Location Services in Wireless Ad Hoc and Hybrid Networks: A Survey*

Roy Friedman Gabriel Kliot
Department of Computer Science
Technion
Haifa, 32000, Israel

April 24, 2006

Abstract

Location services are used in mobile ad hoc and hybrid networks either to locate the geographic position of a given node in the network or for locating a data item (content). One of the main usages of *position location services* is in location based routing algorithms. In particular, geographic routing protocols can route messages more efficiently to their destinations based on the destination node's geographical position, which is provided by a location service. A *content location service* provides to the requesting node either the requested data itself or the identifier of the node that stores this data. Sometimes the position of the node that stores the data is also provided. Such data location services are useful for implementing content-sharing applications, cooperative caching, and publish subscribe systems. In this paper we present a taxonomy of location services and survey known techniques for constructing such a service in wireless ad hoc and hybrid networks.

1 Introduction

Mobile wireless ad hoc networks consist of wireless hosts that communicate with each other in the absence of a fixed infrastructure. Some examples of the possible uses of ad hoc networking include soldiers on the battlefield, emergency disaster relief personnel, and networks of laptops. *Hybrid wireless networks* are networks that are formed by a collection of wireless access points and mobile wireless enabled computing devices. The mobile devices can communicate to each other either through the access points closest to them, as well as on a peer-to-peer basis by forming wireless ad hoc networks.

Two main usages of location services in mobile ad hoc networks are to provide information regarding a geographic location of the nodes in the network and for locating a content. Although used for different purposes, both types of location services have a significant amount of similarity in terms of implementation and are therefore studied together in this work.

Yet, despite their similarity, there are a number of differences between position and content location services:

*This work was funded by France Telecom R&D

Use of geographical knowledge: Position location services usually utilize the geographical position of nodes (inherently available to it) in order to implement the service itself, while data location services may not necessarily possess such an information.

Sensitivity to mobility: In position location services, the service has to be responsive to nodes' mobility. When a node moves, its position changes and the service must be updated according to the mobility pattern. On the other hand, data location services do not necessarily depend on nodes' mobility, but rather on nodes' arrivals and departures, as well as on new data advertisements and disposals (due to limited memory consumption the application may dispose some of the data it possess, and in such cases the service has to be updated as well).

A need for additional routing: Position location services usually only provide the position of the requested node, while in order to communicate with this node geographical routing should be employed. On the other hand, data location services may not only provide the location of the data, but actually the data itself (indeed, if only the location of the data is provided by the service, than additional routing should be employed to fetch the content).

Context of the study: Content location services have been primary studied in the context of sensor networks and service/resource discovery, while position location services have been primary studied in the context of geographic routing. Additionally, there is a significant amount of similarity, in terms of problem space and design criteria, between content location services and publish subscribe systems

This paper starts by presenting the taxonomy of location services. The paper than continues by presenting a collection of recent techniques and algorithms for constructing position and content location services, in both wireless ad hoc, hybrid and wired networks according to the aforementioned taxonomy.

2 Position Location Services

2.1 The taxonomy of location services

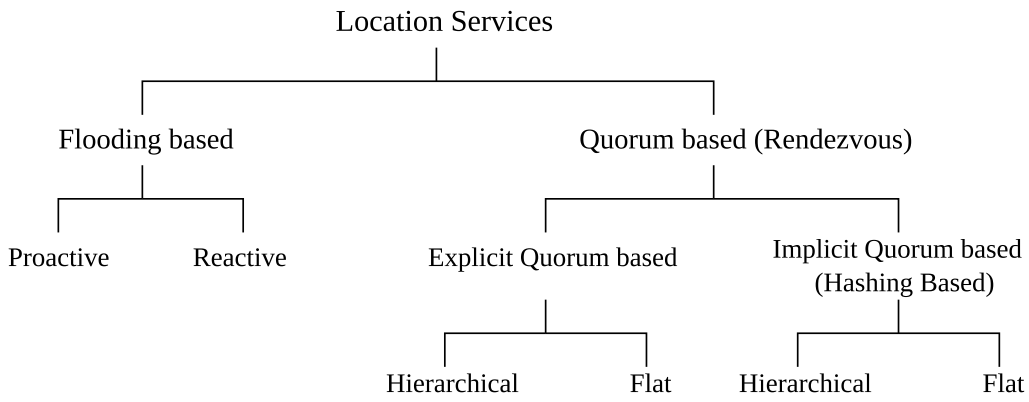


Figure 1: The taxonomy of location services

A taxonomy of position location services has been described in [17] and is depicted in Figure 1. Additional survey on location services is available at [53], while [23, 40] provide surveys on position based routing.

2.2 Flooding-based Location Services

At the top level of the taxonomy, location services can be divided into *flooding-based* and *quorum-based* approaches.[1] Flooding-based protocols can be further divided into *proactive* and *reactive* approaches. In the proactive flooding-based approach, each (destination) node periodically floods its location to other nodes in the network, each of which maintains a location table recording the most recent locations of other nodes. The interval and range of such flooding can be optimized according to the nodes' mobility and the distance effect. For example, flooding should be more frequent for nodes with higher mobility, and flooding to faraway nodes can be less frequent than to nearby nodes.

Camp, Boleng and Wilcox [13] have developed and evaluated the performance of three location services: the Simple Location Service (SLS), the DREAM Location Service (DLS) and the Reactive Location Service (RLS). DLS and SLS are two examples of proactive flooding-based location services. In SLS, nodes periodically transmit tables containing the location information of a few nodes in the system to their neighbors, while in DLS a node transmits its own location information to nearby nodes at a particular rate and to faraway nodes at another lower rate. A node updates its location table if it either receives a location information packet or if it overhears it.

Friedman and Korland suggested an efficient geographic location service, which is based on dividing the network area into a virtual grid as part of the TIGR routing and scheduling framework [21]. The idea is that each node only maintains the direction in the virtual grid towards each other node (North, South, East, West). This way, full flooding is only required once when a node enters the system. After that, only the (at most) two rows and/or (at most) two columns that are affected by a node's mobility need to be notified, and

[1] A subset of the quorum-based protocols was also published under the name *rendezvous-based* protocols.

only when a node changes a virtual grid cell. In particular, minor mobility does not require any notifications.

In reactive (on-demand) flooding-based approaches (e.g. LAR [32], GRID [38] and RLS [13]), if a node cannot find a recent location of a destination to which it is trying to send data packets, it floods a scoped query in the network in search of the destination.

2.3 Quorum-Based Location Services

In quorum-based protocols, all nodes (potential senders or receivers) in the network agree, implicitly or explicitly, upon a mapping that associates each node's unique identifier to one or more other nodes in the network. The mapped-to nodes are the location servers for that node. They will be the nodes where periodical location updates will be stored and location queries will be routed to and looked up at. As long as the lookup reaches one of the servers that was updated about the location corresponding to a given identifier, the correct location will be found. In other words, the quorum of location servers updated on each location change needs to intersect the quorum of location servers consulted with in a lookup.

Often, when the update and lookup quorum sets are the same, quorum-based protocols are also called *rendezvous-based* protocols. This is because the location servers in these quorum sets serve as rendezvous points for updates and lookups.

The mapping of nodes to quorums can be done in either a static or random way, while utilizing nodes identifiers or geographical information and by applying different hashing methods. Both implicit or explicit quorum-based protocols can be further divided into hierarchical or flat, depending on whether a hierarchy of recursively defined subareas are used in order to implement the quorum system.

2.3.1 Explicit Quorum-based Location Services

In the explicit quorum-based approach, each location update of a node is sent to an explicit defined subset (*update quorum*) of available nodes, and a location query for that node is sent to a potentially different subset (*query quorum*). The two subsets are designed such that their intersection is non-empty, and thus the query will be satisfied by some node in the update quorum.

Flat Quorum-based Location Services Flat quorum-based location services base their operation on the notion of quorums. However, in these approaches, all quorum servers are symmetric in their roles. In particular, there is no hierarchical structure in such quorums. Several methods on how to generate flat quorum systems have been discussed in [25, 26, 31, 10].

In the works by Haas and Liang [25, 26], node location information is maintained in location databases that form a virtual backbone. The algorithm for the initiation and management of the virtual backbone is mentioned in [37]. When a node moves, it updates its location with one subset containing the nearest backbone node. Each source node then queries the subset containing its nearest backbone for the location of the destination, and uses that location to route the message. In [26] the choice of quorum system is static and done a-priori, while ensuring uniformity in the sense that all nodes will be members of the same number of quorum groups. On the other hand, in [25] the selection of quorums is done at random during runtime each time a quorum group is needed.

The work reported in [31] focuses on the problem of efficiently utilizing quorum systems in a highly dynamic environment. The nodes are partitioned into fixed quorums, and every operation updates a randomly selected group, thus balancing the load.

In [35], a comparison is made of the strict quorum strategies presented in [31] and an approach using probabilistic quorums. Simulation results show that a better recency rate is obtained if probabilistic quorum based algorithms are used.

In [10], probabilistic quorums are used. The nodes of each quorum are selected randomly, either by picking nodes from the membership list maintained by an underlying routing algorithm or by using random walks on the network to select a random subset of nodes. No means are provided though, to determine the desired size of the random quorum and no theoretical evaluation of the quorum selection algorithms is supplied.

Flat Geographical Quorum-based Location Services In the algorithms discussed below, geographical information is utilized in order to format explicit quorum systems. For example, in the so-called column-row quorum-based protocol [52], the position of each node is periodically propagated in the north-south direction, while any location queries are propagated in the east-west direction. Effectively, each node's location is disseminated to O($\sqrt{N}$) location servers.

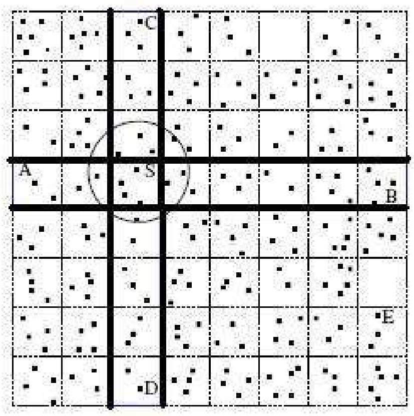


Figure 2: Octopus (figure taken from [41])

In order to achieve a better tradeoff between service accuracy and its load, Octopus [41] enhances the scheme introduced in [52] by applying a new location update technique called *synchronized aggregation*. Octopus divides the network area into horizontal and vertical strips, and stores the location of each node at all the nodes residing in its horizontal and vertical strips. Each location update packet aggregates the locations of several nodes and updates many location servers. Additionally, aggregated updates are synchronized in the sense that only one node initiates the propagation of an aggregate update from a given region, and hence no duplicate updates are sent. Figure 2 depicts node S's neighbors and strips. Nodes A,B,C, and D are the end nodes in the highlighted strips.

In GeoQuorums [19], geometric coordinates determine the location of home servers. These focal point coordinates define geographic areas that must be inhibited by at least one server at any time. Sets of focal points are organized in intersecting quorums. The quorums are further used to implement atomic memory abstraction in mobile ad hoc networks. The algorithm to dynamically reconfigure the set of available quorums is presented as well.

Hierarchical Geographical Quorum-based Location Services Two examples of explicit geographical quorum-based location services, which deploy a hierarchical structure of the quorum nodes, are multi-zone routing [7] and LLS [6].

The position based multi-zone routing method of Amouris et al. [7] stores location information about each node in geometrically increasing discs, each disc referencing the smaller disc that contains the node. When a node moves a distance 2^i, it broadcasts an update about the change to an area of radius 2^{i+1}. Within a 2^i zone, location update is flooded to all nodes. The lookup process tracks the location information in a hierarchical manner; while getting closer to a destination, the information about the position of the destination becomes more precise.

The Locality aware Location Service (LLS) [6] uses hierarchical spiral structure to guarantee *the locality aware* properties of the lookup/publish algorithms. Namely: a location service has a *locality aware lookup* algorithm if the cost of locating destination t from source s is proportional to the cost of the minimal cost path between s and t. A location service has a *locality aware publish* algorithm if the cost of updating the location service due to the move of a node from x to y is proportional to the distance between x and y.

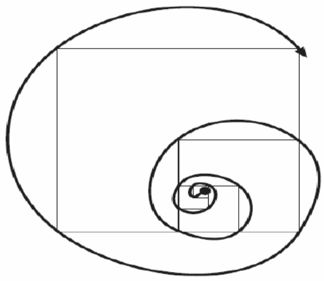


Figure 3: LLS (figure taken from [6])

Figure 3 depicts the basic structure of LLS. For each destination node t, LLS defines virtual hierarchical cover of the M $\times$ M plane consisting of exponentially decreasing squares, whose origin depends on the node's id. In the basic Spiral algorithm, the location of node t is published on a spiral that spans increasingly large squares in the hierarchy. Likewise, searches for t are performed in increasing spirals on the same virtual hierarchy. The lookup and publishing paths are guaranteed to cross at the first hierarchy level in which the squares containing the source and the destination intersect (this structure bears resemblance to the hierarchical grid of GLS [36]. However, GLS uses hashing in addition to the geographic hierarchy).

The mobility is addressed in LLS with techniques borrowed from GHT [44], by using virtual coordinates within the squares for storing information on t rather than search for certain pre-designated nodes. The basic algorithm is further augmented to guarantee the worst case locality aware properties in face of mobility and nodes failures.

LLS achieves worst case lookup cost of O(d^2) and an average case linear cost of O(d), with d being the minimal path length between source and destination. Additionally, when a node moves distance d, the average cost of publishing its new location is O($d \log d$). The inherent locality of LLS makes it fault tolerant both to nodes failures and to network partitions.

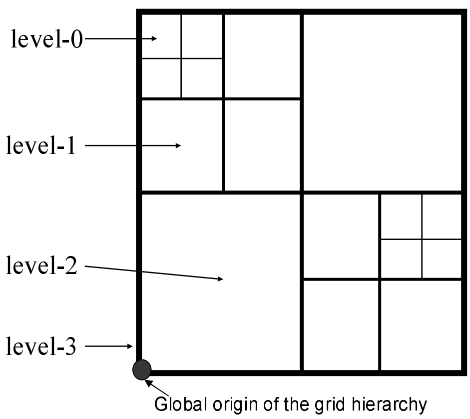
2.3.2 Implicit (Hashing-based) Quorum-based Location Services

In the implicit quorum-based protocols, location servers are chosen via a hashing function, either in the node identifier space or in the location space. Notice that the implicit quorum-based protocols are sometimes called *rendezvous hashing-based* protocols in the papers describing them.

Flat Hashing-based Location Services In the flat hashing-based protocols (for example, [22, 51, 56] and Terminodes [29]), a well-known hash function is used to map each node's identifier to a *home region* consisting of one or more nodes within a fixed location in the network area. All nodes in the home region maintain the location information for the node and can reply to queries for that node; they serve as its location servers. Typically, the number of location servers in the home region is independent of the total number of nodes in the network, and thus effectively each node's location is disseminated to O(1) location servers.

In SLALoM [16], the area is divided into unit regions and each node is assigned multiple uniformly distributed home regions. Nodes in the home regions serve as location servers of the node. Home regions near a node are aware of the node's exact location (that is, the unit region it occupies), while home regions that are far from the node only know a larger region that contains the node.

Hierarchical Hashing-based Location Services In the hierarchical hashing-based protocols, the area in which nodes reside is recursively divided into a hierarchy of smaller and smaller grids (see Figure 4 for GLS's hierarchy). For each node, one or more nodes in each grid at each level of the hierarchy are chosen as its location servers. Location updates and queries traverse up and down the hierarchy. A major benefit of maintaining a hierarchy is that when the source and destination nodes are nearby, the query traversal is limited to the lower levels of the hierarchy.



(a) GLS's spatial hierarchy

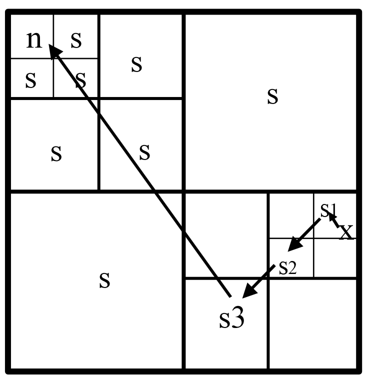

(b) Queries and updates traversals of the hierarchy

Figure 4: GLS (figure taken from [36])

A hierarchical grid structure is very efficient in distributing the load evenly across the network and in achieving efficiency and scalability of the location service. Since the height of the hierarchy is usually O($\log(N)$), effectively each node's location is disseminated to O($\log(N)$) location servers.

The examples of hierarchical hashing-based protocols are Grid Location Service (GLS) [36], Distributed Location Management (DLM) [57], and Geographical Region Summary Service (GRSS) [28]. DLM is similar to GLS except that in DLM location servers for a node are selected by applying a hash function to the node's ID whereas in GLS the location server of a node is the node with the closest ID at the same level of the hierarchy. GRSS uses summary messages and packet forwarding to learn about node locations and in order to further decrease the overhead. In order to reduce the packet size, GRSS uses Bloom filters [11, 12]. To the best of our knowledge, the best known and mostly evaluated example of hierarchical quorum-based location service is GLS [36].

3 Data Location Services

3.1 Geographic Data Location Services

Geographical Quorum-based Data Location Services The Geography-based Content Location Protocol (GCLP) [54], as well as [9] and [42] use similar approaches to column-row quorum-based protocols. In particular, [9] proposes propagating the content and resource advertisements and queries in cross-shaped trajectories, thus guaranteeing two intersections. Queries are answered by nodes at the intersection of the advertising and query trajectories. GCLP (Figure 5) uses similar techniques, augmented with additional improvements: a node may pick whether to forward an advertisement message or not based on the proximity of the content server. In case of multiple advertisements for the same content, only the closest copy is advertised further on.

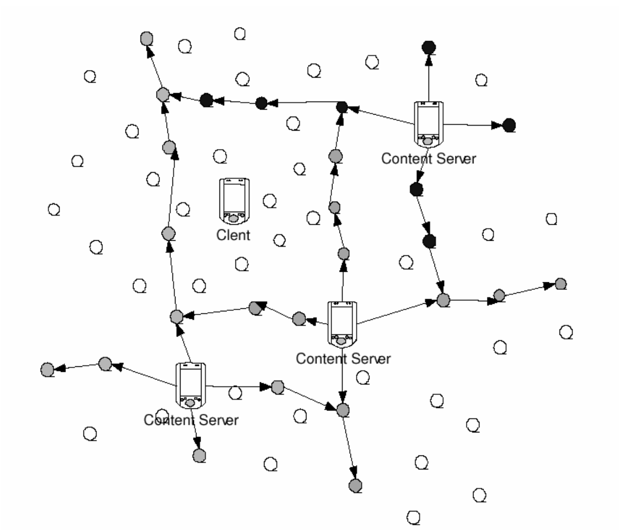


Figure 5: GCLP (figure taken from [54])

Geographical Flat Hashing-based Data Location Services An example of a flat hashing-based data location service is Geographic Hash Table (GHT) [44], that was proposed for data-centric storage in sensor networks. GHT is a geographic hash table system that hashes keys into geographic points, and stores the key-value pair at the sensor node closest to the hash of the key. GHT requires nodes to know their exact geographic location and uses the GPSR [30] routing protocol to identify and to reach a packet's home node (the node closest to the geographic destination). GHT uses GPSR perimeter routing to replicate the keys at nodes in the home perimeter (perimeter that encloses the destination) and to refresh them periodically, in order to detect topology changes after failures or mobility.

Rendezvous Regions (RRs) [49] incorporated a similar approach to GHT, but uses a rendezvous regions instead of rendezvous points. RRs is a general architecture proposed for service location and bootstrapping in ad hoc networks, in addition to data-centric storage, configuration, and task assignment in sensor networks. In RRs, the network topology is divided into geographical regions, where each region is responsible for a set of keys representing the services or data of interest. Each key is mapped to a region based on a hash-table-like mapping scheme. A few elected nodes inside each region are responsible for maintaining the mapped information. The service or data provider stores the information in the corresponding region and the seekers retrieve it from there. Authors claim that RR scales to large number of nodes, is highly robust and efficient to node failures, and is also robust to nodes mobility and location inaccuracy with a significant advantage over point-based rendezvous mechanisms.

3.2 Non Geographic Service Discovery protocols

Centralized Service Discovery Traditional solutions for service discovery, such as Jini [8], UDDI [3], Salutation [5] and Salutation Lite [43], rely on a central directory that stores information about services available in the network in order to enable service discovery and invocations. The discovery of the directory by service seekers and providers is usually based on multicasting. A survey and comparison of central directory based solutions (as well as some flooding based solutions) was presented in [34].

Flooding-based and DNS-based Service Discovery Examples of flooding based decentralized resource discovery protocols, in which service providers and clients discover each others directly by broadcasting service queries or service advertisements, are SLP [24] and UPnP [4].

Universal Plug and Play (UPnP), supported by Microsoft, works primarily at lower layer network protocols suite (i.e. TCP/IP). UPnP uses the Simple Service Discovery Protocol (SSDP) for discovery of services on Internet Protocol based networks. When a service joins a network, it sends out an advertisement message, notifying the world about its presence. The advertisement message contains a Universal Resource Locator (URL) that identifies the advertising service and a URL to a file that provides a description of the advertising service. When a service client wants to discover a service, it can either contact the service directly through the URL that is provided in the service advertisement, or it can send out a multicast query request.

Apple's Bonjour [1], formerly called Rendezvous, extends the DNS-Based Service Discovery [2], and serves as a general method to discover services in a local area network. It is currently used by Mac OS X to find printers and file sharing servers, as well as by additional Mac OS X applications, such as iTunes to find shared music, iPhoto to find shared photos, iChat to find other local iChat users, TiVo Desktop to find digital video recorders, SubEthaEdit to find document collaborators and Safari web browser to find local web servers and configuration pages for local devices.

Konark [27] couples the service discovery protocol with network-layer multicast dedicated to MANET. The scalability and efficiency issues of the discovery protocol are not targeted in this work. Nom [20] uses Gnutella like controlled flooding technique for resource location and discovery in MANET. Examples of additional resource discovery protocols are [39] and [46].

3.3 Service Discovery in Hybrid Networks

In order to increase the scalability and availability of the discovery protocol in hybrid networks, solutions presented in [14, 18] employ a distributed set of directories, which are deployed over base stations (or gateways) and are responsible for a spatial region (e.g., corresponding to a cell). [18] assist geographical knowledge to distribute the advertisement and query workload to multiple directories and for routing.

Scalable Filters-based Service Discovery GSD [15] is a group-based service discovery protocol that performs flooding based local discovery of available services. To achieve global discovery, GSD classifies services into several groups, according to a service hierarchy expressed in DARPA Agent Markup Language (DAML), and includes in each advertisement that is broadcasted the list of groups that the sender has seen in its vicinity. The group information is then used to selectively forward a service request to other nodes. This solution requires the group information to be exchanged and included in each advertisement. The flooding overhead is reduced by the usage of the semantic features present in DAML, efficient service grouping and selective forwarding of the messages.

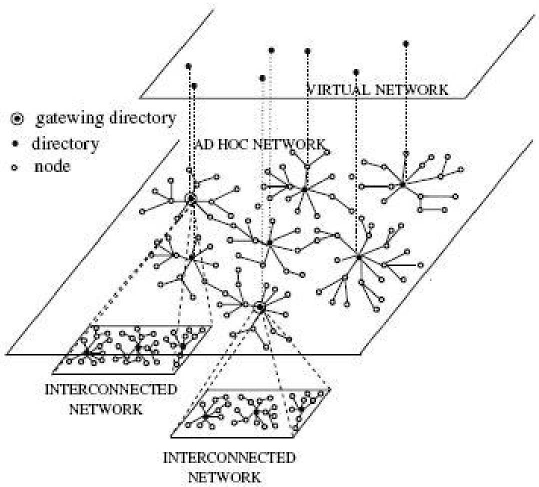


Figure 6: Hierarchical Structure of the Scalable Service Discovery protocol (figure taken from [48])

A Scalable Service Discovery protocol was introduced in [48]. It uses 2 level hierarchical structure overlay based solution (Figure 6). A virtual network backbone is composed of gateway nodes and a subset of MANET nodes acting as directories, responsible for performing service discovery. These directories are deployed so that at least one directory is reachable in at most a fixed number of hops, H, whose value is dependent upon the nodes density. Directories cache the descriptions of services available in their vicinity which is defined by H. A service seeker simply sends a query to the directory for local service discovery. If the description of the requested service is not cached by the local directory, the directory selectively forwards the query to other directories so as to perform global discovery. Selection of directories to which service queries are forwarded is based on the exchange of profiles among directories. The directory profile provides a compact summary of the directory's content and a characterization of the host capacity. The exchange of profiles among the directories is performed proactively, while the size of the profiles is effectively reduced by the usage of bloom filters. Directory profiles allow both guaranteeing that service queries are issued to directories that are likely to cache the description of the requested service and to keep the generated traffic to a minimum. The algorithm is further augmented to enable discovery of services over a hybrid network.

3.4 Bloom filters

Bloom filters [11] is a general technique to reduce memory consumption and processing time of the location service algorithms. Bloom filters are mainly used in algorithms for sensor networks, due to limited resource capacity on the devices in such networks. In addition, bloom filters are used in service/resource discovery algorithms (e.g., [48]) and event publish-subscribe systems, since in those systems data representation (such as service or event description) is very rich and extensive and therefore summarizing techniques, such as bloom filters, could serve as a powerful tool to reduce service location's overhead. A survey on usage of bloom filters for network applications is presented in [12].

4 Location Services in Wired P2P Networks

Location services in wired networks are primarily based on rather centralized solutions (such as Jini [8] and UDDI [3]) or on DNS based solutions (DNS-Based Service Discovery [2]).

In recent years, a substantial amount of research has been devoted to object and content location in large scale P2P (peer-to-peer) networks. In particular, recent works on Decentralized Object Location and Routing (DOLR) systems (CAN [45] , Pastry [47], Chord [50], and Tapestry [58]) combine location and routing in an overlay network layer, promising several important desirable properties such as decentralized operation, scalability, fault-tolerance, and load balancing.

DOLRs usually use a DHT (distributed hash table) abstraction to map object names to network nodes and maintain an overlay based routing scheme in order to route messages to the mapped keys or unique identifiers associated with those objects names. DOLRs use novel search algorithms based on coordinate or high-dimensionality routing to yield routing table storage and a number of hops between nodes that are logarithmic to the size of the total network, while guaranteeing consistent operation.

This basic DHT and routing scheme is often augmented with additional techniques, such as replication and caching to further increase scalability and data availability. An important factor in measuring and comparing the efficiency of such DHT based solutions is its "locality awareness". *Locality awareness* is informally defined as the ability to exploit local resources over remote ones whenever possible. "Local" and "remote" are defined here in the context of heterogeneous network links with varying latency, bandwidth, and distance between the source (requesting) and destination nodes. Intuitively, locality awareness is also the property that enables a system to limit the impact of local operations on wide-area performance, both during regular operation and under fault conditions.

An attempt to provide strong consistency semantics and persistency over P2P networks was undertaken in the OceanStore [33] project. An alternative generic approach to managing routing tables for P2P routing overlay networks, based on probabilistic gossiping, is presented in [55].

Note that despite an attempt to provide "locality awareness" in P2P networks, these service discovery solutions are not suitable for mobile wireless ad hoc networks. First, they are designed for static IP networks, which provide full network connectivity between any two nodes, and exploit an existing routing infrastructure of these networks. In addition they do not consider mobile networks limitations, such as limited memory capabilities, their dynamic nature and the lack of routing infrastructure.

Acknowledgements: We would like to thank Gwendal Simon for motivating us to work on this paper and for interesting discussions.

References

[1] Apple's Bonjour. Available at `http://developer.apple.com/networking/bonjour/`.

[2] DNS-Based Service Discovery. IETF draft, Available at `http://files.dns-sd.org/draft-cheshire-dnsext-dns-sd.txt`.

[3] UDDI. Available at `http://www.uddi.org`.

[4] UPnP White Paper. Available at `http://upnp.org/resources.htm`.

[5] Salutation Architecture Specification. The Salutation Consortium, Available at `http://www.salutation.org`, June 1999.

[6] I. Abraham, D. Dolev, and D. Malkhi. LLS: a Locality Aware Location Service for Mobile Ad Hoc Networks. In *Proc. of the Joint Workshop on Foundations of Mobile Computing (DIALM-POMC)*, pages 75–84. ACM Press, 2004.

[7] K.N. Amouris, S. Papavassiliou, and M. Li. A Position Based Multi-Zone Routing Protocol for Wide Area Mobile Ad-Hoc Networks. In *Proc. 49th IEEE Vehicular Technology Conference*, pages 1365–1369, 1999.

[8] K. Arnold, A. Wollrath, B. O'Sullivan, R. Schei, and J. Waldo. The Jini specication. Addison-Wesley, Reading, MA, USA, 1999.

[9] I. Aydin and C.-C. Shen. Facilitating Match-Making Service in Ad hoc and Sensor Networks Using Pseudo Quorum. In *Proc. of the 11th IEEE International Conference on Computer Communications and Networks (ICCCN)*, Miami, Florida, 2002.

[10] S. Bhattacharya. Randomized location service in mobile ad hoc networks. In *Proc. of the 6th ACM International Workshop on Modeling Analysis and Simulation of Wireless and Mobile Systems (MSWIM)*, pages 66–73. ACM Press, 2003.

[11] B.H. Bloom. Space/Time Tradeoffs in Hash Coding with Allowable Errors. *Communications of the ACM*, 13(7):422–426, July 1970.

[12] A. Broder and M. Mitzenmacher. Network Applications of Bloom Filters: A survey. In *Annual Allerton Conference on Communication, Control, and Computing, Urbana-Champaign, Illinois, USA*, October 2002.

[13] T. Camp, J. Boleng, and L. Wilcox. Location information services in mobile ad hoc networks. In *Proc. of the IEEE International Conference on Communications*, pages 3318–3324, 2002.

[14] P. Castro, B. Greenstein, R. Muntz, P. Kermani, C. Bisdikian, and M. Papadopouli. Locating application data across service discovery domains. In *Proc. of the 7th annual international conference on Mobile computing and networking (MobiCom)*, pages 28–42, New York, NY, USA, 2001. ACM Press.

[15] D. Chakraborty, A. Joshi, T. Finin, and Y. Yesha. GSD: A novel groupbased service discovery protocol for MANETS. In *Proc. of the 4th IEEE Conference on Mobile and Wireless Communications Networks (MWCN)*, 2002.

[16] C. Cheng, H. Lemberg, E. B. S.J. Philip, and T. Zhang. SLALoM: A scalable location management scheme for large mobile ad-hoc networks. In *Proc. IEEE Wireless Communication and Networking Conference*, pages 574–578, 2002.

[17] S. M. Das, H. Pucha, and Y. C. Hu. Performance Comparison of Scalable Location Services for Geographic Ad Hoc Routing. In *Proc. of IEEE INFOCOM, Miami, FL*, 2005.

[18] M. Denny, M. J. Franklin, P. Castro, and A. Purakayastha. Mobiscope: A Scalable Spatial Discovery Service for Mobile Network Resources. In *Proc. of the 4th International Conference on Mobile Data Management (MDM)*, pages 307–324, London, UK, 2003. Springer-Verlag.

[19] S. Dolev, S. Gilbert, N. Lynch, A. Shvartsman, and J. Welch. Geoquorums: Implementing atomic memory in mobile ad hoc networks. In *Proc. of the 17th International Symposium on Distributed Computing (DISC)*, 2003.

[20] D. Doval and D. O'Mahony. Nom: Resource Location and Discovery for Ad Hoc Mobile Networks. In *Proc. of the 1st Annual Mediterranean Ad Hoc Networking Workshop (Medhoc-Net)*, 2002.

[21] R. Friedman and G. Korland. Timed Grid Routing (TIGR) Bites off Energy. In *Proc. 6th ACM International Symposium on Mobile and Ad Hoc Networking and Computing (MobiHoc)*, pages 438–448, May 2005.

[22] S. Giordano and M. Hamdi. Mobility management: The virtual home region. EPFL, Lausanne, Switzerland, Tech. Rep. SSC/1999/037, Oct. 1999.

[23] S. Giordano, I. Stojmenovic, and L. Blazevie. Position based routing algorithms for ad hoc networks: a taxonomy. Available at `http://www.site.uottawa.ca/-ivan/routing-survey.pdf`, July 2001.

[24] E. Guttman. Service Location Protocol: Automatic Discovery of IP Network Services. *IEEE Internet Computing*, 3(4):71–80, 1999.

[25] Z. Haas and B. Liang. Ad Hoc mobility management with randomized database groups. In *Proc. of IEEE ICC*, June 1999.

[26] Z. Haas and Ben Liang. Ad Hoc mobility management with uniform quorum systems. *IEEE/ACM Transactions on Networking*, 7(2):228–240, 1999.

[27] S. Helal, N. Desai, V. Verma, and C. Lee. Konark - A Service Discovery and Delivery Protocol for Ad-hoc Networks. In *Proc. of the 3rd IEEE Conference on Wireless Communication Networks (WCNC)*, March 2003.

[28] Pai-Hsiang Hsiao. Geographical region summary service for geographical routing. In *Proc. of the 2nd ACM international symposium on Mobile ad hoc networking and computing (MobiHoc)*, pages 263–266, 2001.

[29] J.-P. Hubaux, T. Gross, J.-Y. Le Boudec, and M. Vetterli. Towards Self-Organizing Mobile Ad Hoc Networks: the Terminodes Project. *IEEE Communications Magazine*, pages 118–124, 2001.

[30] B. Karp and H. T. Kung. GPSR: greedy perimeter stateless routing for wireless networks. In *Proc. of the 6th annual international conference on Mobile computing and networking (MobiCom)*, pages 243–254. ACM Press, 2000.

[31] G. Karumanchi, S. Muralidharan, and R. Prakash. Information Dissemination in Partitionable Mobile Ad Hoc Networks. In *Symposium on Reliable Distributed Systems*, pages 4–13, 1999.

[32] Y.-B. Ko and N. H. Vaidya. Location-aided routing (LAR) in mobile ad hoc networks. In *Proc. of the 4th annual ACM/IEEE international conference on Mobile computing and networking (MobiCom)*, pages 66–75, 1998.

[33] John Kubiatowicz, David Bindel, Yan Chen, Patrick Eaton, Dennis Geels, Ramakrishna Gummadi, Sean Rhea, Hakim Weatherspoon, Westly Weimer, Christopher Wells, and Ben Zhao. OceanStore: An Architecture for Global-scale Persistent Storage. In *Proc. of ACM ASPLOS*, November 2000.

[34] C. Lee and S. Helal. Protocols For Service Discovery In Dynamic and Mobile Networks. *International Journal of Computer Research*, 11(1):1–12, 2002.

[35] H. Lee. *A randomized memory model and its applications in distributed computing*. PhD thesis, 2001. Chair-Jennifer L. Welch.

[36] J. Li, J. Jannotti, D. De Couto, D. Karger, and R. Morris. A scalable location service for geographic ad-hoc routing. In *Proc. of the 6th ACM International Conference on Mobile Computing and Networking (MobiCom)*, pages 120–130, August 2000.

[37] Ben Liang and Z. Haas. Virtual Backbone Generation and Maintenance in Ad Hoc Network Mobility Management. In *INFOCOM*, pages 1293–1302, 2000.

[38] W.-H. Liao, Y.-C. Tseng, and J.-P. Sheu. GRID: a fully location-aware routing protocol for mobile ad hoc networks. In *Telecommunication Systems, 18, 61-84*, 2001.

[39] J. C. Liu, K. Sohraby, Q. Zhang, B. Li, and W. Zhu. Resource discovery in mobile ad hoc networks. *The handbook of ad hoc wireless networks*, pages 431–441, 2003.

[40] M. Mauve, J. Widmer, and H. Hartenstein. A survey on position-based routing in mobile ad hoc networks. IEEE Network Magazine, 15(6):30–39, November 2001.

[41] R. Melamed, I. Keidar, and Y. Barel. Octopus: A Fault-Tolerant and Efficient Ad-hoc Routing Protocol. In *Proc. of the 24th IEEE Symposium on Reliable Distributed Systems (SRDS)*, pages 39–49.

[42] B. Nath and D. Niculescu. Routing on a curve. *SIGCOMM Comput. Commun. Rev.*, 33(1):155–160, 2003.

[43] B. Pascoe. Salutation-Lite: Find-and-Bind Technologies for Mobile Devices. The Salutation Consortium, Available at `http://www.salutation.org/whitepaper/Sal-Lite.PDF`, June 1999.

[44] S. Ratnasamy, B. Karp, L. Yin, F. Yu, D. Estrin, R. Govindan, and S. Shenker. GHT: A Geographic Hash Table for Data-Centric Storage in SensorNets. In *Proc. of the*

First ACM International Workshop on Wireless Sensor Networks and Applications (WSNA), Atlanta, Georgia, September 2002.

[45] Sylvia Ratnasamy, Paul Francis, Mark Handley, Richard M. Karp, and Scott Shenker. A Scalable Content-Addressable Network. In *SIGCOMM*, pages 161–172, 2001.

[46] O. Ratsimor, D. Chakraborty, S. Tolia, D. Kushraj, A. Kunjithapatham, G. Gupta, A. Joshi, and T. Finin. Allia: Alliance-based service discovery for ad-hoc environments. ACM Mobile Commerce Workshop, 2002.

[47] Antony Rowstron and Peter Druschel. Pastry: Scalable, Decentralized Object Location, and Routing for Large-Scale Peer-to-Peer Systems. *Lecture Notes in Computer Science*, 2218:329+, 2001.

[48] F. Sailhan and V. Issarny. Scalable Service Discovery for MANET. In *Proc. of the 3rd IEEE International Conference on Pervasive Computing and Communications (PERCOM)*, pages 235–244, Washington, DC, USA, 2005.

[49] K. Seada and A. Helmy. Rendezvous Regions: A Scalable Architecture for Service Provisioning in Large-Scale Mobile Ad Hoc Networks. Proc of ACM SIGCOMM, Refereed poster, 2003.

[50] I. Stoica, R. Morris, D. Karger, M. F. Kaashoek, and H. Balakrishnan. Chord: A Scalable Peertopeer Lookup Service for Internet Applications. In *Proc. of the ACM SIGCOMM*, pages 149–160, 2001.

[51] I. Stojmenovic. Home agent based location update and destination search schemes in ad hoc wireless networks. Computer Science, SITE, University of Ottawa, TR-99-10, September 1999.

[52] I. Stojmenovic. A routing strategy and quorum based location update scheme for ad hoc wireless networks. Computer Science, SITE, University of Ottawa, TR-99-09, September 1999.

[53] I. Stojmenovic. Location updates for efficient routing in ad hoc networks. *Handbook of wireless networks and mobile computing*, pages 451–471, 2002.

[54] J. Tchakarov and N.H. Vaidya. Efficient Content Location in Wireless Ad Hoc Networks. In *Proc. IEEE International Conference on Mobile Data Management (MDM)*, January 2004.

[55] Spyros Voulgaris and Maarten van Steen. An epidemic protocol for managing routing tables in very large peer-to-peer networks. In *Proc. of the 14th IFIP/IEEE International Workshop on Distributed Systems: Operations and Management, (DSOM)*, pages 41–54. Springer, 2003.

[56] S.-C. M. Woo and S. Singh. Scalable routing protocol for ad hoc networks. *Wireless Networking*, 7(5):513–529, 2001.

[57] Y. Xue, B. Li, and K. Nahrstedt. A Scalable Location Management Scheme in Mobile Ad-Hoc Networks. In *Proc. of the 26th Annual IEEE Conference on Local Computer Networks (LCN)*, page 102, Washington, DC, USA, 2001.

[58] B. Y. Zhao, J. D. Kubiatowicz, and A. D. Joseph. Tapestry: An Infrastructure for Fault-tolerant Wide-area Location and Routing. Technical Report UCB/CSD-01-1141, UC Berkeley, April 2001.