P2P Models and Complexity in MANETs

Boon-Chong Seet
Nanyang Technological University, Singapore

Chiew-Tong Lau
Nanyang Technological University, Singapore

Wen-Jing Hsu
Nanyang Technological University, Singapore

INTRODUCTION

Computing systems are playing an essential role as an indispensable nervous system of modern society. In recent years, peer-to-peer (P2P) computing has gained significant attention from both industry and research communities (Barkai, 2001). A key attraction of P2P systems is their ability to scale without requiring expensive and powerful servers, primarily because P2P systems work by distributing the functionality and harnessing the resources across a large number of independent peers. In addition to having high scalability, such systems are also inherently robust and fault tolerant since there is no centralized server, and the network is inherently self-organized. Today, P2P technology has been widely embraced by Internet users and has seen highly successful applications in such areas as digital data sharing, voice over peer-to-peer, distributed computing, and distributed storage (Miller, 2001). With the advent of wireless technology, the number of mobile users has increased tremendously over the years. It is envisioned that a significant portion of future users of P2P systems would be based on mobile ad-hoc networks (MANETs), which are dynamic networks formed by peers with no support of fixed infrastructure networks (Ramanathan & Redi, 2002; Perkins, 2001). As contemporary P2P models are designed mainly for the Internet, research is needed to examine their viability for P2P computing in a mobile environment.

This article first overviews some of the most common P2P architectures in use today by Internet users, and then proceeds to evaluate and discuss each of their architectural strengths and weakness for MANET through a qualitative complexity analysis.

INTERNET P2P MODELS

Internet P2P systems are a popular paradigm for data exchange in a decentralized manner between computer users across the Internet. A P2P network over the Internet is a highly dynamic overlay network where users join and leave the network frequently. Numerous P2P systems that have been proposed for the Internet in recent years may be broadly classified into two categories: *unstructured* and *structured* (Lua, Crowcroft, Pias, Sharma, & Lim, 2005).

Unstructured P2P systems do not have tight control of the overlay topology. The network is typically formed by nodes in a random manner, and thus the object or data that one wishes to locate could be anywhere in the system. A query has to be flooded through the network to search for peers that store the desired object. Besides a high amount of signaling traffic, the search may be prematurely terminated if the object could not be found within the query lifetime, often specified in terms of number of hops.

Structured P2P systems, on the other hand, impose a certain structure on the overlay topology and control of data placement to enable more efficient and reliable location of objects in a bounded number of hops through the use of distributed hash table or DHT (Stoica et al., 2003). In DHT, every object is mapped to some peer. A description (key) containing a link (value) to where the object could be found is then stored at the peer to which the object is mapped. This forms a deterministic relationship between objects and peers, and consequently, the query need not be flooded but routed directly to the peer responsible for storing the object location.

Until recently, Internet P2P systems (including both unstructured and structured systems) assumed all peers are equal and uniform in resource capacity. System functionality is thus distributed without considering real-world heterogeneity of peer capabilities. For example, some peers have less memory and slower processor than others, but they perform the same role and responsibility as other peers with higher capabilities. This results in instances of bottlenecks in performance due to very limited resources of these peers. To account for and even exploit the heterogeneity of peer capabilities, the notion of *super peers,* or nodes more well provisioned in terms of resources, have recently been introduced and advocated (Yang & Garcia-Molina, 2003; Singh, Ramabhadran, Baboescu, & Snoeren, 2003). Super peers take on a greater role and responsibility by serving as local search hubs that manage (receive and resolve) object queries of ordinary peers. Each super peer in turn communicates with other super peers as equals in a pure P2P way. As a result of clustering heterogeneous devices and elevating certain well-provisioned nodes to the role of super peers, the impact of performance bottlenecks presented by some less capable peers could be minimized.

P2P IN MANETs: A COMPLEXITY ANALYSIS

P2P systems for the Internet are designed with a decentralized architecture, which makes them potentially suited to the infrastructure-less MANET environment. However, as each overlay link could consist of multiple hops in the underlying physical network, a dynamic MANET topology due to peer mobility could pose significant problems for data management, particularly in structured P2P systems (Hsiao & King, 2005). The lower capacity of MANET compared to the Internet and the resource constraints of mobile devices further limit the usability of P2P architectures that have high traffic overhead.

This section provides a complexity analysis of the asymptotic cost and performance of three popular Internet P2P architectures—unstructured super peer, structured non-super peer, and structured super peer—in a mobile environment. The cost measure is the message complexity, or number of hop-wise message transmissions required to perform a specific operation such as node join, node leave, and object query. The performance measure is the time complexity or number of time steps needed to perform an operation.

Unstructured Super Peer

The popular file-sharing application Kazaa (Liang, Kumar, & Ross, 2005) and voice-over-IP (VoIP) client Skype (Baset, & Schulzrinne, 2006) are prominent examples of commercial-grade P2P systems based on unstructured super peer architecture. As these systems are originally designed for wired networks, as a first step, we need to imagine how such a system may be used in a mobile environment. For super peer-based architectures, we reason that the most natural choice of an underlying physical network would be a cluster-based MANET (Yu, & Chong, 2005), since the cluster-heads are ready candidates for super peers in the P2P system.

We assume each ordinary peer (mobile user) would know a super peer (cluster-head) through the underlying clustering mechanism. When a mobile user u wishes to join, it sends a *join* message to its super peer P with a list of objects it wishes to share, and records P as the super peer through which it joins the system. Upon receiving, P adds the object list of u into its index. When u wishes to leave, it sends a *leave* message to its super peer. If the current super peer is not P due to node mobility—that is, u is now associated with a different super peer in another cluster—the super peer forwards the message to P. Upon receiving, P deletes the object list of u from its index. When u wishes to query for an object, it sends a *query* message to its super peer. If the super peer knows the object, it replies to u immediately with the object location. Otherwise, it broadcasts the query to other super peers (via cluster gateways). Assume from the reply, u knows that a mobile user v is holding the object. u then proceeds to contact and retrieve the object from v.

Having outlined the join, leave, and query operation of the unstructured super peer system in a MANET environment, we may now begin our analysis. Consider a network of n nodes, of which a fraction m (where $0 < m < 1$) are super peers. Thus, we have mn super peers and $(1-m)n$ ordinary peers. Let us denote n_s and n_o as the number of super peers and ordinary peers, respectively. If m is a constant, then n_s and n_o would increase in proportion to n. Under constant node density, n itself would increase in proportion to A, the network area—that is, $n \propto A$.

If the communication pattern is generally uniform, the average path length L or number of hops between any pair of nodes is expected to grow with the spatial diameter of the network, which in turn grows with the square root of the area, or equivalently $\sqrt{n}$ (Li, Blake, Couto, Lee, & Morris, 2001). Similarly, we expect path length L_s between any pair of super peers, and path length L_o between any pair of ordinary peers, to grow with $\sqrt{n}$.

During a join, the *join* message from a mobile user to its super peer travels a path length of c, where c is a positive constant denoting the number of hops to a cluster head, which depends only on the clustering scheme used—that is, 1- or multi-hop clustering (Yu & Chong, 2005). The join process thus has a constant message and time complexity of $O(c)$. During a leave, the *leave* message travels under worst case a path length of $c+L_s$: mobile user → its current super peer → the super peer that originally hosts its object list, leading to a message and time complexity of $o(\sqrt{n})$ for the leave process.

The analysis for the query process is slightly more involved, thus we present for only the most common case of single-hop clustering: $c = 1$. First, consider a mobile user sending a *query* message to its super peer. If the super peer does not know the object, it broadcasts the query to other super peers. We denote a fraction q of the ordinary peers as gateways that rebroadcast the query to other super peers. If k is the fraction of super peers that reply with the object location, there would be kn_s replies and $(1-k)n_s$ rebroadcast queries by the super peers. Each reply would travel a path length of L_s+1: replying super peer → super peer of mobile user → mobile user. Summing up, this gives a total message cost of: