protocol exploits the broadcast nature of the wireless propagation, by resorting to broadcast communications instead of traditional unicast ones, to provide connectivity in the presence of hostile conditions.

ODR exploits the same location-dependent address space of ITR, and it pushes down the stack the P2P functionalities at the network layer by resorting to indirect key-based routing. Differently from ITR, ODR is explicitly designed for disruption tolerant networks and its performances have been evaluated in such a scenario.

To accomplish the packet routing, each forwarder locally broadcasts the packet to all its neighbors, together with an estimate of its distance from the destination. By means of such a distance, the receiving nodes are able to understand if they are potential forwarders, that is if they belong to the candidate set, by comparing their distances with the one stored in the packet header as shown in Fig. 6.15. Clearly, the candidate set is composed by all the neighbors closer than the forwarder to the destination as well as the forwarder.

Each candidate node delays the packet forwarding by an amount of time which depends on its distance estimate from the destination: the more a node is close to the destination, the more the delay is short. A subsequent reception of the same packet from a neighbor closer to the destination allows the node to discard that packet, while a subsequent reception from a farther neighbor gives rise to an acknowledge transmission.

This proposal, like VRR, does not deal with P2P overlay networks; nevertheless both can be used as P2P overlay networks since they provide all the necessary P2P functionalities directly at the network layer. However, ODR, differently from VRR, has been designed to operate in DTNs. We underline that both ODR and VRR need further evaluations to assess their performances as P2P overlay networks.

6.4.4 Summary and Open Problems

Delay Tolerant Networks and Peer-To-Peer systems are emerging technologies sharing a common underlying decentralized networking paradigm. Nevertheless, the related research activities have been mainly developed by different research communities, nullifying therefore the idea of a unitary approach able to assure effective integrated solutions.

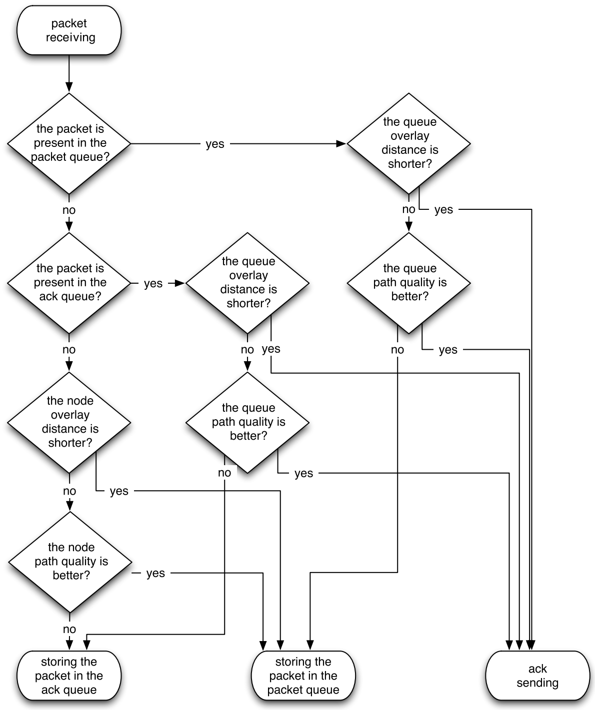

Figure 6.15: ODR packet forwarding process

In the last years, different proposals based on cross-layered or integrated approaches have been presented to overcome the poor performances due to the lack of cooperation and communication between the two layers, namely the application layer and the networking one. Despite these efforts, the design of Peer-to-Peer overlay networks on wireless multi-hop networks is still an open problem, and DTNs pose additional issues related to the lack of persistent connectivity.

In Table 6.2 we state a comparison among the systems previously described in terms of the following characteristics:

- adopted paradigm: resource-location aware (structured) or not (unstructured);
- adopted architecture: traditional layered, cross-layered, or integrated;
- P2P overlay network;
- routing protocol adopted for the network substrate;
- applicability: explicitly designed for DTNs or not;

Table 6.2: Comparison among the Considered Mobile Peer-to-Peer Overlay Networks

System	Paradigm	Architecture	Overlay network	Routing protocol	Applicability	Performance evaluation
ASOS	Unstructured	Cross-layered	Gnutella-like	AODV-like	Maybe DTN	Simulation
ITR	Structured	Integrated	Tree-based	M-DART	MANET	Simulation
MADPastry	Structured	Cross-layered	Pastry	AODV	MANET	Simulation
MPP	Unstructured	Cross-layered	Gnutella-like	EDSR	MANET	Simulation
ODR	Structured	Integrated	Tree-based	ODR	DTN	Incomplete
Orion	Unstructured	Layered	Gnutella-like	AODV	MANET	Simulation
P2PSI	Unstructured	Cross-layered	Gnutella-like	ARA	MANET	Simulation
VRR	Structured	Integrated	Pastry-like	VRR	MANET	Incomplete
Yang and Chuang	Unstructured	Cross-layered	Spraying-based	Prophet	DTN	Simulation

- performance evaluation: how the proposed P2P system has been evaluated.

We note that most of the unstructured P2P overlay networks adopt a Gnutella-like as a P2P application, modified to couple with node mobility. On the other hand, Pastry is a popular solution for structured P2P overlay networks. In both the classes, reactive routing is often used as a network substrate.

In theory, structured systems are able to efficiently retrieve data items thanks to their content-location awareness. However, they usually suffer from overlay route stretch since the overlay neighborhood concept is not related to the physical one. Such an effect is particularly significant in resource-constrained networks as in DTN ones. Moreover, the stretch effect implies additional latency for data items, disseminate/discovery. Finally, the maintenance of the structure in the overlay network can introduce a considerable overhead. On the other hand, unstructured systems are able to react quicker to changes in the network topology since they do not maintain topological information. However, their flooding-like strategies for resource discovery exhibit poor scalability with respect to the number of nodes in the network.

In the future, we expect that a new class of P2P overlay networks able to provide connectivity when both the assumptions of dense network topology and stationary wireless conditions are not verified will be developed. The design of these systems requires the exploration of the similarities of P2Ps and DTNs in terms of the commons store-carry-forward paradigm.

6.5 Conclusion

In this chapter we have focused on the issue of allowing the P2P functionalities to operate over a Delay Tolerant Network. More specifically, we have described the P2P system characteristics, capabilities, applications, and design constraints, thus providing an opportunity for beginner readers to acquire familiarity with such a very active research area.

As it has been shown in this chapter, there exists a variety of P2P systems designed specifically for mobile ad hoc networks, but few proposals deal with the problem of providing content information dissemination/discovery in delay tolerant networks.

It is likely that currently a single solution is not available that can satisfy the needs of every conceivable DTN scenario. However, the understanding gained from these first proposals can be used, in the coming years, to improve future designs of Mobile P2P systems, since there still remains much to do in terms of understanding, developing, and deploying a P2P overlay network for DTN scenarios.

Bibliography

[1] E. K. Lua, J. Crowcroft, M. Pias, R. Sharma, and S. Lim, "A survey and comparison of Peer-to-Peer overlay network schemes," *IEEE Communications Surveys and Tutorials*, Vol. 7, No. 2, pp. 72–93, Quarter 2005.

[2] "The napster website." [Online]. Available: http://free.napster.com

[3] "Gnutella protocol specification v0.6." [Online]. Available: http://rfc-gnutella.sourceforge.net/rfc-gnutella.zip

[4] "The emule website." [Online]. Available: http://www.emule-project.net

[5] V. Borrel, M. H. Ammar, and E. W. Zegura, "Understanding the wireless and mobile network space: a routing-centered classification," in *CHANTS '07: Proceedings of the second ACM workshop on Challenged networks*, 2007, pp. 11–18.

[6] M. Caleffi and L. Paura, "P2p over manet: Indirect tree-based routing," in *PerCom '09: The Seventh Annual IEEE International Conference on Pervasive Computing and Communications*, Mar. 2009, pp. 340–344.

[7] R. Schollmeier, I. Gruber, and M. Finkenzeller, "Routing in mobile adhoc and Peer-to-Peer networks: a comparison," in *Revised Papers from the NETWORKING 2002 Workshops on Web Engineering and Peer-to-Peer Computing*, 2002, pp. 172–186.

[8] A. C. Viana, M. D. de Amorim, S. Fdida, and J. F. de Rezende, "Self-organization in spontaneous networks: the approach of dht-based routing protocols," *Ad Hoc Networks*, vol. 3, no. 5, pp. 589–606, Sept. 2005.

[9] S. Ratnasamy, P. Francis, M. Handley, R. Karp, and S. Schenker, "A scalable content-addressable network," in *SIGCOMM '01: Proceedings of the 2001 conference on Applications, technologies, architectures, and protocols for computer communications.* ACM, 2001, pp. 161–172.

[10] B. Zhao, L. Huang, J. Stribling, S. Rhea, A. Joseph, and J. Kubiatowicz, "Tapestry: a resilient global-scale overlay for service deployment," *Selected Areas in Communications, IEEE Journal on Selected Areas in Communication*, vol. 22, no. 1, pp. 41–53, Jan. 2004.

[11] I. Stoica, R. Morris, D. Karger, M. F. Kaashoek, and H. Balakrishnan, "Chord: A scalable Peer-to-Peer lookup service for internet applications," in *SIGCOMM '01: Proceedings of the 2001 conference on Applications, technologies, architectures, and protocols for computer communications*, Aug. 2001, pp. 149–160.

[12] A. Rowstron and P. Druschel, "Pastry: Scalable, decentralized object location and routing for large-scale Peer-to-Peer systems," in *IFIP/ACM International Conference on Distributed Systems Platforms (Middleware)*, Nov. 2001, pp. 329–350.

[13] P. Maymounkov and D. Mazières, "Kademlia: A Peer-to-Peer information system based on the xor metric," in *IPTPS: 1st International workshop on Peer-to-Peer Systems*, 2002, pp. 53–65.

[14] D. Malkhi, M. Naor, and D. Ratajczak, "Viceroy: a scalable and dynamic emulation of the butterfly," in *PODC '02: Proceedings of the Twenty-First Symposium on Principles of Distributed Computing*, 2002, pp. 183–192.

[15] "The seti@home website." [Online]. Available: http://setiathome.berkeley.edu

[16] "The freenet website." [Online]. Available: http://freenetproject.org

[17] "The kazaa website." [Online]. Available: http://www.kazaa.com

[18] "The morpheus website." [Online]. Available: http://www.morpheus.com

[19] F. P. Franciscani, M. A. Vasconcelos, R. P. Couto, and A. A. Loureiro, "(re)configuration algorithms for Peer-to-Peer over ad hoc networks," *Journal of Parallel and Distributed Computing*, vol. 65, no. 2, pp. 234–245, 2005.

[20] F. Warthman, "Delay-tolerant networks (dtns): A tutorial v1.1." [Online]. Available: http://www.dtnrg.org

[21] "Nasa jpl's deep impact networking (dinet)." [Online]. Available: http://deepspace.jpl.nasa.gov/dsn

[22] L. Wood, W. Eddy, W. Ivancic, J. McKim, and C. Jackson, "Saratoga: a delay-tolerant networking convergence layer with efficient link utilization," in *International Workshop on Satellite and Space Communications, 2007. IWSSC '07*, Sept. 2007, pp. 168–172.

[23] S. Guo, M. H. Falaki, E. A. Oliver, S. Ur Rahman, A. Seth, M. A. Zaharia, and S. Keshav, "Very low-cost Internet access using kiosknet," *SIGCOMM Comput. Commun. Rev.*, vol. 37, no. 5, pp. 95–100, 2007.

[24] P. Mundur and M. Seligman, "Delay tolerant network routing: Beyond epidemic routing," in *3rd International Symposium on Wireless Pervasive Computing, 2008. ISWPC 2008*, May 2008, pp. 550–553.

[25] A. Vahdat and D. Becker, "Epidemic routing for partially-connected ad hoc networks," Duke University, Tech. Rep., 2000.

[26] A. Lindgren, A. Doria, and O. Schelén, "Probabilistic routing in intermittently connected networks," *SIGMOBILE Mob. Comput. Commun. Rev.*, vol. 7, no. 3, pp. 19–20, 2003.

[27] T. Spyropoulos, K. Psounis, and C. S. Raghavendra, "Spray and wait: an efficient routing scheme for intermittently connected mobile networks," in *WDTN '05: Proceedings of the 2005 ACM SIGCOMM workshop on Delay-tolerant networking*, 2005, pp. 252–259.

[28] T. Small and Z. J. Haas, "Resource and performance tradeoffs in delay-tolerant wireless networks," in *WDTN '05: Proceedings of the 2005 ACM SIGCOMM workshop on Delay-tolerant networking*, 2005, pp. 260–267.

[29] W. Zhao, M. Ammar, and E. Zegura, "A message ferrying approach for data delivery in sparse mobile ad hoc networks," in *MobiHoc '04: Proceedings of the 5th ACM international symposium on mobile ad hoc networking and computing*, 2004, pp. 187–198.

[30] M. C. Chuah and P. Yang, "A message ferrying scheme with differentiated services," in *Military Communications Conference, 2005. MILCOM 2005. IEEE*, Oct. 2005, pp. 1521–1527, Vol. 3.

[31] S. Jain, R. C. Shah, W. Brunette, G. Borriello, and S. Roy, "Exploiting mobility for energy efficient data collection in wireless sensor networks," *Mob. Netw. Appl.*, vol. 11, no. 3, pp. 327–339, 2006.

[32] K. A. Harras and K. C. Almeroth, "Inter-regional messenger scheduling in delay tolerant mobile networks," in *WOWMOM '06: Proceedings of the 2006 International Symposium on on World of Wireless, Mobile and Multimedia Networks*, 2006, pp. 93–102.

[33] M. Conti, G. Maselli, G. Turi, and S. Giordano, "Cross-layering in mobile ad hoc network design," *Computer*, vol. 37, no. 2, pp. 48–51, 2004.

[34] M. Caesar, M. Castro, E. Nightingale, G. O'Shea, and A. Rowstron, "Virtual ring routing: network routing inspired by dhts," in *SIGCOMM '06: Proceedings of the 2006 conference on Applications, technologies, architectures, and protocols for computer communications*, 2006, pp. 351–362.

[35] M. Caleffi and L. Paura, "Opportunistic routing for disruption tolerant networks," in *AINA '09: the IEEE 23rd International Conference on Advanced Information Networking and Applications*, May 2009, pp. 826–831.

[36] A. Klemm, C. Lindemann, and O. Waldhorst, "A special-purpose Peer-to-Peer file sharing system for mobile ad hoc networks," in *Vehicular Technology Conference, 2003. VTC 2003-Fall. 2003 IEEE 58th*, vol. 4, Oct. 2003, pp. 2758–2763.

[37] I. Gruber, R. Schollmeier, and W. Kellerer, "Performance evaluation of the mobile Peer-to-Peer service," in *IEEE International Symposium on Cluster Computing and the Grid, 2004. CCGrid 2004*, April 2004, pp. 363–371.

[38] G. Yang, L.-J. Chen, T. Sun, B. Zhou, and M. Gerla, "Ad-hoc storage overlay system (asos): A delay-tolerant approach in manets," in *IEEE International Conference on Mobile Adhoc and Sensor Systems (MASS), 2006*, October 2006, pp. 296–305.

[39] C.-C. Hoh and R.-H. Hwang, "P2p file sharing system over manet based on swarm intelligence: A cross-layer design," in *Wireless Communications and Networking Conference, 2007.WCNC 2007. IEEE*, Mar. 2007, pp. 2674–2679.

[40] P. Yang and M. Chuah, "Performance evaluations of data-centric information retrieval schemes for dtns," *Computer Networks*, vol. 53, no. 4, pp. 541–555, 2009.

[41] C. Perkins and E. Royer, "Ad hoc on-demand distance vector routing," in *2nd IEEE Workshop on Mobile Computing Systems and Applications*, 1999, pp. 90–100.

[42] D. Johnson and D. Maltz, "Dynamic source routing in ad hoc wireless networks," in *Mobile Computing*, Kluwer Academic Publishers, 1996, vol. 353, pp. 153–181.

[43] E. Bonabeau, M. Dorigo, and G. Theraulaz, *Swarm Intelligence: From Natural to Artificial Systems*, Oxford University Press, 1999.

[44] M. Gunes, U. Sorges, and I. Bouazizi, "Ara-the ant-colony based routing algorithm for manets," in *International Conference on Parallel Processing Workshops, 2002. Proceedings*, 2002, pp. 79–85.

[45] B. H. Bloom, "Space/time trade-offs in hash coding with allowable errors," *Commun. ACM*, vol. 13, no. 7, pp. 422–426, 1970.

[46] T. Zahn and J. Schiller, "Madpastry: A dht substrate for practicably sized manets," in *Proc. of 5th Workshop on Applications and Services in Wireless Networks (ASWN2005)*, June 2005.

[47] K. Xu, X. Hong, and M. Gerla, "Landmark routing in ad hoc networks with mobile backbones," *Journal of Parallel and Distributed Computing*, vol. 63, no. 2, pp. 110–122, Feb. 2003.

[48] K. Takeshita, M. Sasabe, and H. Nakano, "Mobile p2p networks for highly dynamic environments," in *PERCOM '08: Proceedings of the 2008 Sixth Annual IEEE International Conference on Pervasive Computing and Communications*, 2008, pp. 453–457.

[49] M. Caleffi, G. Ferraiuolo, and L. Paura, "Augmented tree-based routing protocol for scalable ad hoc networks," in *MASS '07: the IEEE Internatonal Conference on Mobile Adhoc and Sensor Systems*, Oct. 2007, pp. 1–6.

[50] M. Caleffi and L. Paura, "M-dart: Multi-path dynamic address routing," *Wireless Communications and Mobile Computing*, 2010.

[51] M. Caleffi, G. Ferraiuolo, and L. Paura, "A reliability-based framework for multi-path routing analysis in mobile ad-hoc networks," *International Journal of Communication Networks and Distributed Systems*, vol. 1, no. 4-5-6, pp. 507–523, 2008.