Computing and Informatics, Vol. 30, 2011, 559–594

PEER-TO-PEER NETWORKS AND COMPUTATION: CURRENT TRENDS AND FUTURE PERSPECTIVES

Ankur Gupta

Model Institute of Engineering and Technology
Kot Bhalwal
Jammu, Jammu and Kashmir
India
e-mail: ankurgupta@mietjammu.in

Lalit K. Awasthi

National Institute of Technology, Hamirpur
Himachal Pradesh
India
e-mail: lalitdec@yahoo.com

Communicated by Bogdan Wiszniewski

Abstract. This research papers examines the state-of-the-art in the area of P2P networks/computation. It attempts to identify the challenges that confront the community of P2P researchers and developers, which need to be addressed before the potential of P2P-based systems, can be effectively realized beyond content distribution and file-sharing applications to build real-world, intelligent and commercial software systems. Future perspectives and some thoughts on the evolution of P2P-based systems are also provided.

Keywords: P2P networks/computing, state-of-the-art in P2P networks, future trends in P2P domain

1 INTRODUCTION

Peer-to-Peer (P2P) networks and the computations that they facilitate have received tremendous attention from the research community, simply because of the huge untapped potential of the P2P concept – extending the boundaries of scale and decentralization beyond the limits imposed by traditional distributed systems, besides enabling end users to interact, collaborate, share and utilize resources offered by one another in an autonomous manner. Moreover, P2P architectures are characterized by their ability to adapt to failures and dynamically changing network topology with a transient population of nodes/devices, while ensuring acceptable connectivity and performance. Thus, P2P systems exhibit a high degree of self-organization and fault tolerance.

The P2P concept represents a paradigm shift from the client-server or hub-spoke model to a more decentralized device to device model. The devices perform the role of either client or server depending on the application and the nature of interaction. Since the interaction among peer devices is direct in nature it frees up the most basic of resources – network bandwidth, which was placed under tremendous strain due to millions of users accessing information over the internet, using the traditional client-server paradigm, where a few servers cater to the ever increasing demand for information from the end users. The peer model allows end users to directly connect to other peers on the internet, forming groups and collaborating, leading to the creation of virtual supercomputers, immense file systems offering potentially limitless storage, user created search engines and other novel applications.

A formal definition of P2P systems was provided by Theotokis [1] et al. as follows: "*Peer-to-peer systems are distributed systems consisting of interconnected nodes able to self-organize into network topologies with the purpose of sharing resources such as content, CPU cycles, storage and bandwidth, capable of adapting to failures and accommodating transient populations of nodes while maintaining acceptable connectivity and performance without requiring the intermediation or support of a global centralized server or authority.*"

In the last ten years P2P based systems have caught the fancy of millions of internet users' worldwide beginning with Napster [2], a hybrid P2P application which revolutionized the way digital music files were shared between users of the internet. Another flooding based P2P system, Gnutella [3] has been deployed and used extensively for sharing and exchanging files with over one million users and more than 10 terabytes of shared data. Other examples of popular P2P applications/protocols include Seti@Home [4] using the idle CPU cycles of millions of computers connected to the internet to download and analyze data from radio telescopes in the search for extra-terrestrial intelligence, OceanStore [5] (providing a persistent and secure data store scaling to billions of users built on top of untrusted servers), KaZaA [6] (for distributed file sharing and concurrent searching using search agents), BitTorrent [7] (a P2P communications protocol for content sharing and distribution) and CoolStreaming [8], based on P2PTV (designed to redistribute video streams in real-time to P2P networks). Different studies peg P2P traffic at 20 to 80 % of all internet traffic. The steep variation arises out of the thin line of classification employed between distributed, client/server and P2P systems. Some consider social networking sites like Facebook [9], YouTube [10] and other similar applications to be hybrid P2P since peers are responsible for generating and sharing content with each other. Cisco [11] projects an annual growth rate of over 100 % for P2P traffic for the next five years with new applications and software systems expected to be built on the P2P paradigm. In any case, the emergence of P2P as a widely adopted concept for current and future internet applications cannot be denied.

P2P researchers and developers have focused mostly on content distribution and file sharing systems, with a special emphasis on algorithms to improve the efficiency of query routing and correctness of data search in P2P networks, with the result that the issues, techniques and solutions involved are well understood within the community. The next generation of P2P systems shall focus increasingly on allowing peers to not only share information in a more secure and trust-worthy manner, but also allow peers to offload execution of tasks to other peers and allow commercial software systems to be built and deployed enabling advanced P2P interactions/collaborations. Some examples of early work in this direction include Groove [12] and JXTA [13].

The aim of this research paper is to identify the challenges that still exist in the field of P2P networks/computing, based on a comprehensive review of existing research literature, besides attempting to provide some insights into the future trends that shall characterize the evolution of P2P based systems. The rest of the paper is organized as follows: Section 2 provides an overview of the P2P system model, while Section 3 highlights the areas which current research in the field is focusing on. Section 4 lists the open issues which still require to be addressed and Section 5 provides some future trends that could be significant in the evolution of P2P systems. Finally, Section 6 concludes the paper.

2 SYSTEM MODEL

P2P networks are organized as overlay topologies on top of the underlying physical network topologies and are formed by peers connecting to each other in either a structured or unstructured manner. Figure 1 provides a conceptual representation of the P2P overlay topology. Since P2P networks are fault-tolerant, not susceptible to single-point-of-failure and required to cater to a transient population of nodes, P2P overlay topologies are multiply-connected Broadly, there are 3 classes of P2P systems:

Pure P2P Systems – in which 2 nodes/devices interact with each other without requiring intervention of any central server or service.

Hybrid P2P Systems – in which peers rely partially on a central server to provide certain services, although the interaction between peers still takes place independently.

Federated P2P Systems – in which peer interactions take place inside pre-defined domains, such as within an organization.

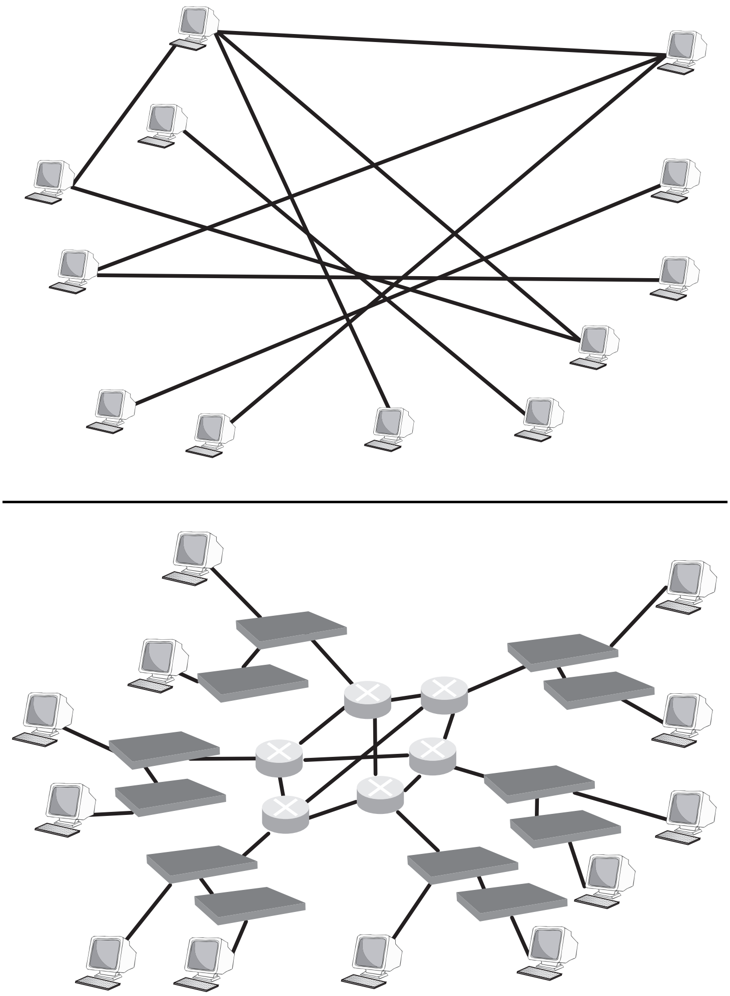


Fig. 1. An example of P2P overlay topology, over a physical network

Moreover, P2P systems can either be structured, where the overlay graph is well-structured and a mathematical scheme (e.g. Distributed Hash Tables) is applied to make sure that new nodes are added in a manner which maintains the structure or unstructured, where the overlay is random graph type and new nodes are added to the network in an unpredictable manner. Structured P2P systems are formed on the basis of node-identifiers, guaranteeing information retrieval in bounded time for simple queries and are self-organizing in the face of failures whereas unstructured

P2P systems can support large complex queries, but do not guarantee information retrieval in bounded time with not so efficient self-organization capabilities.

Some of the hybrid approaches to P2P topology organization rely on the fact that not all peers are equal in terms of the computing resources that they offer and hence treating them as equal peers would not be fair. Peers which offer more computing power to enable them to perform a more central role in a P2P environment are referred to as super-peers and P2P topologies are organized around these super-peers. Super peers are expected to maintain distributed indices for the entire network, forward search queries and aggregate resources of the peers directly connected to them etc. Figure 2 provides a high-level view of a super-peer based P2P topology.

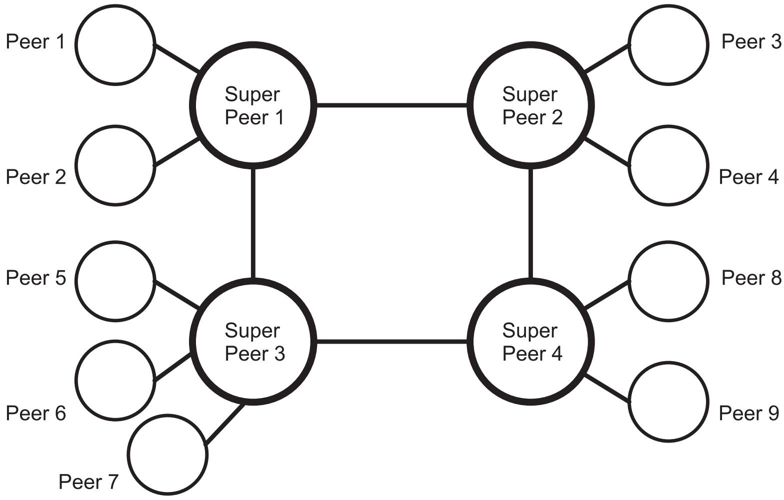


Fig. 2. An example super-peer based P2P topology

3 CURRENT RESEARCH

Recent research in P2P systems has focused on three categories of P2P systems, offering

- communication and collaboration
- distributed computation
- content distribution.

Out of these most of the research is focused on techniques and strategies of content distribution which focus on authoring and publishing, indexing and organizing and searching and retrieval of information. Communication and collaboration systems have been around for some time and represent mostly hybrid P2P systems, like chat and instant messaging systems, with some element of centralized control like identity management etc. A truly collaborative environment integrating communication, content distribution and distributed computation is still not available. Figure 3 provides a timeline depicting the evolution of P2P systems, although many of the developments took place in parallel and cannot be represented as occurring in a strictly sequential manner. The following sections discuss the state-of-the-art in P2P networks.

3.1 Construction/Organization of P2P Networks

One of the basic areas of research has been organization of P2P networks (both structured and unstructured). Essentially, P2P topologies can either be connected in a structured (a well-defined mathematical scheme is applied to maintain the overlay topology on addition or deletion of nodes) or unstructured manner (overlay is random graph type and new nodes are added in an unpredictable manner). The focus of all topology construction schemes is to reduce overlay routing hops and consequently improve the search performance for P2P applications deployed on the overlay network.

Solutions for organization of structured P2P networks have ranged from Content Addressable Networks (CANs) based on Distributed Hash Tables [14, 15]. Many of the DHTs evolved from the early work of Plaxton et al. [16], which describes a locality-aware tree-based scheme for efficient storage and retrieval of distributed objects. Another variation on CAN is the Multi-Hypercube organization [17, 18] of P2P networks which are extremely fault tolerant in the face of heavy-node transience and network churn. Other researchers have proposed tree-based structures [19, 20] enabling efficient organization of P2P overlay topology. These schemes map content (essentially files) to a unique key. Each peer in the overlay topology maintains a subset of indices containing a range of keys, essentially mapping content to the nodes which store the content. Peers issue key-based queries, which are then routed through the overlay topology based on the index information maintained at each peer.

Other examples of structured overlay network construction include Chord [21], Pastry [22] and Tapestry [23]. Chord and Pastry tend to construct ring overlay topologies by ordering all peers into a circular identifier space to reduce the routing hops and to improve search performance. Tapestry goes a step further by placing object replicas and location pointers throughout the overlay network and handling search queries closer to the requesting peer, improving search performance. Viceroy [24] is a butterfly network organizing the peers in a crossbar-switch configuration allowing only unidirectional communication, with a ring of successor and predecessor links which help in performing vicinity searches. Kademlia [25] utilizes the XOR metric to measure distance between two peers, to be used for routing between peers in the overlay space. Two node ID's or a node ID and a key are XORed and the result is the distance between them. For each bit, the XOR function returns zero if the two bits are equal and one if the two bits are different. Koorde [26] is another DHT which is based on Chord [21] and De Bruijn graphs [27] to achieve the $O(\log N)$ lookup upper bound with $O(\log N)$ neighboring peers.

Hierarchical P2P networks [28, 29] rely on clustering the overlay topology around super-peers (peers which have greater compute power and are resource-wise better endowed than other peers in the overlay topology) or combining both structured and unstructured peer topologies. This helps in improving the scalability and performance of P2P networks while retaining the resilience offered by traditional P2P networks.

Recently the concept of one-hop DHTs has been proposed [30]. If the network churn is not excessive, then one-hop DHTs can provide one-hop lookup by storing the global lookup table at each peer. This scheme is suitable only for very stable environments and conserves bandwidth.

At the other end of the spectrum are random unstructured P2P networks, which have a desirable property that the overheads of topology creation and maintenance are avoided. All structured P2P networks need to implement a maintenance function, which may need to be invoked frequently depending on the churn/transience in the network. However, unstructured P2P networks suffer from the drawback that they are unable to provide search success guarantees and results in finite time. Detailed comparison of various network topologies is provided in [31]

3.2 Overlay Topology Optimization

Once the basic topologies were established, the research community began to focus on optimizing the overlay topologies primarily to reduce the search overheads associated with large P2P networks and handling frequent churn in the overlay topology. Several schemes have been proposed for optimization of P2P overlay topologies, both structured and unstructured, with a view to achieving performance improvements over the flooding based search mechanism prevalent in earlier P2P systems. Most of these schemes rely on an awareness of the underlying network topology in the construction or adaptation of the overlay topology to reduce the ratio of overlay hops to physical hops and consequently the latency for content lookup. Other schemes rely on creating clusters of high-bandwidth connection nodes to improve the overall network performance.

An efficient overlay construction scheme for random unstructured P2P networks is provided in [32] by Vishnumurthy et al. They provide a scheme called SwapLinks which maintains the indegree and outdegree of each peer in the network ensuring that the load is equitably distributed amongst peers, thereby improving overall performance. Wan et al. [33], propose an autonomous topology optimization scheme for unstructured P2P networks, wherein each peer optimizes its connections based on the underlying physical topology. This significantly reduces the average hop count between any pair of nodes. Another scheme proposing a self-organizing, adaptive overlay topology can be found in [34]. This scheme attempts to build resilience and adaptability into the overlay topology by ensuring that each peer proactively

Fig. 3. Timeline depicting evolution of P2P systems

maintains a minimum number of links at all times, thus improving connectivity, service availability and resilience to attacks.

Topology optimization schemes for structured P2P overlay topologies include Virtual Binary Index (VBI) Trees [35] which are a balanced tree structure guaranteeing content retrieval in O(log N) hops, construction of small-world overlay networks [36] to handle flash-crowd scenarios (an overload condition when many peers try and access the same content), and random landmarking [37] (in which peers use well-known landmark peers to organize themselves into groups of nodes which are in physical proximity to each other).

For applications focusing on multicasting or sharing streaming media and video broadcasting availability of a high-bandwidth route from the requester to the responding peer is a high priority, even in the face of node transience. For such applications topology organization schemes based on connection management [38, 39] to improve overall available bandwidth have been proposed. These schemes rely on clustering high-bandwidth peers into clusters or multi-cast groups so that traffic can be efficiently routed through them.

3.3 Overlay Routing and Content Search

Overlay routing and content search has been the single most important focus area for P2P researchers. In fact, the plethora of overlay topologies and the optimizations available for the topologies serve only to reduce the routing overheads, making content search more efficient. The amount of work done in this specific area can be judged by the fact that though P2P networks are perceived to provide only best-effort services, for content search and retrieval Quality-of-Service (QoS) parameters have been well-defined for quite some time now and include guaranteed results, efficient utilization of resources, fast response time, correctness of results and resilience to changes in network topology. Different search mechanisms have been employed for structured and unstructured P2P networks. Table 1 summarizes the search-related issues for various overlay topologies and contrasts some well-known search schemes.

Early search mechanisms in unstructured P2P file sharing networks were based on flooding, with queries being propagated by a peer to all connected peers, placing a lot of overheads on network bandwidth. Simple optimizations such as introduction of time-to-live (TTL) parameter for queries and discarding duplicate messages, helped improve the performance of flooding-based schemes. Gnutella [4] is a well-known flooding-based content search application.

Another important development for improving search performance in P2P networks was to introduce content replication, ensuring that the probability of content location was drastically improved. Theotokis et al. [2] classify replication techniques into 5 categories: passive replication (occurs naturally as nodes copy content from each other), cache-based replication (caching copies of content as it passes through intermediate peers during the download process, thereby improving its availability; prevalent in OceanStore [6], MojoNation [40] and FreeNet [41]), active replication (content is actively replicated at various locations in the network), introspective replication (traffic and content requests are observed and content is replicated to better satisfy future requests for the same content) and dynamic replication (placing a minimum number of replicas throughout the network while taking care of server capacity and ensuring QoS compliance).

An improvement over flooding was parallel random walks [42] in which peers randomly select a neighbor to forward the query to. The query is executed in parallel at multiple locations and combined with content replication, improved search performance significantly. However, a need was felt to have more resource-preserving search algorithms. To make search performance more deterministic, peers started maintaining routing indices for known content to shape query propagation and ensuring that query responses can be received in a finite time. The use of more sophisticated broadcast policies, selecting which neighbors to forward search queries to, based on their past history, as well as the use of local indices was first presented in [43]. An Intelligent Search Mechanism built on top of a Modified Random Breadth-First Search (BFS) Mechanism is proposed in [44]. Each peer forwards queries to a subset of its neighbors, selecting them based on a profile mechanism that maintains information about their performance in recent queries.

Proximity-based routing for structured P2P networks exploits the knowledge of the underlying physical topology in making optimal routing decisions to peers which are physically closer. P3ON (Proximity Based Peer-to-Peer Overlay Network) [45] describes a two-tier ringed topology with the first tier comprising all peers in the topology and second tier comprising peers within an Autonomous System (AS), which are naturally clustered together. The lookup latency is significantly reduced if the lookup is successful in the second tier. Zeng-de et al. [46] describe a proximity-based routing scheme which combines the knowledge of internet topology with the routing table information for the overlay network to select peers which are physically closer. These schemes are resource efficient since they do not construct a topology-aware overlay. However, they tend to work well with smaller overlays. Topology-based node ID assignment, as employed in [47], is another scheme which maps the overlay ID space onto the physical network such that peers which are closer in the physical space are also closer in the virtual space. However, it suffers from load-balancing problems since the uniformity of the ID space is destroyed. A better approach is proximity neighbor selection as used in Chord [21], which constructs an efficient topology-aware overlay while achieving comparable performance to topology based node ID assignment, besides being more robust.

Content-Based Routing (CBR) is also another approach which is an extension of the CAN and publish-subscribe concepts. Here messages are routed based on their content rather than the destination addresses (overlay IDs). The typical strategy in this approach is for the receiver to broadcast subscription requests specifying the areas of interest. Whenever messages are sent they are delivered to all recipients who had registered their interest in the contents of that particular message. Thus, CBR is a recipient driven routing technique rather than the sender specifying the recipient address as is the case with other routing techniques for P2P networks. Due to the sheer scale of P2P networks, the scalability and efficiency of CBR is limited since the number of subscription requests to be catered to can impact the performance significantly. Recently some techniques have been proposed to overcome the scalability challenge for CBRs. These include HyperCBR [48] and ROSE [49]. Both schemes rely on dividing the overlay space into multi-dimensional partitions for the actual messages and subscription requests to travel independently. These partitions do intersect to allow messages and subscriptions to be matched and routed. This improves the load-balancing and increases efficiency allowing the scheme to scale to large topologies.

Not all search based schemes rely on forwarding of query messages to other peers. iSearch [50] incorporates an intelligent technique based on the Fuzzy Adaptive Resonance Theory (Fuzzy ART) neural network to perform document clustering in order to support content-based publishing and retrieval over P2P networks. This helps avoid indexing and improves performance over flooding-based systems. Yang et al. provide an evaluation of the GUESS [51] non-forwarding search technique and demonstrate that non-forwarding search provides performance an order of magnitude better than forwarding-based techniques.

Recent work in routing and content focuses on improving the search efficiency for a wide-variety of application areas. Chen et al. [52] have proposed an efficient search mechanism based on a two-phase Ticket-Based Search algorithm (TBS), based on ticket broadcasts. Each ticket represents permission to search a node. This scheme optimizes the search for both popular (by reducing lookup delay) and rare files (by minimizing duplicate messages). Mashayekhi et al. [53] propose a semantic based search for unstructured P2P networks, in which they maintain ontology based indices for each outgoing link at a peer. These indices also reflect the number of files accessible by following a particular outgoing link and average hop distances. This helps in improving the search performance. Vishnevsky et al. [54] propose a Recursive Partitioning Search (RPS) mechanism which prevents duplicate queries and significantly reduces search overheads by ensuring that each peer is visited only once during the search.

Range queries, which seek multiple objects within a particular range of overlay space IDs, have also been addressed by several schemes. It is based on the premise that looking for multiple possible results is more efficient than seeking single objects. Since range queries look for a range within a dataset it helps if semantically similar content is stored in a cluster of peers. Thus, this scheme is suitable for structured P2P networks, built over non-DHT based topologies, since hashing destroys the data locality of the hashed content. Schütt et al. [55] have proposed an efficient scheme to handle range queries over unstructured P2P networks, while González-Beltrán [56] propose handling range queries over skip-tree graphs, a specialized data structure. For hierarchical networks Tran et al. [57] propose a multi-dimensional search scheme, EZSearch, which is based on the Zigzag hierarchical overlay topology, enabling k-nearest neighbor and range queries to be handled with low overheads.

Along with maintaining routing indices for locating content faster, another class of systems focuses on intelligent overlay routing of P2P queries to optimize the search performance. These schemes are generally for unstructured networks, since in structured overlay topologies, the routing is based on efficient organization of content based on Distributed Hash Tables (DHTs). Brocade [58] focuses on landmark based routing, differentiating between local and inter-domain routing to reduce wide-area traffic and routing latency. Landmark peers are used to define the boundaries for local and remote areas. SkipNet [59] is an overlay network with locality properties which helps in making dynamic routing decisions to locate content faster with reduced hops for query propagation. It uses a multidimensional skip-list data structure to support overlay routing, maintaining both a DNS-based namespace for operational locality and a randomized namespace for network locality. In recent work Kumar et al. [60] proposed an efficient and scalable routing scheme for unstructured networks, while Gatani et al. [61] propose an adaptive routing scheme for locating specific compute resources in P2P computational grids. Shi et al. [62], propose a dynamic routing protocol for keyword-based search in unstructured P2P networks.

3.4 Security

As always, security is a major research area with the anonymous nature of P2P systems offering malicious users the cover to indulge in disruptive acts. In fact, P2P based systems have become notorious for blatant copyright violations and propagation of security threats and distributed attacks. Table 2 summarizes the security threats prevalent in P2P networks/computation and proposed solutions in literature to counter these threats. It is evident that due to the autonomy, anonymity, censorship-resistance, overlay routing, large scale churn afforded by P2P networks no comprehensive security scheme exists which alleviates all the security vulnerabilities. In contrast, other distributed computing architectures like grids are much better at providing security with centralized elements ensuring authentication and authorization for registered users.

Anonymity remains the core issue affecting security for P2P applications. How can a peer know whether its counterpart peer is genuine or malicious? Can the content from a peer be trusted? Can the other peer be allowed to access your computer resources, content without harming your computer? The basis for such interactions is the "Trustworthiness" of a peer, which needs to be established. Several solutions based on trust and reputations of peers participating in the P2P network have been proposed. EigenTrust [63] and TrustMe [64] are two well-known schemes for establishing the trust ratings of a peer through its interactions with other peers. The sheer scale of P2P systems makes trust computation and ensuring the security of every peer a Herculean task. As such, the solutions proposed by researchers have focused only on certain class of P2P applications like file sharing and storage management. These distributed solutions have focused on identifying trustworthy peers and attempting to isolate malicious peers within the realm of the application under consideration. Trust and reputation values are computed at a per peer basis and then aggregated and communicated to the entire network of peers, which can be time consuming and less than efficient. Over the years several other researchers have pro-

Unstructured P2P Networks		**Structured P2P Networks (DHT-Based)**		**Structured P2P Networks (non DHT-Based)**	
Not required to maintain structured topology and distributed index for locating content		Requires to maintain strictly defined overlay topology and distributed index to locate content		Avoids hashing to prevent overheads associated with DHT-based search techniques.	
Low overheads since index need not be updated in face of frequent node-transience		High-overheads since there is frequent node transience/churn in P2P networks		Overheads involved in maintaining *Content Locality* and *Path Locality* to store data close to users. Lower overheads compared to DHT-based techniques	
Widely-used in practice		All early schemes for searching content in P2P networks were built on top of DHTs.		Recently real-world applications have begun to incorporate search schemes for non-DHT-based networks	
Offers no guarantees on locating requested content even if it exists in the network		Typically provide guarantees of returning content in bounded-time		Typically provide guarantees of returning content in bounded-time	
Gnutella		Chord, CAN, Pastry, Tapestry, Kelips		SkipNet, SkipGraph and TeraDir	
Conventional Flooding	**Random Walks**	**Flat DHT**	**Hierarchical DHT**	**SkipNet**	**Range Query**
Every peer forwards query	Multiple queries with peer clustering	All peers perform routing.	Hierarchical routing performed. Super Peers maintain routing indices for lesser peers.	Built over Skip Lists. Each peer maintains pointers to other peers in the network. These pointers are used for routing.	Search for items within a specified range of dataset. DHTs do not support range queries well since hashing destroys data locality.
Search is done in k * N time (k = avg degree of nodes, N = total number of nodes)	On average search is done in log time	On average search is done in log time	On average search is done in log time	On average search is done in log time	On average search is done in log time
Resource-intensive and inefficient. Normally TTL and hop-constrained flooding used to improve efficiency	Efficiency depends on effective clustering of peers. Overhead of clustering peers.	Efficient routing due to strict ordering of peers and their content. High overhead in maintaining ordering	Efficient routing, routing indices maintained at each node are reduced. Overhead in maintaining indices.	Fairly efficient. Overhead of maintaining link pointers for peers in face of node transience and churn.	Less efficient than SkipNets. Overhead of maintaining doubly-sorted linked lists to support range queries effectively.

Table 1. Summary of search schemes for various categories of P2P networks

posed schemes for computing and disseminating trust values for peers. A detailed discussion on various trust and reputation-based systems is available in [65].

Security solutions for P2P applications operating within federated domains, such as organizations or those relying on setting up specialized communities/groups range from identity management [66], authentication and authorization [67], admission control in peer groups [68], access control [69] and role-based access control (RBAC) [70]. However, these solutions tend to introduce centralized elements for issuing certificates, public/private keys etc, which does not fit well in the decentralized model for P2P networks. Moreover, these represent a single-point-of-failure in the scheme. Some attempts are being made by researchers to propose decentralized security models. Some attempts have focused on letting peers define security policies suited to their interactions with other peers [71].

3.5 Distributed Computation

P2P networks enable computation-in-the-large with potentially millions of computers connected to the internet offering billions of MHz of computing power and storage, helping create virtual supercomputers. The research community has focused on several issues involved in distributed computation such as resource location and management, utilizing idle compute cycles, distributed scheduling, remote work processing, security issues, fault-tolerance etc. However, the thin line between issues related to computational grids and P2P networks used for distributed computation seems blurred. While grids involve the use of dedicated resources and fall into the realm of high-performance computing, P2P networks can be viewed as providing a best-effort service with heavy node-transience to contend with. Not many researchers have focused on distributed computation in pure P2P networks, making many simplifying assumptions about the nature of P2P networks. Moreover, existing schemes tend to focus on creating P2P computational grids within federated domains, such as organizations, rather than on the internet. Also, due to the serious security issues involved in allowing remote code to execute on their systems, organizations have stayed away from the potentially benefiting interactions with other organizations. Hardly any cross-organizational distributed computational scheme based on the P2P concept exists. Table 3 provides an overview of the various issues involved in distributed computation and the solutions proposed by the research community.

There have been instances of utilizing idle CPU cycles commercially and in other real-world applications. Some of these applications are listed below:

- DataSynapse [72]: This company maintains a network of broadband-only members. It resells its members' idle PC capacity, but it also has a platform for corporate infrastructures. The service targets energy and financial industries. It does not support cross-organizational interactions.
- Entropia [73]: Entropia resembles DataSynapse, but most PC resources go to research projects. Members choose the projects to support, such as AIDS research.

- Boinc [74]: This site resells members' processing power and offers a corporate platform. It can harness power from PCs, from Macs, and from Linux-based systems.

P2P Feature	Resultant Security Threat	Proposed Solutions
Anonymity	Untrusted Content, Cannot Share Resources, Cover for disruptive activities, No traceability	Centralized Identity Management, Trust and Reputation
Overlay Application Layer Routing	Man-in-the-Middle (MitM) and Distributed-Denial-of-Service (DDos) attacks, Sybil attacks, Network/Content Poisoning, Eclipse Attack	Strong Encryption, Trust and Reputation, Secure Routing
Sharing of Compute Resources/Remote Work Processing	Viruses/malware/spyware, potential damage to peer hosting remote work, cannot trust result of remote computation.	Role-Based-Access-Control (RBAC), Admission Control in Peer Groups, Trust and Reputation, Containment-Based Security
Transience/Churn	Leeching, no guarantee of retrieval of remote stored content, malicious peers can vanish with critical data.	Fault-tolerance schemes, Content Replication, Trust and Reputation
Censorship-Resistance	Sharing of copyright content, classified information, pornography	No major research-based solution proposed - legal recourse is only action as in case of Napster.
Lack of Centralized Control	Not possible to shutdown, not possible to control activities of individual peers	Distributed schemes for building trust and reputation, individualized security policies built into P2P middleware
Social Networking	Exploitation, Crime	None exist

Table 2. Summary of security threats and applicable solutions

The above-mentioned systems are based on the Grid Computing system model, with centralized elements such as data repositories, coordination managers and process workflow managers required to facilitate interactions between participating nodes. Hence, they do not qualify as pure decentralized P2P systems. Yet, some of the challenges such as security, resource provisioning and management, fault-tolerance for deployed applications etc. are common to both approaches.

The issues involved in P2P computing have been examined in detail in [75]. Early work in this regard was the SETIHOME [5] project of NASA, which attempted to utilize the idle CPU cycles available at computers connected to the internet, which could download data collected from NASA's radio telescopes in an attempt to search for signals indicating the existence of extra-terrestrial intelligence. Other schemes for harvesting idle CPU cycles include Condor [76] and Avaki [77]. Other P2P based computing frameworks include G2-P2P [78], CompuP2P [79] and P3 [80]. Many of these frameworks introduce some centralized elements to manage and locate distributed resources, scheduling of remote work, load balancing and collation of results. Also, these frameworks tend to rely on the self-organizing fault-tolerant nature of P2P networks without providing explicit fault-tolerance for the deployed applications, making simplifying assumptions regarding the frequency of node transience. The fault-tolerance issue has been addressed in [81] and overcomes one of the major hurdles in ensuring the widespread deployment of cycle-stealing P2P applications.

Scheduling in P2P distributed computation is a major area of focus for the research community, since it needs to take into account the availability of peers, their resource usage/availability, heterogeneity and load. For this reason most of the well-known schemes tend to be centralized in nature. For a purely decentralized setup, each peer would need to schedule its tasks independently and would lead to many complications.

Distributed Computation Issues	Proposed Solutions
Resource Location	Centralized Coordinator registers peers along with their available resources or a decentralized model in which peers advertise available resources and broadcast requests for required resources.
Remote Work Scheduling	Centralized scheduler looks at available peers meeting job execution resource requirements and which are lightly loaded or individual peers locate and schedule remote work at peers which respond to its broadcast request.
Node Transience/Network Churn/Fault-Tolerance	Checkpointing, Task Migration, Remote work submitted to multiple peers.
Validity of Returned Results	Trust and Reputation, Voting-based schemes trusting results from a majority of peers.
Security	Trust and Reputation, Role-Based Access Control, Authentication and Encryption, Membership Service for accessing compute resources in peer groups, Containment-based security models.

Table 3. Summary of issues involved in distributed computation and applicable solutions

A scheduling algorithm for high-performance P2P computing based on the XtremWebCH (XWCH) Global Computing platform is described in [82]. It relies on a central coordinator to accept job requests and schedule them on peers meeting the job execution requirements. A Wave-Scheduler which organizes the overlay topology according to geographic time-zones and exploits large chunks of idle CPU cycles during the night time to deliver better performance is proposed in [83].

To provide a balance between resource providers and resource consumers in distributed computing, an incentive-based scheduling algorithm is proposed in [84]. A detailed discussion on various scheduling strategies is provided in [85].

Other initiatives in distributed computation include creation of large-scale test-beds for deploying and testing real-world applications involving distributed computation. PlanetLab [86] is a research network comprising of over 1 000 nodes at over 450 locations worldwide, used to deploy network services, P2P applications and the like. OurGrid [87] is another open-source computational grid for running parallel applications.

Another focus area within the distributed computing space is the availability of application development frameworks for creating P2P-based applications. JXTA [10] is the most well-known application development framework providing primitives for the creation and maintenance of P2P networks, besides providing query handling mechanisms for location of content/resources via advertisements. Many applications have been built using the JXTA framework. More recently an application development framework has been proposed in [88].

3.6 P2P Economics

Since P2P systems involve resource exchanges and utilization of services, the proposal of several economic models represents a natural progression in the evolution of the P2P domain. An early discussion on the possible economic models in P2P and grid computing is available in [89]. The work classifies the economic models into the following categories:

- Commodity Market Model (service providers price their resources and charge users a fee as per usage of the resource; price depends on the demand-supply ratio for the resource)
- Posted Price Models (advertise special offers to attract customers)
- Bargaining Model (negotiation to get the best deal and optimize resources)
- Tendering/Contract-Net Model (resource/service users advertise requirements which are responded by potential service providers; a contract is agreed upon and fulfilled by both parties.)
- Auction Model (a one-to-many model allowing service provider to seek the best deal from many potential users)
- Bid-based Proportional Resource Sharing Model (resources allocated to a consumer are directly proportional to its bid, relative to bids of other consumers)
- Community/Coalition/Bartering Model (a community of users contribute resources and share the resources available in a group)
- Monopoly, Oligopoly (when one service provider exists and price is non-negotiable).

An evaluation of the feasibility of revenue models for P2P-based applications such as digital content sharing, instant messaging, collaboration, web services and distributed computation is available in [90]. The authors conclude that implementing a micro-payments facility for P2P services may not be feasible for pure decentralized P2P systems. A novel economic model for information exchange in mobile P2P networks is presented in [91]. This work integrates sensors into a P2P network of moving vehicles to exchange information such as available parking slots around current location, average speed and traffic conditions etc. PeerMart [92] creates a marketplace for P2P services where services are traded via a double auction, in a distributed manner. In PeerMart a peer can be a service provider, a service user or a broker. The broker is responsible for routing service requests to the appropriate provider. This is done by indexing the services and service providers over a structured DHT-based P2P topology. A monetary incentive based scheme has been proposed in [93], where peers are paid money for every forwarded message. The incentive combined with distributed reputation management ensures that the entire network works optimally. The financials involved in paying incentives are managed by the Credit Authority (CA) a centralized entity. MojoNation [34] is another interesting economic model where participating peers earn a notional currency "mojo" for services provided. The greater the "mojo" the greater are the services which the peer is entitled to utilize in the network. However, very little of this research has found its way into real-world applications and the P2P business model can be considered a non-starter. Some of the reasons for the failure of the P2P business model have been analyzed by Hughes et al. [94] and the following seven constraints have been identified:

- the technical constraint
- the economic constraint
- the cognitive constraint
- the structural constraint
- the legal constraint
- the political constraint
- the cultural constraint.

The challenges in building successful real world economic models for P2P systems are immense. Some of them are:

1. A credible system for accounting management and micro-payments for huge internet scale topologies in a completely decentralized and distributed manner is challenging to say the least.
2. Most of the P2P applications are content-sharing in nature. Since content gets replicated each time it is shared ensuring that the creator of the content gets duly compensated each time its content is shared is a big challenge. This and several related issues has been examined in detail by Lin [95].
3. P2P content-sharing applications are notorious for blatant sharing of copyright information such as music files. The global music industry has been lobbying for stricter copyright violation laws and hence the P2P business model involves many legal implications.
4. P2P networks are transient by nature and categorized by heavy network churn. The QoS parameters are pretty much non-existent in P2P networks, with peers

offering best-effort service. P2P applications would therefore need to be much more robust and offer minimum guarantees for service provisioning before a reliable payment mechanism is feasible.

5. Without a centralized accounting and auditing agency, any disputes arising out of service usage between two peers would be very hard to settle.

3.7 Resource Contribution

It is estimated that in P2P networks a majority of the peers utilize the resources on offer without offering resources/content in return. This behavior is referred to as "free-riding" and results in resource/content-providing nodes being swamped with lookup and download requests, creating the same bottlenecks associated with traditional client-server systems. The research community has been actively proposing schemes to counter free-riding. A good discussion on the issues of voluntary resource contributions, free-riding and encouraging peers to contribute resources is provided in [96]. Resource trading [97] between peers attempts to discourage free-riders from using resources of other peers without offering anything in return. Incentive based schemes [98] have been proposed which encourage peers to participate by offering resources in terms of information, storage and compute cycles in exchange for higher trust values and greater usage of resources of other peers. BitTorrent [7] is a polular file-sharing application which uses the tit-for-tat approach by matching peer capacities. A peer which contributes more resources to the network is typically serviced by a peer with higher or matching resources. A connection-management protocol promoting cooperation in P2P networks is proposed in [99], while a neighbor management scheme which monitors and identifies neighboring peers as free-riders and isolates them has been proposed in [100].

3.8 Novel P2P Applications

With the popularity of the P2P concept and some maturity in the domain, a new set of applications is being developed to take P2P beyond content and file sharing. One of the most interesting applications under development/testing currently is the Peer Phone [101], a P2P voice service, proposed by Swedish company TerraNet, in which individual phones are responsible for audio call-routing to neighboring peer phones, thereby creating a small network providing audio call services for communities without needing any additional infrastructure. Such an application is envisaged to have tremendous potential in third-world developing countries and can help bridge the digital divide.

P2P TV is another hot area currently, in which P2P applications seek to distribute multiple video streams through the overlay topology, supporting the video-on-demand functionality. Many established players including Microsoft have jumped onto the P2P TV bandwagon announcing their backing to the LiveStation [102] project. The European Union has decided to invest 14 million Euros on the P2P

Next [103] project which aims to build a P2P based internet television. Members of this group include BBC, European Broadcasting Union and Pioneer. Other known players include TVU Networks [104] and Sopcast [105] Early issues with P2P TV include its flaky video quality and requirement of a very high-bandwidth connection to watch uninterrupted videos.

3.9 General Trends

According to Wittenburg [106], the following general trends can be observed in the field of P2P networks/computing:

1. Emerging standards: Fundamental concepts have evolved and matured to an extent that they are now being translated into standards, which are expected to remain stable for some time to come. Middleware for instance, JXTA represents an important development in this direction.
2. Evolution of existing concepts: The fundamental concepts are being agreed upon by the community of researchers, developers and users and revolutionary changes to the P2P paradigm cannot be expected.
3. Analysis and performance testing: Performance metrics to evaluate various P2P approaches shall be increasingly standardized as a means of comparison between existing and new ideas.
4. Hype: There is some level of hype surrounding the P2P concept, with a general belief that all systems based on P2P are better.

Although the P2P concept represents a significant advance in distributed systems, it is being applied randomly to all application domains without evaluating the feasibility. Roussopoulos et al. [107] provide a decision tree to evaluate whether the P2P concept can be successfully applied to a particular application. The elements of the decision tree in order of relevance include:

- Budget (Low)
- Resource Relevance to Individual Peer (High)
- Mutual Trust Required (Low)
- Rate of Change (Low)
- Criticality (Low).

Generally, applications which require a high degree of trust deal with critical data or operate in environments where the rate of change is extremely high are not amenable to the P2P concept. Also, a higher budget usually indicates that a centralized infrastructure catering to all resource requirements can be created, alleviating the requirement for a P2P system model.

4 CHALLENGES

The following challenges/open-issues still persist in the P2P domain.

4.1 Topology Organization

1. Efficient dynamic organization of P2P overlay topology in the face of high node transience, i.e. when nodes join the network or leave the network frequently, while ensuring acceptable performance levels.
2. Reorganization of the overlay topology when the P2P network attains a critical mass, without impacting the network performance.
3. Strategies to select the best neighbor peer to attach to when new nodes join the network.
4. Selection of peers in an optimal manner to provide fault-tolerance and high-availability services to P2P applications, so as to reduce the message exchange overheads and improve performance.
5. Overlay topology optimization keeping in mind the underlying physical topology.

4.2 Search, Retrieval and Routing

1. Performance testing, analysis and optimization of existing schemes to help formulate benchmarks to evaluate new schemes/techniques against.
2. Techniques for maintaining consistency and efficiency of search indices in face of a highly transient population of nodes.
3. Formulation of new search techniques which take into account not only the information required by the peer, but also the requirement of compute resources (CPU cycles, storage, platform dependencies, etc.)
4. Formulation of search techniques which can utilize the underlying topology information to reduce unnecessary message forwarding and ensure faster retrieval of information.
5. Exploring feasibility of building routing strategies for P2P traffic into the physical network infrastructure, since P2P traffic already accounts for over 60 % of all internet traffic. This might allow more efficient routing of P2P traffic at the network layer compared to the application layer.

4.3 Resource Management

1. Formulation and validation of new frameworks which take into account sharing of idle compute resources (CPU, storage, peripherals etc.) and expert knowledge, promoting new avenues for collaboration and sharing.

2. Formulation of new negotiation mechanisms for sharing of computing resources between peers.
3. Ensuring fairness in sharing of resources.
4. Load balancing amongst peers in sharing of resources.
5. Formulation of new incentive schemes to discourage free-riding and promote greater contribution of resources from peers.
6. Schemes for decentralized, distributed scheduling and monitoring of remote work (work submitted by a remote peer) and ensuring its success in face of node transience.
7. Security issues in sharing computing resources and their solutions.
8. Evolving Quality of Service (QoS) parameters for peers regarding computing resource usage and ensuring compliance.

4.4 Security and Trust Management

1. Formulation of new trust management schemes when computing resources such as CPU cycles, storage and peripherals attached to a peer are shared. Such schemes would take into account the level of service provided by the peers and its reliability factor.
2. Evolution of new security frameworks for P2P systems, which allow P2P applications to be deployed within and across organizations which have stringent security policies regarding sharing of confidential data and critical resources.
3. Formulation of effective security mechanisms for the individual peer, alleviating the need for complex decentralized trust and reputation management schemes which add to the operational overheads of P2P networks.
4. Ensuring privacy, authentication, nonrepudiation and data integrity even when the participating peers are anonymous.

4.5 Legal and Social Issues

1. Formulation of technical strategies to prevent sharing of unauthorized information and protection of copyrighted information/intellectual property.
2. Lack of legislation in a majority of the countries regarding fair use of copyrighted material/information and existence of outdated IPR laws.
3. Dealing with the double-edged sword of "anonymity", which is a desirable property of P2P networks to avoid censorship and centralized control. However, anonymity also provides cover to indulge in illegal acts. The use of social networking sites by criminal elements, as seen in the kidnapping and murder of Adnan Patrawala [108] in Mumbai, India, exposes the dark side of this phenomenon. With 80 % of the users of popular social networking sites estimated to be young teens and children, who tend to use the anonymity offered by P2P

networks to lead a parallel digital existence, the possibility of their exploitation is real.

5 FUTURE DIRECTIONS

This section attempts to identify some areas within the P2P domain which could see some advances in the near future.

5.1 Extending the Power of P2P Across Enterprises

It is the view of the authors that advances in the P2P domain have focused mainly on the individual computers/users connected to the internet and have somehow left out the large enterprises, small and medium businesses and other organizations which also have requirements for information/data sharing, collaboration, scalable storage and specialized compute resources. Further the solution should be able to adapt to dynamically changing environments, be extremely fault-tolerant and resilient to security attacks requirements which are fulfilled in design by P2P systems.

Although the concept of cross-organizational interactions using a technology-based framework has been espoused earlier through concepts like desktop grids [73, 74] and virtual organizations [109, 110], these are essentially systems with centralized control and suffering from drawbacks such as single point of failure and lack of scalability.

Only recently have researchers begun to focus on the possible cross-enterprise P2P applications. Schroth et al. [111] talk about a purely decentralized P2P model for service composition and utilization in a cross-organization manner. Chen et al. [112] talk about a trust-based method for secure resource sharing across organizations, while Gupta et al. [113] talk about collaborative knowledge management using P2P-based technology. There is not enough literature promoting the use of the P2P concept for cross-organizational collaboration and to build applications spanning organizations. Neither has a comprehensive framework enabling content sharing, communication and distributed computation been proposed.

Thus, future work in the P2P domain should include P2P systems which cater to the specific requirements of the enterprise and which permit peer interactions across enterprises in a secure and transparent manner. This would enable the creation of new business models, enabling organizations to trade specialized resources, generate revenue by leasing out idle resources or charge for resource usage on a pay-per-use basis. This would greatly benefit the small business which may not have the necessary financial resources to create specialized compute resources. Such frameworks, if available can also have a deep impact in third-world countries where a majority of educational institutions are unable to provide access to the latest technologies to their students. By effectively leveraging the computer/human resources available at other organizations/institutions, the potential benefits for these countries, small organizations and institutions can only be imagined. Specific issues like organizational computer resource aggregation and location, resource management, fault-tolerance and security would however need to be resolved before such systems can be envisaged. Details of preliminary work done in this area can be found in [114].

5.2 Beyond File/Content Sharing

Existing P2P research has focused on addressing issues related mostly to content sharing systems with only some research addressing the issue of active distributed work management amongst peers. Specific issues like security when peers share CPU cycles and storage, need addressing by the research community in order to build comprehensive P2P systems. Also, remote work distribution and management, distributed scheduling, fault-tolerance etc. become relevant research issues for such systems. For P2P systems to be transformed from being perceived as providing best-effort service to providing tangible Quality-of-Service, issues like security and fault-tolerance need to be addressed urgently. Peers-for-Peers (P4P) [81] is one such scheme for achieving fault-tolerance for cycle-stealing P2P applications.

5.3 Quality-of-Service (QoS) in P2P

P2P networks have traditionally been considered too transient in nature to perform useful computations for real-world applications, leave aside the formulation of a viable QoS framework. The only mature application that P2P networks successfully cater to, are those related to content sharing, where the sheer scale of P2P networks along with strategies for content-caching and replication enable content to be located and downloaded in a time-bound manner. A few attempts have been made by researchers to propose QoS parameters for P2P applications, but they have been limited to content-sharing applications. Specific QoS parameters have been proposed for:

1. guaranteed content location and retrieval (if it exists) – query success
2. time-bound content location and retrieval – query performance
3. correct content retrieval – content quality
4. video streaming and multicasting – delay, bandwidth and jitter.

A good discussion on QoS pertaining to content management for P2P file-sharing applications is provided by Meo & Milan [115], wherein a QoS policy at individual peers works out the best content management strategy for ensuring high-availability of content and its assured retrieval. QCast [116] is an example of a DHT-based multicasting strategy which complies with QoS parameters for multi-media streaming. Participating peers first organize themselves into a distribution tree according to their buffer sizes and bandwidths to create and end-to-end distribution path for streaming. This approach attempts to optimize the delay and reduce the packet loss. The issue of ensuring QoS for media streaming in the face of rapidly changing network topology and communication channels is addressed by Nemati & Takizawa [117]. The strategy selects multiple sources which can meet the QoS requirements of the receiver. As the QoS from a particular source drops below a certain threshold, another source steps in to ensure QoS compliance.

Hardly any QoS parameters have been proposed for P2P applications involving distributed computation, simply because frequent node-transience makes the task of distributed computation extremely challenging. This is especially true for applications on the open internet. SETI@HOME remains the only real-world distributed computing P2P application utilizing the computation power of millions of PCs connected to the internet. For a more viable computing model researchers have focused on federated P2P systems (those which operate within a domain) and successful examples like G2-P2P Condor, Avaki, CompuP2P and P3 etc. do exist for such systems. However, these applications make over-simplifying assumptions such as little or no node transience. Also, these applications make no mention of Quality-of-Service for distributed computation performed. Hence, proposing a generic QoS model for P2P applications is venturing into unchartered territory.

Figure 4 provides an indicative, unvalidated QoS model for P2P applications. Clearly, performance of any P2P application can emerge only if the underlying compute resources are available and dependably so. Hence, any move towards providing quality assurances or entering into Service-Level-Agreements (SLAs) with other peers or organizations, is feasible only if basic assurances regarding the general availability of resources and their dependability can be secured.

Although P2P systems may never match the high-performance computing systems such as dedicated grids, they can surely provide acceptable levels of performance by controlling some aspects of their adhoc nature and making provisions for security and fault-tolerance.

5.4 New Security Paradigms

Several existing P2P networks/applications utilize encryption for ensuring data privacy. Trust and reputation based systems have also been proposed to isolate malicious peers. However, the overhead of trust computation and communication to other peers throughout the P2P network is far from efficient. Also, several scenarios where groups of malicious peers connive to create artificially high trust values for some peers complicate the trust and reputation computations. Ideally peers should be able to blindly trust all other peers, but have a comprehensive security mechanism which is fool-proof and allows the peer to engage in resource sharing without paying the overheads for trust management schemes. The Entropia DC Grid [73] provides a novel solution based on binary-sandboxing of the remote application. A binary submitted for execution is modified to intercept the system calls and control its interaction with the operating system and its resources as per the defined security policies. The system however provides no control over the quantum of resource usage. A containment-based security model for individual peers [118] has been proposed which addresses the shortcomings of traditional trust and reputation

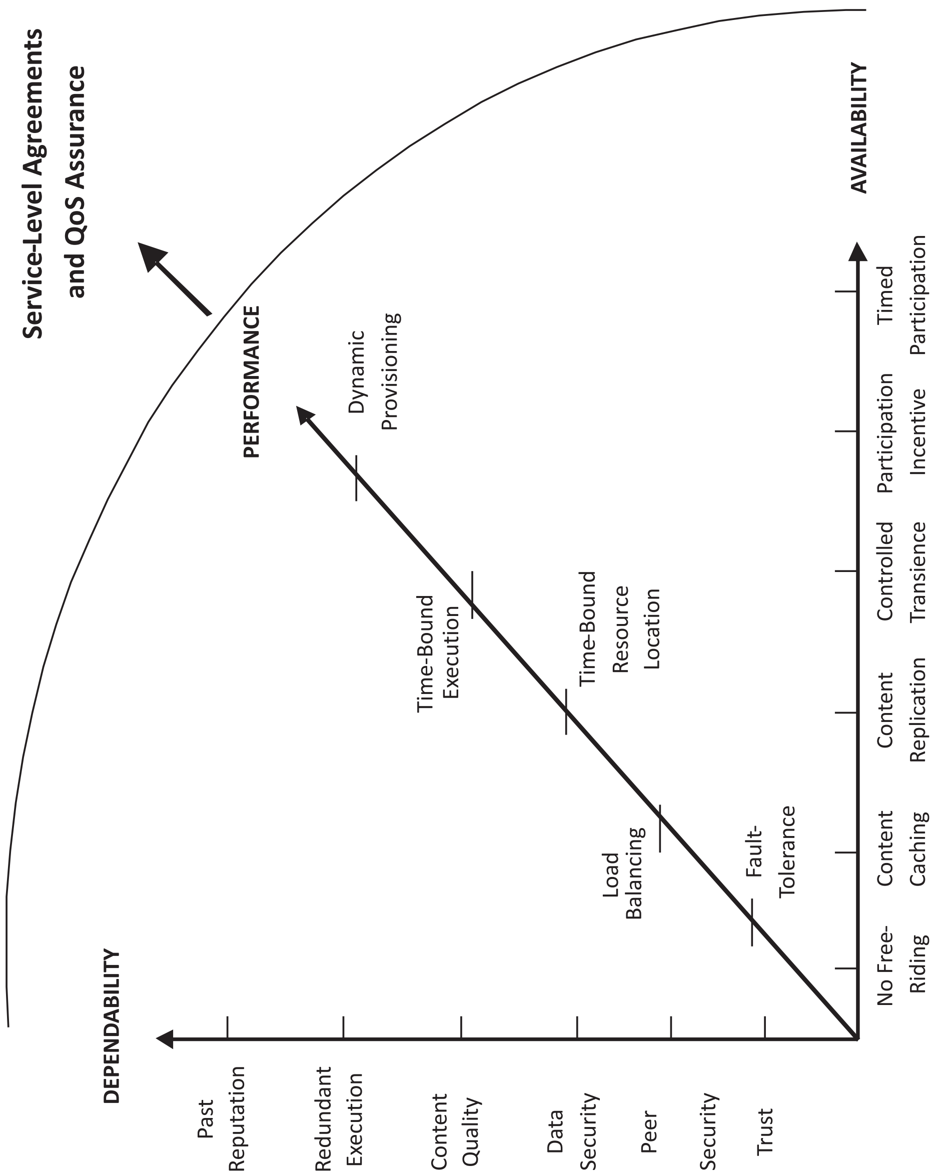


Fig. 4. A generic QoS model for P2P applications

management schemes. Such a containment model, utilizing fine-grained privileges and access control techniques, limits the activities carried out by remote applications utilizing the host peer resources and ensures its security in all scenarios. Moreover, it slays remote applications if they violate any resource usage quotas specified by the host peer, allowing run-time monitoring for any malicious behavior.

5.5 Countering the Ills of P2P Social Networking

The social networking phenomenon spawned by many P2P applications such as FaceBook etc., is fast becoming a playground for sexual predators and criminals who take advantage of the millions of vulnerable users subscribing to these applications. The anonymity, lack of censorship and centralized control afforded by such P2P applications, also cause wide-spread misuse. The research community would do well to build P2P networks with intelligent peers having a "conscience", who would not store or forward objectionable content, scan shared content being routed through the overlay topology for viruses and spam and effectively identify and block out malicious peers. Such strategies could be built into the P2P middleware, without affecting the nature of P2P interactions significantly.

5.6 Standardization of P2P Protocols

Currently, there are many P2P middleware available; probably more than there are applications, each suited to the kind of application it supports. Moreover, interoperability between heterogeneous P2P networks would also become an issue in the near future. The time is right for an organized shift in the P2P community towards standardization of protocols, frameworks to build and deploy P2P systems, performance metrics and benchmarks and mechanisms for integrating heterogeneous P2P systems. This will allow rapid application development in the P2P domain and spawn a whole new community of P2P developers.

5.7 P2P and the Cloud

Cloud computing [119] is the new buzzword for describing an internet-based computing infrastructure which encompasses a variety of domains such as grid computing, cluster computing, utility computing, pervasive computing and on-demand computing. It represents a partnership between legacy computing infrastructure and accessing the resources available on the web. The cloud consists of a multitude of data-centers offering virtualized servers and storage allowing clients access to on-demand computing, with detailed Service Level Agreements (SLA's). The potential of cloud computing can be gauged from the fact that industry big-wigs Sun, Microsoft, IBM and Google have jumped on the cloud computing bandwagon and are in the process of proposing their own architectures and application development platforms.

Although the P2P concept has not yet been applied to specific issues involved in cloud computing, there is a strong case where the scalability and natural fault-tolerance offered by P2P networks is a perfect fit for the cloud computing paradigm. Otherwise the cloud computing paradigm suffers the risks involved in traditional client-server computing model single-point-of-failure and performance bottlenecks, the same issues which necessitated the evolution of the P2P paradigm.

5.8 The Future Shock

The authors expect the next generation of P2P systems to be completely autonomous in nature, with the ability of executing complex tasks independently and in an adaptive manner based on peer interactions. This will result in the creation of intelligent systems built on P2P architectures, which shall be able to achieve shared objectives independently without user intervention. Also, the desirable properties of the P2P concept will attract more and more developers to develop the next killer applications in the fields of P2P gaming, e-commerce, network management, data management and expert systems for collaborative software development and maintenance systems and the like.

6 CONCLUSION

The P2P concept holds great potential in designing and building complex software systems of the future, which shall integrate many different domains from distributed autonomic computing to intelligent systems. Before that the community of researchers and developers need to alleviate the shortcomings of existing approaches like security, resource management and non-standardized protocols and come up with new strategies and techniques to enable the development and deployment of next generation of P2P applications.

REFERENCES

[1] Androutsellis-Theotokis, S.—Spinellis, D.: A Survey of Peer-to-Peer File Sharing Technologies. White paper, `http://citeseer.ist.psu.edu/androutsellistheoto02survey.html`.
[2] Napster Website: `http://www.napster.com`.
[3] Gnutella Website: `http://www.gnutella.wego.com`.
[4] Gnutella Website: `http://www.gnutella.wego.com`.
[5] SETI@Home Website: `http://setiathome.berkeley.edu`.
[6] OceanStore Website: `http://oceanstore.cs.berkeley.edu`.
[7] KaZaA Website: `http://www.kazaa.com/us/index.htm`.
[8] BitTorrent Website: `http://www.bittorrent.com`.
[9] CoolStreaming Website: `http://www.coolstreaming.us`.

[10] Facebook website: `http://www.facebook.com`.

[11] YouTube website: `http://youtube.com`.

[12] Cisco Visual Networking Index. Global Mobile Data Traffic Forecast Update. White Paper, `http://www.cisco.com/en/US/solutions/collateral/ns341/ns525/ns537/ns705/ns827/whitepaperc11-520862.html`.

[13] JXTA HomePage: `http://www.sun.com/jxta`.

[14] Ratnasamy, S.—Francis, P.—Handley, M.—Karp, R.—Shenker, S.: A Scalable Content-Addressable Network. In ACM SIGCOMM 2001 Conf. on Applications, Technologies, Architectures, and Protocols for Computer Communication, 2001, pp. 161–172.

[15] Saia, J.—Fiat, A.—Gribble, S.—Karlin, A.—Saroiu, S.: Dynamically Fault-Tolerant Content Addressable Networks. Lecture Notes in Computer Science, Vol. 2429, 2002, pp. 270–279.

[16] Plaxton, C. G.—Rajaraman , R.—Richa, A. W.: Accessing Nearby Copies of Replicated Objects in a Distributed Environment. Proceedings of the Ninth Annual ACM Symposium on Parallel Algorithms and Architectures, June 23–25, 1997, pp. 311–320.

[17] Datar, M.:. Butterflies and peer-to-peer networks. In ESA 2002, 10$^{\text{th}}$ Annual European Symposium, 2002, pp. 310–322.

[18] Malkhi, D.—Naor, M.—Ratajczak, D.: Viceroy: A Scalable and Dynamic Emulation of the Butterfly. In Proc. ACM SIGACT-SIGOPS, 21$^{\text{st}}$ Symp. on the Principles of Dist. Comp., 2002, pp. 183–192.

[19] Crainiceanu, A.—Linga, P.—Gehrke, J.—Shanmugasundaram, J.: Querying Peer-to-Peer Networks Using P-Trees. In Proc. 7$^{\text{th}}$ Int. Workshop on the World Wide Web and Databases (WebDB), 2004, pp. 25–30.

[20] Jagadish, H. V.—Ooi, B. C.—Vu. Baton, Q. H.: : A Balanced Tree Structure for Peer-to-Peer Networks. In Proc. 31$^{\text{th}}$ Int. Conf. on Very Large Data Bases, 2005, pp. 661–672.

[21] Stoica, I.—Morris, R.—Liben-Nowell, D.—Karger, D. R.—Kaashoek, M. F.—Dabek, F.—Balakrishnan, H.: Chord: A Scalable Peer-to-Peer Lookup Protocol for Internet Applications. IEEE/ACM Transactions on Networking, Vol. 11, 2003, No. 1, pp. 17–32.

[22] Rowstron, A.—Druschel, P.: Pastry: Scalable, Decentralized Object Location, and Routing for Large-Scale Peer-to-Peer Systems. Lecture Notes in Computer Science, Vol. 2218 2001, pp. 329.

[23] Zhao, B.—Kubiatowicz, J.—Joseph, A.: Tapestry: An Infrastructure for Fault-Tolerant Wide-Area Location and Routing. Tech. Rep. UCB/CSD-01–1141, Computer Science Division, U. C. Berkeley, April 2001.

[24] Malkhi, D.—Naor, M.—Ratajczak, D.: Viceroy: A Scalable and Dynamic Emulation of the Butterfly. In ACM Principles of Distributed Computing, July 2002, pp. 183–192.

[25] Maymounkov, P.—Mazières, D.: Kademlia: A Peer-to-Peer Information System Based on the XOR Metric. In International Workshop on Peer-to-Peer Systems, February 2002, pp. 5365.

[26] Kaashoek, M. F.—Karger, K. D.: Koorde: A Simple Degree-Optimal Distributed Hash Table. Lecture Notes in Computer Science, Vol. 2735, Oct. 2003, pp. 98–107.

[27] De Bruijn, N.: A Combinatorial Problem. In Proc. Koninklijke Nederlandse Akademie van Wetenschappen, Vol. 49, 1946, pp. 758–764.

[28] Eng Keong Lua: Hierarchical Peer-to-Peer Networks Using Lightweight SuperPeer Topologies. Proceedings of the 10^{th} IEEE Symposium on Computers and Communications, 2005, pp. 143–148.

[29] Peng, Zh.—Duan, Zh.—Qi, J. J.—Cao, Y.—Lv, E.: HP2P: A Hybrid Hierarchical P2P Network. First International Conference on the Digital Society, Jan. 2007, pp. 8–18.

[30] Risson, J.—Harwood, A.—Moors, T.: Stable High-Capacity One-Hop Distributed Hash Tables. 11^{th} IEEE Symposium on Computers and Communications, 2006, pp. 687–694.

[31] Lua, K.—Crowcroft, J.—Pias, M.—Sharma, R.—Lim, S.: A Survey and Comparison of Peer-to-Peer Overlay Network Schemes. IEEE Communications Surveys & Tutorials, 2005, pp. 72–93.

[32] Vishnumurthy, V.—Francis, P.: On Overlay Construction and Random Node Selection in Heterogeneous Unstructured P2P Networks. In IEEE International Conference on Computer Communications, April 2006, pp. 1–12.

[33] Wan, H.—Ishikawa, N.—Hjelm, J.: Autonomous Topology Optimization for Unstructured Peer-to-Peer Networks. In Proceedings of the 11^{th} International Conference on Parallel and Distributed Systems, 2005, pp. 488–494.

[34] Robert, A. G.—Fang, W.—Yaoru, S.:, Self-Organizing and Adaptive Peer-to-Peer Network. IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics, Vol. 36, 2006, No. 6, pp. 1230–1236.

[35] Jagadish, H. V.—Ooi, B. C.—Vu, Q. H.—Zhou, A. Y.—Zhang, R.:. VBI-Tree: A Peer-to-Peer Framework for Supporting Multi-Dimensional Indexing Schemes. In Proc. Int. Conf. on Data Engineering, April 2006, pp. 34.

[36] Hui, K. Y. K.—Lui, J. C. S.—Yau, D. K. Y.: Small-World Overlay P2P Networks: Construction, Management and Handling of Dynamic Flash Crowds. Computer Networks: The International Journal of Computer and Telecommunications Networking, Vol. 50, 2006, No. 15, pp. 2727–2746.

[37] Winter, R.—Zahn, T.—Schiller, J.: Random Landmarking in Mobile, Topology-Aware Peer-To-Peer Networks. In IEEE Workshop on Future Trends of Distributed Computing Systems, 2004, pp. 319–324.

[38] Karakaya, M.—Körpeoglu, I.—Ulusoy, Ö.: A Connection Management Protocol for Promoting Cooperation in Peer-to-Peer Networks. Computer Communications, Vol. 31, 2008, No. 2, pp. 240–256.

[39] Horiuchi, H.—Wakamiya, N.—Murata, M.: A Network Construction Method for a Scalable P2P Video Conferencing System. In Proceedings of the IASTED European Conference: Internet and Multimedia Systems and Applications, 2007, pp. 196–201.

[40] MojoNation 2003. The MojoNation web site. `http://www.mojonation.net`.

[41] Clarke, I.—Sandberg, O.—Wiley, B.: Freenet: A Distributed Anonymous Information Storage and Retrieval System. In Workshop on Design Issues in Anonymity and Unobservability. 2001, pp. 46–66.

[42] Lv, Q.—Cao, P.—Cohen, E.—Li, K.—Shenker, S.: Search and Replication in Unstructured Peer-to-Peer Networks. In 16th ACM International Conference on Supercomputing, 2002, pp. 84–95.

[43] Yang, B.—Garcia-Molina, H.: Improving Search in Peer-to-Peer Networks. In Proceedings of the 22nd International Conference on Distributed Computing Systems, 2002, pp. 5.

[44] Kalogeraki, V.—Gunopoulos, D.—Zeinalipouryazti, D.: A Local Search Mechanism for Peer-To-Peer Networks. In Proceedings of the 11th International Conference on Information and Knowledge Management, 2002, pp. 300–307.

[45] Park, K.—Pack, S.—Kwon, T.: Proximity Based Peer-to-Peer Overlay Networks (P3ON) with Load Distribution. Lecture Notes in Computer Science, Vol. 5200, 2008, pp. 334–343.

[46] Wu, Z. D.—Rao, W. X.—Ma, F. Y.: Super Proximity Routing in Structured P2P Overlay Networks. Journal of Zhejiang University of Science, Vol. 5, Jan. 2004, No. 1, pp. 16–21.

[47] Ratnasamy, S.—Handley, M.—Karp, R.—Shenker, S.:. Topologically-Aware Overlay Construction and Server Selection. In Proc. 21st IEEE INFOCOM, June 2002.

[48] Dong, B.—Chen, J. H.—Feng, R.—Sun, Y.-M.: ROSE: Large-Scale Content-Based Routing for P2P Networks. International Symposium on Information Science and Engineering, Vol. 2, 2008, pp. 384–388.

[49] Castelli, S.—Costa, P.—Picco, G. P.: HyperCBR: Large-Scale Content-Based Routing in a Multidimensional Space. IEEE Conference on Computer Communications, April 2008, pp. 1714–1722.

[50] Maxim, R.—Hui, S. C.: Intelligent Content-Based Retrieval for P2P Networks. International Conference on Cyberworlds, 2003, pp. 318–325.

[51] Yang, B.—Vinograd, P.—Garcia-Molina, H.: Evaluating GUESS and Non-Forwarding Peer-to-Peer Search. In International Conference on Distributed Computing Systems, 2004, pp. 209–218.

[52] Chen, Sh.—Zhang, Zh.—Chen, Sh.—Shi, B.: Efficient File Search in Non-DHT P2P Networks. Computer Communications, Vol. 31, 2008, No. 2, pp. 304–317.

[53] Mashayekhi, H.—Habibi, J.—Rostami, H.: Efficient Semantic Based Search in Unstructured Peer-to-Peer Networks. Second Asia International Conference on Modeling & Simulation, May 2008, pp. 71–76

[54] Vishnevsky, V.—Safonov, A.—Yakimov, M.—Shim, E.—Gelman, A. D.: Scalable Blind Search and Broadcasting Over Distributed Hash Tables. Computer Communications, Vol. 31, 2008, No. 2, pp. 292–303.

[55] Schütt, T.—Schintke, F.—Reinefeld, A.: Range Queries on Structured Overlay Networks. Computer Communications, Vol. 31, 2008, No. 2, pp. 280–291.

[56] González-Beltrán, A.—Milligan, P.—Sage, P.: Range Queries over Skip Tree Graphs. Computer Communications, Vol. 31, No. 2, pp. 358–374.

[57] Tran, D. A.—Nguyen, T.: Hierarchical Multidimensional Search in Peer-to-Peer Networks. Computer Communications, Vol. 31, 2008, No. 2, pp. 346–357.

[58] Zhao, B. Y.—Duan, Y.—Huang, L.—Joseph, A.—Kubiatowicz, J.: Brocade: Landmark Routing on Overlay Networks. In International Workshop on Peer-to-Peer Systems, March 2002, pp. 34–44.

[59] Harvey, N. J. A.—Jones, M. B.—Saroiu, S.—Theimer, M.—Wolman, A.: Skipnet: A Scalable Overlay Network with Practical Locality Properties. In Proc. USITS, March 2003, pp. 113–126.

[60] Xu, J.—Zegura, E. W.—Kumar, A.: Efficient and Scalable Query Routing for Unstructured Peer-to-Peer Networks. In Proceedings of IEEE Infocom 2005, pp. 1162–1173.

[61] Gatani, L.—Lo Re, G.—Gaglio, S.: An Adaptive Routing Mechanism for P2P Resource Discovery. In IEEE International Symposium of Cluster Computing and the Grid, 2005, pp. 205–212.

[62] Shi, C.—Han, D.—Liu, Y.—Meng, Sh.—Yu, Y.: A Dynamic Routing Protocol for Keyword Search in Unstructured Peer-to-Peer Networks. Computer Communications, Vol. 31, 2008, No. 2, pp. 318–33.

[63] Kamwar, S. D.—Schlosser, M. T.—Garcia-Molina, H.: The EigenTrust Algorithm for Reputation Management in P2P Networks. In Proceedings of the 12^{th} International Conference on World Wide Web, May 2003, pp. 640–651.

[64] Singh, A.—Liu, L.: TrustMe: Anonymous Management of Trust Relationships in Decentralized P2P Systems. In Proceedings of the Third International Conference on Peer-to-Peer Computing, September 2003, pp. 142–149.

[65] Marti, S.—Garcia-Molina, H.: Taxonomy of Trust: Categorizing P2P Reputation Systems. Computer Networks, Special Issue on Trust and Reputation in Peer-to-Peer Systems, Vol. 50, 2006, No. 4, pp. 472–484.

[66] Lesueur, F.—Me, L.—Tong, V. V. T.: A Sybilproof Distributed Identity Management for P2P Networks. IEEE Symposium on Computers and Communications, July 6–9, 2008, pp. 246–253.

[67] Gupta, R.—Manion, T. R.—Rao, R. T.—Singhal, S. K.: Peer-to-Peer Authentication and Authorization. United States Patent: 7350074.

[68] Kim, Y.—Mazzocchi, D.—Tsudik, G.: Admission Control in Peer Groups. Second IEEE International Symposium on Network Computing and Applications, April 2003, p. 131.

[69] Tran, H.—Hitchens, M.—Varadharajan, V.—Watters, P.: A Trust Based Access Control Framework for P2P File-Sharing Systems. In Proc. International Conference on System Sciences, 2005, pp. 302.

[70] Park, J. S.—An, G.—Chandra, D.: Trusted P2P Computing Environments with Role-Based Access Control. Information Security, IET Volume 1, Issue 1, March 2007, pp. 27–35.

[71] Gaspary, L. P.—Barcellos, M. P.—Detsch, A.—Antunes, R. S.: Flexible Security in Peer-to-Peer Applications: Enabling New Opportunities Beyond File Sharing. Comput. Networks 51, 2007, pp. 4797–4815.

[72] DataSynapase: Dynamic Application Service Management. Website: `http://www.datasynapse.com`.

[73] Calder, A.—Chien, A. A.—Wang, J.—Yang, D.: The Entropia Virtual Machine for Desktop Grids. 1st ACM/USENIX International Conference on Virtual Execution Environments, 2005, pp. 186–196.

[74] Boinc: Volunteer Computing. Website: `http://boinc.berkeley.edu`.

[75] Milojicic, D. S.—Kalogeraki, V.—Lukose, R.—Nagaraja, K.—Pruyne, J.—Richard, B.—Rollins, S.—Xu, Z.: Peer-to-Peer Computing. HP Laboratories, Palo Alto, HPL-2002–57, March 2002.

[76] Avaki Website. `http://www.avaki.com`.

[77] Condor Website. `http://www.wisc.com/condor`.

[78] Kelly, W.—Mason, R.: G2-P2P: A Fully Decentralized Fault-Tolerant Cycle Stealing Framework. In Australian Workshop on Grid Computing and e-Research, 2005, pp. 33–39.

[79] Gupta, R.—Sekhri, V.—Somani, A. K.: CompuP2P: An Architecture for Internet Computing Using Peer-to-Peer Networks. In: IEEE Transactions on Parallel and Distributed Systems, Vol. 17, 2006, No. 11, pp. 1306–1320.

[80] Shudo, K.—Tanaka, Y.—Sekiguchi, S.: P3: P2P-Based Middleware Enabling Transfer and Aggregation of Computational Resources. IEEE International Symposium on Cluster Computing and the Grid, May 2005, pp. 259–266.

[81] Gupta, A.—Awasthi, L. K.: P4P: Ensuring Fault Tolerance for Cycle-Stealing P2P Applications. International Conference on Grid Computing and Applications (GCA), June 2007, pp. 151–155.

[82] Abdennadher, N.—Boesch, R.: A Scheduling Algorithm for High-Performance Peer-to-Peer Computing Platform. European Conference on Parallel Computing 2006.

[83] Lo, V.—Zhou, D.—Liu, Y.—Zhao, Sh.: Cluster Computing on the Fly: P2P Scheduling of Idle Cycles in the Internet. Proc. of the 3rd International Workshop on Peer-to-Peer Systems, 2004, pp. 227–236.

[84] Zhu, Y.—Xiao, L.—Ni, L. M.—Xu, Zh.: Incentive-Based P2P Scheduling in Grid Computing. Lecture Notes in Computer Science, Vol. 3251, 2004, pp. 209–216.

[85] Liu, F.: Distributed Task Scheduling Algorithms in Peer-to-Peer High-Performance Computing.

[86] PlanetLab Website: `http://www.planet-lab.org`.

[87] OurGrid Website: `http://www.ourgrid.org`.

[88] Walkerdine, J.—Hughes, D.—Rayson, P.—Simms, J.—Gilleade, K.—Mariani, J.—Sommerville, I.: A Framework for P2P Application Development. Computer Communications, Vol. 31, 2008, pp. 387–401.

[89] Buyya, R.—Stockinger, H.—Giddy, J.—Abramson, D.: Economic Models for Management of Resources in Peer-to-Peer and Grid Computing. International Symposium on the Convergence of Information Technologies and Communications 2001.

[90] Hummel, T.—Stramme, O.—La Salle, R. M.: Earning a Living Among Peers – The Quest for Viable P2P Revenue Models. International Conference on System Sciences, Jan. 2003, pp. 10.

[91] Wolfson, O.—Xu, B.—Sistla, A. P.: An Economic Model for Resource Exchange in Mobile Peer to Peer Networks. Proceedings of the 16th International Conference on Scientific and Statistical Database Management 2004, pp. 235.

[92] Hausheer, D.—Stiller, B.: PeerMart: The Technology for a Distributed Auction-Based Market for P2P Services. IEEE International Conference on Communications, May 2005, Volume 3, pp. 1583–1587.

[93] Wongrujira, K.—Seneviratne, A.: Monetary Incentive with Reputation for Virtual Market-Place Based P2P. 2005 ACM Conference on Emerging Network Experiment and Technology, 2005, pp. 135–145.

[94] Hughes, J.—Lang J. R.—Vragov, R.: An Analytical Framework for Evaluating Peer-to-Peer Business Models. Electronic Commerce Research and Applications, Vol. 7, 2008, No. 1, pp. 105–118.

[95] Lin, F.—Lo, H. M.—Wang, C.: Can a P2P File-Sharing Network Become an e-Marketplace? International Conference on System Sciences, January 2008, pp. 298.

[96] Feldman, M.—Chuang, J.: Overcoming Free-Riding Behavior in Peer-to-Peer Systems. ACM SIGecom Exchanges, Volume 5, I2005, No. 4, pp. 41–50.

[97] Cooper, B. F.—Garcia-Molina, H.: Peer-to-Peer Resource Trading in a Reliable Distributed System. Lecture Notes in Computer Science, Vol. 2429, 2002, pp. 319–327.

[98] Sun, Q.—Garcia-Molina, H.: SLIC: A Selfish Link-Based Incentive Mechanism for Unstructured Networks. In International Conference on Distributed Computing Systems, 2004, pp. 506–515.

[99] Karakaya, M.—Körpeoglu, I.—Ulusoy, Ö.: A Connection Management Protocol for Promoting Cooperation in Peer-to-Peer Networks. Computer Communications, Vol. 31, 2008, No. 2, pp. 240–256.

[100] Karakaya, M.—Körpeoglu, I.—Ulusoy, Ö.: Counteracting Free Riding in Peer-to-Peer Networks. Computer Networks: The International Journal of Computer and Telecommunications Networking, Vol. 52, 2008, No. 3, pp. 675–694.

[101] TerraNet Website: `http://www.terranet.se`.

[102] LiveStation Website: `http://www.livestation.com`.

[103] P2P Next Website: `http://www.p2pnext.org`.

[104] TVU Networks Website: `http://www.tvunetworks.com`.

[105] Sopcast Website: `http://www.sopcast.com`.

[106] Trends in P2P Networks and Computing. Wittenburg, Website: `http://page.mi.fuberlin.de/wittenbu/studies/p2ptrends.pdf`.

[107] Roussopoulos, M.—Baker, M.—David, S.—Rosenthal, H.—Giuli, T. J.—Maniatis, P.—Mogul, J.: 2 P2P or Not 2 P2P? In Proceedings of International Workshop on Peer-to-Peer Systems (IPTPS), February 2004, pp. 33–43.

[108] Adnan Patrawala Murder Case. News Story, Website: `http://www.ndtv.com/convergence/ndtv/processarchive.aspx?id=NEWEN20070023393`.

[109] Wikipedia Definition of Virtual Organization. `http://en.wikipedia.org/wiki/Virtualorganization`.

[110] Ripeanu, M.—Singh, M. P.—Vazhkudai, S. S.: Virtual Organizations. Guest Editorial, IEEE Internet Computing, Vol. 12, 2008, No. 2, pp. 10–12.

[111] Schroth, C.—Janner, T.—Hoyer, V.: Strategies for Cross-Organizational Service Composition. International MCETECH Conference One-Technologies, Jan. 2008, pp. 93–103.

[112] Chen, T.—Chen, Y.—Wang, C.—Chu, H.—Yang, H.: Secure Resource Sharing on Cross-Organization Collaboration Using a Novel Trust Method. Robotics and Computer-Integrated Manufacturing, Vol. 23, 2007, No. 4, pp. 421–435.

[113] Gupta, S.—Robert, B.: Using Peer-to-Peer Technology for Collaborative Knowledge Management: Concepts, Frameworks and Research Issues. Knowledge Management Research & Practice, Vol. 4, 2006, No. 3, pp. 187–196.

[114] Gupta, A.—Awasthi, L. K.: Peer Enterprises: Possibilities, Challenges and Some Ideas Towards Their Realization. LNCS 4806, November 2007, pp. 1011–1020.

[115] Meo, M.—Milan, F.: QoS Content Management for P2P File-Sharing Applications. Future Generation Computer Systems, Vol. 24, 2008, No. 3, pp. 213–221.

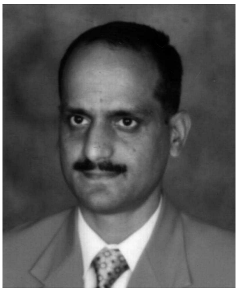
[116] Nemati, A. G.—Takizawa, M.: Application Level QoS for Multi-Media Peer-to-Peer Networks. International Conference on Advanced Information Networking and Applications, 2008, pp. 319–324.

[117] Cai, Z.—Lin, X.: QCast: A QoS-Aware Peer-to-Peer Streaming System with DHT-Based Multicast. Lecture Notes in Computer Science, Vol. 5036, 2008, pp. 287–295.

[118] Gupta, A.—Awasthi, L. K.: Secure Thyself: Securing Individual Peers in Collaborative Peer-to-Peer Environments. International Conference on Grid Computing and Applications (GCA), July 2008, pp. 140–146.

[119] Cloud Computing Definition. Wikipedia Website: `http://en.wikipedia.org/wiki/Cloudcomputing`.

Ankur Gupta is the Joint Director at the Model Institute of Engineering and Technology, Jammu, India. He received his B. E. (Hons) Computer Science and M. Sc. (Software Systems) degrees from BITS, Pilani, India and his Ph. D. from the National Institute of Technology. His research interests include network management, P2P networks/computing and software engineering. He has 2 patents pending at the USPTO, 2 filed patents at the Indian Patents Office and a defensive publication to his name. Prior to joining the academia, he worked with Hewlett-Packard in various technical positions for 7 years. He has published over 30 peer-reviewed research papers in various international journals and conferences.



Lalit K. Awasthi is a Professor of Computer Science and Engineering at the National Institute of Technology, Hamirpur, India. He received his Ph. D. degree from Indian Institute of Technology, Roorkee and M. Tech. in Computer Science from Indian Institute of Technology, Delhi. His research interests include mobile distributed systems, checkpointing based fault tolerance, ad hoc networks and P2P networks/computing. He has published over 70 peer-reviewed research papers in various international journals and conferences.