Survey and new Approach in Service Discovery and Advertisement for Mobile Ad hoc Networks

Seyed Amin Hosseini Seno, Rahmat Budiarto, Tat-Chee Wan

Network Research Group, School of Computer Sciences, Universiti Sains Malaysia

Summary
Service advertisement and discovery is an important component for mobile ad hoc communications and collaboration in ubiquitous computing environments. The ability to discover services offered in a mobile ad hoc network is the major prerequisite for effective usability of these networks. This paper aims to classify and compare existing Service Discovery (SD) protocols for MANETs by grouping them based on their SD strategies and service information accumulation strategies, and to propose an efficient approach for addressing the inherent issues.

Key words:
MANETs, Service Discovery, Mobile Ad hoc Networks, Service Advertisement.

1. Introduction

A mobile ad-hoc network (MANET) is a self-configuring network of mobile nodes [2]. All of the nodes are routers connected via wireless links. The routers are free to move randomly and organize themselves arbitrarily, thus, the network topology may change rapidly and unpredictably. Such a network may operate in a standalone manner, or may be connected to the larger Internet or may be used as a hybrid wireless network. Minimal configuration and quick deployment make ad hoc networks suitable for emergency situations like natural or human-induced disasters, military conflicts, emergency medical situations, etc [2].
There are many attributes in ad hoc wireless, they are summarized as Table 1.

Table 1. Some of attributes of MANETs

Some of Attributes of MANETs	
Infrastructure less	multi hop wireless
shared radio channel (more suitable for best effort data traffic)	dynamic frequency reuse based on carrier sense mechanism
distributed routing	quick and cost effective deployment
time synchronization is difficult and consumes bandwidth	main aim of routing is to find paths with minimum overhead and also quick reconfiguration of broken paths
self organization	frequent path breaks due to mobility
mobile hosts require more intelligence	

There are many issues and challenges that need to be considered in designing an ad hoc wireless system. These issues affect the design, deployment, and performance of ad hoc wireless system. Some of these issues that affect the design, deployment and performance of ad hoc wireless system are shown in Table 2. One of the important issues is ***service discovery*** [2].

Table 2. Some of issues in MANETs

Some of issues in the MANETs	
Routing	Medium Access Scheme
Multicasting	Transport Layer Protocol
Pricing Scheme	Deployment Consideration
Security	Self Organization
Scalability	Energy Management
Addressing	Quality of Service
Service Discovery	

Service Discovery (SD) Definition: A service in the network can be any software or hardware entity that a user might be interested to utilize. A definition of service discovery is: "Service discovery protocols are network protocols which allow automatic detection of devices and services offered by these devices on a computer network.". When the location of the requested service (typically the address of the service provider) is determined, the user may then access and use it [22].

Significant of Service Discovery: Existing approaches have been to share available resources among interested parties. In this way resources could be shared based on various criteria and hence solve the resource shortage problem. This concept is more advanced than most peer to peer resource sharing models where only one type of resource such as files are being shared. These collaborative resource sharing environments are called pervasive environments [23]. To find a suitable service or resource in MANETs is a challenging problem, because of the absence of any central intelligence in the network. Consequently service discovery is an important component for ad hoc communications and collaboration in an ubiquitous computing environment.
Manner of the Service Discovery in the network: A service is only advertised by its host machine after it has joined a network. If the service owner is willing to share the service in the network among different users, it would announce the service availability through a *service advertisement* mechanism. Various algorithms have been proposed for the service advertisement functionality which show dissimilar performance; comparisons were based on different parameters such as the number of local and global messages passed in the advertisement process [23]. Choosing an unsuitable service advertisement procedure

Manuscript received February 5, 2007
Manuscript revised February 25, 2007

may cause severe network traffic congestion inside the computing environment. Selecting appropriate attributes of a service for advertisement is also vital.

On the other hand, a task being submitted to the environment for execution must identify its service needs. In this step the service management system should search for suitable services conforming to the constraints introduced by the task and allocate the matching services immediately to the task, or reserve the service if it is currently occupied using *service discovery*. Various service discovery models based on information store strategies have been implemented [23]. If all service information in an environment is gathered into a single service information store, service discovery can simply be done through querying this database; however the information and status of the available services is usually scattered throughout the environment in different repositories in a decentralized manner. Although a decentralized service information store complicates the functionality of the service discovery algorithm, it serves to avoid network bottlenecks.

In recent years, many surveys in service discovery on MANETs have been done [30].

2. Network Layer SD Strategies

In these protocols, service discovery is coupling with routing layer. Service query and response messages are often piggybacked on to ad hoc routing protocols. In this way, a node requesting a service in addition to discovering the service also be informed of the route to the service provider at the same time. Examples are: AODV, ODMRP, M-ZRP, LSD and DSD [3].

In this paper we classify and compare existing service discovery protocols for MANETs based on SD strategies and service information accumulation strategies and survey the features of the discovery architecture. The SD taxonomy defines mechanism based on:

- Service Discovery Strategies
  - Service Discovery based on Supporting Layer
    - Network Layer
    - Application Layer
  - Service Discovery based on Multicast DNS
- Service Information Accumulation Strategies
  - Without Directory
  - Central Directory
  - Distributed Directory

Service Discovery Strategies

Depending on which layer network that protocols work in, the service discovery architectures can be divided into two categories: network layer based and service caching strategy based.

These protocols were designed for Ad hoc network to store general attributes of services. Among the protocols, only ODMRP was designed to be scalable to different applications.

AODV

Ad hoc On-demand Distance Vector (AODV) is a protocol based on Service Location protocol (SLP) which has been implemented in NS. In fact AODV has been obtained from integration of the SLP with the reactive ad hoc routing protocol. After integration, the control overhead and latency required for discovering services has been improved. Since QoS is essential in real-time high-quality applications, has been used QoS-aware service discovery protocol in AODV. It is a suitable protocol in layer 3 for service discovery. The process of service discovery in ad hoc networks with AODV is very similar to ODMRP. The process will be explained in the next section. When a device or application needs a service, it should prepare a service request including property of the desired service and then include the query as an extension route request message (piggybacked) and send it out. To send reply message, the nodes acts the same as use in the route reply [4].

ODMRP

On Demand Multicast Routing Protocol (ODMRP) is an important multicast routing protocol. According to the performance of comparison study of ad hoc wireless multicast protocols in [15] ODMRP which is very effective and efficient in most simulation scenarios have been selected as a good protocol through five multicast protocols. The protocol are as follows: ad hoc multicast routing (AMRoute) [16],on-demand multicast routing protocol (ODMRP) [17], ad hoc multicast routing protocol utilizing increasing ID numbers (AMRIS) [18], core-assisted mesh protocol (CAMP) [19], and flooding. In regards on [1] and of the basis of the above result, ODMRP is used for service discovery in the mobile ad hoc network. For this purpose, format of ODMRP packet with format of service advertisement and query and reply have been joined. In [1] On Demand Multicast Routing Protocol Service Discovery (ODMRP_SD) is a lightweight protocol which avoids excessive traffic overhead and which is scalable to different appliances.

For service discovery with ODMRP in Push model, every device that has a service multicast an advertisement piggybacked in ODMRP Join Query packet. Interested clients cache it, and send a join reply (ODMRP Join Reply packet with no changes.) If clients reply more than one clients respond, the server will send updated message in ODMRP Join Query packets. Otherwise, it reverts to the pull model. And In Pull model A client sends a query piggybacked in ODMRP Join Query packet to the service query multicast group. A server sends back a reply message after a timeout to suppress repeated replies [5].

MZRP

MZRP is service discovery protocol for ad hoc networks which is based on Zone Routing protocol (ZRP). It is either a proactive or a reactive protocol. It has some sub protocols do some functions:

- Intrazone Routing Protocol (IARP) to take the advantage of proactive discovery within a node's local neighborhood.
- Interzone Routing Protocol (IERP) to use a reactive protocol for communication between these neighborhoods
- Neighbor Discovery Protocol (NDP), to create a zone via broadcasting.
- Broadcast Resolution Protocol (BRP) to the forwarding of a route request.

ZRP divides a network to different zones based on neighborhoods node and number of hops that define it.

In order to add service discovery capabilities to ZRP, in [6] embedded an extra field in NDP "hello" message for store IDs and used the concept of Unique Universal IDentifiers (UUIDs) instead of service descriptions in order to keep small packet lengths for routing messages, in this case all of nodes know mapping between services offered in mobile ad hoc and UUIDs. In this environment, the roles of every participating node are couple and can be easily classified in types of services. Thus, by extending "hello" messages with service UUIDs, a node is able to denote both its presence and the services it provides. ZRP was further modified in order to include service information in every routing entry of the IARP routing messages and tables. IARP listens to information gathered from NDP messages, updates its table and then periodically broadcasts its table to its neighbors. This way each node knows the routes to all the nodes in its zone and also the services that these nodes offer; thus adding the service discovery capability to the proactive part of ZRP. This modified version of ZRP, (M-ZRP) is capable of providing routing layer support for proactive service discovery.[6] this approach combines route discovery with service discovery, thus allowing nodes to find available services and routes to them simultaneously. With this way fewer messages are broadcasted into the network; hence each node saves great amounts of energy.

LSD

Lightweight Service Discovery (LSD) is a protocol for service discovery on mobile ad hoc networks that support routing protocol. In [20] F Zhu et al, have classified the service discovery models into two categories: client-*service model* and *client-service-directory model*. In the client-service model, if a client needs to any services, it broadcasts or multicasts its queries on the network. The services that match with received query, returns a reply. In this model there is not any directory server to keep the services information. This model is suitable for small networks. In the client-service-directory model, there is a directory that keeps all of services information which are on the network. A client sends a query to directory server to find service and then contacts to service. LSD protocol supports both of above mention categories to service discovery in mobile ad hoc. It is an automatic protocol, If a directory server available on the network, clients send them query to directory otherwise, they broadcast or multicast mechanism.

Moreover, for service of query and response, LSD uses piggybacked message on-to proactive ad hoc routing protocol mechanism to service discovery in mobile ad hoc networks. In LSD a directory server either advertises periodically or changes configuration information itself. If a service of node does not hear any directory server, at a definite time, it uses the broadcast mechanism to discover directory for registration, however, if no directory is known, a node replies directly to a query provided that it has a matching service. LSD offers scalability to handle the size changes of the network for large sized MANET[7].

DSDP

Distributed Service Directory Protocol (DSDP) is another protocol for mobile ad hoc networks which support routing protocol. This protocol has used distributed directory architected for keep the services information and virtual backbone for locating and registering available services within a dynamic network topology. In this virtual backbone each node in the network is either a part of the backbone or one hop away from at least one of the backbone nodes. The DSDP has 2 important algorithms:

- Backbone Management (BBM) Algorithm, that can be described in three components: (i) initial selection of backbone nodes, (ii) mesh formation by finding the paths between backbone nodes, (iii) and maintenance against topology changes. All the components rely on the periodically broadcasted hello beacons and Each node creates a neighborhood information table (NIT) and a routing table using the information carried by these beacons The nodes in the virtual backbone act as service brokers and form a mesh structure that is interconnected by virtual links.
- Distributed Service Discovery (DSD) Algorithm, this algorithm manages service advertisement and service discovery on created virtual backbone.

DSDP uses ODMRP for routing, in fact in DSDP, Application Layer has cooperated with network layer for service discovery [21][9].

Table 3 provides an overview of network layer protocols for service discovery in MANETs. These protocols were designed for Ad hoc network to store general attributes of services. Among the protocols, only ODMRP was designed to be scalable to different applications.

Service Discovery based on Application layer

In these protocols, service discovery functionality is supported above the routing layer and usually services belong to the application level. Many papers have been written on the development of variety of protocols such as DEAPspace, Konark, GSD, SSD and SANDMAN.

Table 3. Comparison of network layer SD protocols

Protocol	Network Architecture	Storage Service Information	Search Method
AODV [4]	Central and Distributed Directory	Central and Distributed Directory	route request/reply
ODMRP [1],[5],[15]	Central and Distributed Directory	Cache of nodes	Push and Pull
MZRP [6]	Use cluster	Central Directory + every node in cluster	Push and Pull
DSDP [15]	Use cluster	Distributed Directory	multicast trees rooted
LSDP [14]	Central and Distributed Directory	Central and Distributed Directory	Push and Pull

Konark

Konark is a service discovery and delivery protocol designed specifically for mobile ad hoc networks. This protocol enables each device to act as a service server and a client simultaneously. There are three main components: (i) The SDP manager (ii) Registry (iii) Micro HTTP. The SDP manager and the registry together maintain service objects and information of services and the micro-HTTP manage the service delivery requests. For service discovery and advertisement, The SDP manager installs on every node in network. Every node stores the information of its services or other services that discovered or received via advertisements. Information of services in this protocol defines XML-base based on WSDL and stores in a tree based structure.

Konark uses multicasts to service advertise and service discovery. If any node needs to a service, it sends a discovery message on a fixed multicast address. Each node receiving a discovery message checks the requested service information. If the matching is successful, the node creates an advertisement message for each match found.

When a node advertises its service, other nodes pick up the path and match it with their registry tree. If the matching is successful the service is added under the specified path, otherwise, the node adds the service information to its registry tree [8][9][23].

GSD

GSD is a Group-based distributed Service Discovery protocol for mobile ad hoc networks. It is based on the concept of peer-to-peer caching of service advertisements and group-based intelligent forwarding of service requests to reduce the broadcast storm problem. It does not require a service to register to a lookup server. This protocol uses DAML+OIL to define ontology to describe services in a MANET.

DAML supports XML and semantic definition and semantic matching. The services present on the nodes are classified into several groups based on the class-subclass hierarchy present in DAML.

Each node in GSD has one or more services, these services describe the DAML. This node advertises its services to all nodes in its radio range with a certain hop (number of nodes >= hops >= 1) after a certain time (this time can change in the algorithm). Every node has infant cache to store services information, when the node receives an advertisement, if the service information do not duplicate, store it with a lifetime. Each packet of advertisement has *Other-Groups* field, containing a list of non local services. This field is empty for a local service. If a node on the MANETs or an application needs to a service, it prepared a request and the first time the request (A Service Request is described using DAML-based ontology) description is matched with the services present in the local cache, (By virtue of peer-to-peer caching, the local cache (in an ideal scenario) contains the descriptions of services present in the nodes within ADV DIAMETER hops), if it does not found , it broadcasts its request to network with emphasis to the vicinity (in this step can define hops count) [24][9].

SANDMAN

Service Advertisement and Discovery for Mobile Ad hoc Networks (SANDMAN) is service discovery algorithm based on node clustering. Nodes within a cluster may sleep to save energy when idle. A cluster head node is always active and answers discovery requests on behalf of other nodes to achieve low discovery latencies. This protocol has 3 main objectives: (*i*) energy efficiency by letting nodes sleep, (*ii*) minimize discovery latency and (*iii*) decentralize operation.

In this algorithm, if Cluster Nodes (CNs) are idle, they go to sleep periodically, to save energy. After waking up, a CN waits for incoming client requests for the duration of a timeout interval. If no requests are received, it informs the Cluster Head (CH) and returns to its sleep mode. In this way, it will save a substantial amount of energy. The CH stays awake all the time. It acts as a local Lookup Service therefore CH can instantly answer any client's discovery request with a suitable list of service descriptions and wake up times of the respective CNs.

Briefly SANDMAN is a protocol for energy efficient service discovery in privacy computing environments. SANDMAN uses a node clustering approach to identify a number of nodes the cluster heads that stay awake permanently and answer discovery requests on behalf of the nodes in their clusters. This allows the nodes in the cluster to maximize their sleep times. The discovery delays are unaffected, as the CH can answer every discovery request instantly [10].

Table 4 gives an overview of Application layer protocols for service discovery in MANETs.

Table 4. Comparison of application layer SD protocols

Protocol	Network Architec-ture	Storage of Services Information	Search Method	Service Descrip-tion
Konark [8,9,16]	Point 2 point	All nodes	Push& Pull	XML
GSD [9,17]	Point 2 point	All nodes	Push& Pull	DAML +OIL
SANDMAN [10]	Clustered	Cluster head	Pull	Wakeup time +general

Service Discovery based on Multicast DNS

Multicast DNS has been devised for the resolution of domain names to IP addresses in the link-local scoped network. multicast DNS comprises a Responder and a Sender. The Sender is the resolver that sends a query in link-local multicast and the Responder is the name server that sends the response to the Sender using unicast. When the Sender receives the response, it verifies if the response is valid the Sender stores it in its cache and passes the response to the application that initiated the DNS query. Otherwise, the Sender ignores the response and continues to wait for other responses [12]. It is possible to extend the response with the name of services by changing the DNS entries. using this method, we can implement service discovery through multicast DNS.

In [12], a proposed architecture for name service called ANS (Ad-hoc Name Service for IPv6 MANET) implement SD in IPv6 MANET. ANS allows mobile nodes to perform the service discovery as well as the name-to-address resolution. ANS supports the discovery of unicast and multicast service. In addition, ANS provides auto configuration technology for zero configurations related to name service, such as generation of unique domain names for mobile node and zone files for the name service. In this manner ANS will be a suitable DNS service and service discovery for MANETs and all networks where there are no network managers especially home and small office networks.

Comparison of the various SD strategies

Protocols that implement service discovery at the routing layer instead of the application layer significantly reduce the communication and energy consumption overheads. By implementing service discovery in the routing layer through piggybacking the service information into the routing protocol control messages, devices are able to acquire both service and routing information simultaneously. This approach decreases communication overheads and saves battery power.

3. Service Information Accumulation Strategies

A directory is an entity that stores information about services available in the network so as to enable service discovery and invocation.

Services Information Accumulation (SIA) methods can be classified as:

- *SD without using directory(directory-less)*
- *SD using centralized directory*
- *SD using distributed directory*

SD without using directory

In the directory-less architecture, nodes do not distribute their service descriptions onto other nodes in the network. A device interested in a special service typically sends its search message to all reachable nodes. If one or more of these nodes can satisfy the request, a response is sent back to the requestor.

UPnP

The Universal Plug and Play (UPnP) is a simple extension of the Plug and Play peripheral model. It is designed to support zero configuration, "invisible" networking, and automatic discovery for a breadth of device categories from a wide range of vendors. With UPnP, a device can dynamically join a network, obtain an IP address, convey its capabilities, learn about the presence and capabilities of other devices, truly enabling zero configuration networks.

The UPnP network consists of three basic building blocks: *devices*, *services* and *control points*. An UPnP device is a container of services and nested devices. A service is the smallest unit of control. A control point in an UPnP network is a controller capable of discovering and controlling other devices. When a device added to network, it advertises its services to the control points and control point get the service information via discovery message.

control point to look for devices and services, the Simple Service Discovery Protocol (SSDP) was created as a lightweight discovery protocol for UPnP initiative, and defines a minimal protocol for multicast-based discovery. The SSDP allows a device to announce its availability. A UPnP device in turn listens to the multicast port. UPnP is based on wire protocols[11] [8].

DEAPspace

DEAPspace provides a framework to connect devices over a wireless medium. It is a push-model-based approach to fast and resource efficient service discovery. All of services attributes are storeed on service providers. DEAPspace services are specified as a data hierarchy. The root node of this hierarchy is the DSService class. Each service description has a field to keep expire time (time-to-live). Nodes advertise their services by a broadcast mechanism to their neighbors.

In fact, a device which provides a service that can be utilized by other requesting devices, acts as a provider. Conversely, the device requesting the provision of a service is called a requester. To provide its service, a provider in turn can act as a requester making use of other services. This protocol does not have any central directory [26].

PDP

Pervasive Discovery Protocol (PDP) is a fully distributed protocol that merges characteristics of both pull and push solutions for ad hoc networks. PDP prioritizes the replies of the less limited devices, allowing the others to abort their answers. In this protocol, each device has a cache containing a list of the services that have been heard from the network. Each service has an expire time and the service is removed from the cache when they timeout. PDP uses Generic Service Description Language (GSDL), for description services. In PDP all messages are broadcast, and all devices cooperate by coordinating their replies and sharing the information in their caches. PDP protocol is to minimize the number of transmissions necessary to discover services, and so the battery consumption, especially of the most mobile and limited devices. There are two agents in PDP, PDP_UA and PDP_SA. PDP_UA is a user agent for searching information about services offered in the network on behalf of user. PDP_SA is service agent for advertising services that offered by devices. When a user or application need to a service, it calls its PDP_UA. In the first time PDP_UA searches the local area for service, if could not find, it broadcasts PDP_Service_Request [25].

SD using Centralized Directory

The centralized directory architecture rely on a central directory that stores the descriptions of all services available in the network so as to enable service discovery and invocation. Discovery of the directory by clients and service providers is in general based on multicasting. Then, service providers advertise their services to the central directory using a unicast message. And, to access a service, a client first contacts the central directory to obtain the service description, which is then used to interact with the service provider.

Centralized resource discovery is much suited to wireless infrastructure-based networks. However, this architecture makes the service discovery process dependent upon the availability of the central directory, which further constitutes a bottleneck. In addition, a centralized directory limits its scope to devices within a local service discovery domain. The boundaries of a service discovery domain can be administratively defined such as an IP subnet, or they can be the result of a physical property such as the range of a wireless network. Examples include JINI and SLP.

JINI

Java Intelligent Network Interface (JINI) is a protocol that has an environment for creating dynamically networked components, applications and services based on Java. In JINI there is a simple mechanism to support ad hoc networks, called JINNI federation. In the JINNI federation services and information may shared by users and it provides a mechanism for devices that they can join to network and detach form network dynamically without the need for configuration any devices. There is a main protocol in JINI called Lookup Service (central directory) that registers devices and services available on the network. When a device connects to network, it locates the lookup service and registers its service there (Service Advertisement), this device and its service are accessible by sending a query to lookup service (service discovery) [27].

SLP

The Service Location Protocol (SLP) provides a flexible and scalable framework for providing hosts with access to information about networked services. There are three main agents in the SLP framework: (i) User Agent (UA), issues a 'Service Request' on behalf of the client application, the User Agent will receive a Service Reply specifying the location of all services in the network which satisfy the request. (ii) Service Agent (SA), advertises the location and attribute on behalf of services and after receiving a request for a service it unicasts a reply containing the service's location. (iii) Directory Agent (DA), there is one or more DAs in a large network, they act as a cache and store information about the service announced in the network. SLP has two different modes of operation: (1) when a DA is present, it collects all service information advertised by SAs, and UAs unicast their requests to the DA, and (2) when there is not a DA, UAs repeatedly multicast the request. SAs listen for these multicast requests and unicast responses to the UA [28].

SD Using Distributed Directory

The motivation that support the use of the distributed directory architecture for service discovery is that scalability can be achieved when the network size becomes larger. This architecture is quite suited to the mobile ad hoc network scenario. Directories are dynamically selected among mobile nodes which have suitable capability (e.g. battery power, memory, processing power, node coverage, etc). It does not require pre-defined infrastructure. Mobile ad hoc networks are characterized by their highly dynamic, multi-hop, and infrastructure-less nature. If any fixed infrastructure is assumed in the protocol architecture, it would be against the nature of the ad hoc networking [8]. Example includes Sailhan.

Sailhan

Sailhan is a scalable service discovery protocol for

MANETs, which is based on the homogeneous and dynamic deployment of cooperating directories within the network.

It assumes a mobile ad hoc network composed of nodes holding the same network interface, with IP-level connectivity using the underlying routing protocol. However, it also considers that some nodes hold several network interfaces, and act as gateways with other networks, either ad hoc or infrastructure-based. This defines a hybrid network bridging mobile ad hoc networks and infrastructure-based networks, according to the specific networking capabilities of the wireless nodes.

One of long term research goal in [29] is interoperability among mobile devices , that pervasiveness of the Web allows assuming that a significant level of interoperability can already be achieved using the Web services architecture. Thus it considers a service-oriented architecture based on Web services, with wireless nodes hosting Web services and uses XML languages for describing Web services' interfaces. Specifically, it uses WSDL for service description.

Sailhan In [29], proposes a service discovery protocol aimed at large MANETs (i.e., comprising at least about 100 nodes, its design is based on centralized discovery architecture, as it induces less traffic. Directories are further distributed and deployed dynamically for the sake of scalability. Specifically, its discovery architecture is structured as a virtual network. A virtual network is composed of a subset of nodes of the MANET acting as directories. These directories represent a backbone of nodes responsible for performing service discovery. They are deployed so as at least one directory. It is reachable in at most a fixed number of hops, *H*, whose value is dependent upon the nodes density. Directories cache the descriptions of Web services available in their vicinity which is defined by *H*. Hence, wireless nodes that do not act as directories do not have to maintain a cache of service descriptions, and the network is not flooded by service advertisements. 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. 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 to a minimum the generated traffic [29].

Table 5 shows an overview over comparison Directory Architecture protocols for service discovery of mobile ad hoc networks.

In Table 5, Sailhan search method is Centralized and other search method protocols are pull and push. Security in JINI is defined based on Java security.

Comparison of the Various Directory Architectures

In directory-less architectures, broadcasting is generally used for service discovery and advertisement. These broadcasting mechanisms are not suited for mobile ad hoc networks due to their heavy consumption of bandwidth and energy, which are limited in mobile devices. Therefore, the network size supported by the directory-less architecture is very limited. Nevertheless, in regions with extremely high mobility, broadcasting could be the only possible technique.

In the central directory architecture, although Centralized resource discovery is much suited to wireless networks but the central server further constitutes a bottleneck. In addition, a centralized directory limits its scope to devices within a local service discovery domain.

Distributed directory architectures are quite well suited to the mobile ad hoc network scenario, but when we have many nodes in the network, the overhead will increase exponentially.

Table 6 provides a comparison of Service Directory Strategies for service discovery in mobile ad hoc networks. In order to have an efficient service discovery and service advertisement architecture, it is better to combine the distributed directory architecture and central directory architecture approaches to achieve the benefits of both.

4. Proposed Enhancement to SD Mechanisms for MANETs

Service discovery on a fairly large mobile ad hoc network must employ Zero Configuration network technologies, to minimize flooding, energy consumption and communication overheads. This would be achieved in proposed enhanced Hierarchical SD protocol via:

(i) Clustering mechanism to combine distributed architecture for maintaining service information with centralized stores for each cluster to decrease delay for finding a service in the network.

(ii) Hybrid mechanisms for Service Discovery using network and application layers approaches to decrease connection overhead and power consumption. Network layer finds Service based a request and Application layer finds a well suitable Service heuristically.

Hierarchical service discovery and advertisement is based the concept of clusters. Each cluster consists of one Cluster Head (CH) and an arbitrary number of Clustered Nodes (CNs). Both CH and CNs can offer use services. In each cluster, the CH acts as a representative for its CNs in terms of service discovery and the CH has a public view of the entire network but each of the CNs has a complete

view only of its own cluster. Each CN can send packets to another cluster only via CHs.

Hierarchical Service advertisement is started when a service is added to a machine located within the ad hoc network environment and the owner decides to share the service based on some criteria.
In contrast to existing clustered SA approach, service attribute is divided into 2 categories, static and detailed attribute. Detailed information including static service attributes, service access models and the service physical location is sent to every machine in the same cluster. In this way, all machines in the cluster are aware of the exact details of the service.
Advertisements of the service to machines in other clusters involve sending only the static attributes of the service. Only one message is sent to the destination cluster's cluster head for distributing the service advertisement. The cluster head, having received the message, would then broadcast the message to all of its neighboring machines within the cluster. This approach is expected to improve the accessibility of services information and decrease communication overhead.

Hierarchical Service Discovery means that, client first searches for the required service within the same cluster. If the queried service is shown to be suitable for the waiting task it is selected and the discovery process is terminated. But if a suitable service could not be found, search of the other clusters using hierarchical service discovery will be performed. In addition, searching service in all of adjacent clusters is done parallel and the number of moving packet between clusters is a few then this approach also is expected to decrease communication overhead and improved service finding.
Moreover in this approach we use intelligent agents in layer seven for electing the most suit service, hierarchically.

5. Conclusion

SD and SA are very important issues in MANETs. Choosing an unsuitable SD and SA protocol will increase overheads network communications and power consumption. As a result this will also increase delay for finding a suitable service in the MANET.
Protocols that implement service discovery at the routing layer by implementing through piggybacking methodology, significantly reduces the communication and energy consumption overheads.
In comparison of the various directory architectures, Central and Distributed Directory Architectures are well suited to the mobile ad hoc network scenario but in the Central Directory the central server further constitutes a bottleneck and in the Distributed Directory, when we have many nodes in the network the overhead will increase exponentially. In attentive to the above term, to an efficient service discovery and service advertisement, without the above limitation, it is better to combine the distributed directory architecture with central directory architecture. In this case we will make use of the benefits of both of them and they will cover the weak point of each other.
The proposed algorithms will be tested using simulation approaches to verify its suitability for improving SA & SD performance. This approach is expected decreases communication overhead and improves the accessibility of services information and service finding.

Table 5. Comparison of Directory Architecture protocols

Protocol	Network Type	Network Architecture	Storage Information	Service Description	Scalability
UPnP[8,11]	Enterprise	Point 2 point	Any devices	XML	N/A
PDP[18]	Ad hoc	Fully Distributed	Any devices	GSDL	N/A
DEAPspace[19]	Ad hoc	Point 2 point	Any devices	General	N/A
JINI[20]	Enterprise	Centralize	Central Directory	Java objects	Service Grouping
SLP[21]	Small Scale Ad-hoc & enterprise	Point 2 point & centralize	Central Directory	General	service scope
Sailhan[22]	Ad hoc	Hybrid	Distributed	XML(WSDL)	Scalable SD

Table 6. Comparison of Directory strategies

Architecture ▶ Parameters ▼	Directory-less	Centralized Directory	Distributed Directory
Store Information	Stores the descriptions of services on their own device	Stores the descriptions of all services available in a directory	Directories are dynamically selected among mobile nodes
Network size	very limited	Limited	not limited
Search Method	Use reactive service trading	Use proactive service trading	Use proactive service trading
Problem	Heavy consumption of bandwidth and energy,	Further constitutes a bottleneck.	Complexity of management of network
Scalability	N/A	Service discovery is scalable when the network size becomes larger	Service discovery is scalable when the network size becomes larger

Service Request	Normally Use Broadcast	Normally Use Unicast	Normally Use multicast
Advantage	In regions with extremely high dynamics, broadcasting could be the only possible technique	Response time for locating services reduced, Scalability is achieve, Servers are not flooding with service requests, Can use load balancing algorithm.	Response time for locating services reduced, Scalability is achieve, Servers are not flooding with service requests Can use load balancing algorithm,improved efficiency

6. References

[1] Liang Cheng and Ivan Marsic,Service Discovery and Invocation for Mobile Ad Hoc Networked Appliances, University of New Jersey, 2nd International Workshop on Networked Appliances (IWNA'2000), New Brunswick, NJ (December 2000).

[2] C. Siva Ram Murthy and B.S.Manoj, Ad hoc Wireless Networks, Architectures and protocols pages 204,214,Prentice Hall, 2004.

[3] Choonhwa Lee, Service Discovery for MANETs, College of Information and Communications Hanyang University June 28, 2006

[4] Zhong Fan and Eduardo Guerreiro Ho, Service Discovery in Mobile Ad Hoc Networks,IEEE,2005

[5] M. Gerla, G. Pei, S.J. Lee, and C.C. Chiang ,On-Demand Multicast Routing Protocol (ODMRP) for mobile Ad-Hoc Networks, WAM Lab, UCLA, December 1998

[6] Christopher N. Ververidis and George C. Polyzos , Routing Layer Support for Service Discovery in Mobile Ad Hoc Networks, 2005 IEEE

[7] Li Li and Louise Lamont, A Lightweight Service Discovery Mechanism for Mobile Ad Hoc Pervasive Environment Using Cross-layer Design, 2005 IEEE

[8] Chunglae Cho and Duckki Lee. Survey of service discovery architectures for mobile ad hoc networks. Unpublished 2005, Computer and information sciences and Engineering Department, University of Florida Gainesville, USA.

[9] Rolf Gr¨uninger, Service Provisioning in Mobile Ad hoc Networks, Master's Thesis,swess federal institute of technology zurich,2004

[10] Gregor Schiele, Christian Becker and Kurt Rothermel, Energy-Efficient Cluster-based Service Discovery for Ubiquitous Computine, Institute for Parallel and Distributed Systems (IPVS)Universität Stuttgart, Universitätsstr. 38, 70569 Stuttgart, Germany, 11th ACM SIGOPS European Workshop, 2004.

[11] UPnP Forum: Understanding Universal Plug and Play White Paper, http://www.upnp.org (2000)

[12] M. Nidd: Service Discovery in DEAPspace. IEEE Personal Communications, (2001) 39-45

[13] Jaehoon Jeong, Jungsoo Park, Hyoungjun Kim,Service Discovery based on Multicast DNS in IPv6 Mobile Ad-hoc Networks,Protocol Engineering Center, ETRI,2003,IEEE

[14] IEIT Zeroconf working p u p ,http://www.ietf.org

[15] S. Lee, W. Su, J. Hsu, M. Gerla, and R. Bagrodia, "A performance comparison study of ad hoc wireless multicast protocols," *Proc. IEEE Infocom'2000*, pp. 565-574, Tel-Aviv, Israel, March 2000.

[16] E. Bommaiah, M. Liu, A. McAuley, and R. Talpade, "AMRoute: adhoc multicast routing protocol," *Internet Draft*, work in progress, August 1998

[17] S. Lee, M. Gerla, and C. Toh, "On-demand multicast routing protocol (ODMRP) for ad hoc networks," *Internet Draft*, work in progress, June 1999.

[18] C. Wu, Y. Tay, and C. Toh, "Ad hoc multicast routing protocol utilizing increasing id-numbers (AMRIS) functional specification," *Internet Draft*, work in progress, November 1998

[19] protocol," *IEEE Journal on Selected Areas in Communications*, vol. 17, no. 8, pp. 13801394, August 1999

[20] F Zhu, M Mutka and L Ni, "PrudentExposure: A Private and User-centric Service Discovery Protocol", Proceedings of the Second IEEE International Conference on Pervasive Computing and Communications, PerCom'04, Orlando, Florida, USA, March 2004

[21] Ulas C. Kozat and Leandros Tassiulas. Network Layer Support for Service Discovery in Mobile Ad Hoc Networks Proceedings of IEEE INFOCOM 2003, April 2003]

[22] R. Koodli and C. E. Perkins, "Service discovery in ndemand ad hoc networks," IETF Internet Draft, draft-koodlimanet-servicediscovery-00.txt, October 2002.

[23] Victoria Beltrán Martínez ,MOBILITY IN TCP/IP NETWORKS Politechnic University of Catalunya (UPC) Phd CURSE ,June, 2006

[24] Dipanjan Chakraborty, Anupam Joshi, Yelena Yesha and Tim Finin.GSD: A Novel Group-based Service Discovery Protocol for MANETS.4th IEEE Conference on Mobile and Wireless Communications Networks (MWCN 2002), Stockholm, September 2002.

[25] C. Campo, M. Munoz, J. C. Perea, A. Marın, C. Garcıa-Rubio: PDP and GSDL: a new service discovery middleware to support spontaneous interactions in pervasive systems, Proceedings of the 3rd Int'l Conf. on Pervasive Computing and Communications Workshops (2005).

[26] M. Nidd: Service Discovery in DEAPspace. IEEE Personal Communications, (2001) 39-45

[27] Sun Microsystems. Jini Network Technology. http://wwws.sun.com/software/jini/

[28] Erik Guttman, Charles Perkins, John Veizades and Michael Day. Ser-vice Location Protocol, Version 2 RFC 2608, Network Working Group,Internet Society, June 1999.)

[29] F. Sailhan, V. Issarny: Scalable Service Discovery for MANET, Proceedings of the 3rd IEEE Int'l Conf.on Pervasive Computing and Communications (2005)

[30] Adnan Noor Mian, Roberto Beraldi, Roberto Baldoni, Surve od service discovery protocols in Mobile Ad hoc Networks,technical report 4/06, Dipartimento di Informatica e Sistemistica "Antonio Ruberti" Università degli Studi di Roma "La Sapienza" Via Salaria, 113 – 00198 Rome, Italy.

Seyed Amin Hosseini Seno received B.Sc and M.Sc., degree in Computer Science from Ferdowsi University of Mashhad, Iran in 1990 and 1998 respectively. Currently, he is a PhD Student at Network Research Group (NRG)/ National Advanced IPv6 (NAv6) lab of School of Computer Sciences, USM. His current research interests Service Discovery and Service Advertisement in MANETs.

Rahmat Budiarto received B.Sc. degree from Bandung Institute of Technology in 1986, M.Eng, and Dr.Eng in Computer Science from Nagoya Institute of Technology in 1995 and 1998 respectively. Currently, he is an associate professor at School of Computer Sciences, USM. He is the deputy director of National Advanced IPv6 (NAv6) Center, USM. His research interest includes IPv6, network security, Ethnomathematics and Intelligent Systems. He is chairman of Security Working Group of APAN.

Tat-Chee Wan received his BSEE and MSECE from University of Miami, FL in 1990 and 1993 respectively; and his Ph.D from Universiti Sains Malaysia, Penang, Malaysia, in 2005. He is currently Programme Chairman for Computer Systems at the School of Computer Sciences, Universiti Sains Malaysia. His research interests include Wireless and Sensor Networks, Multicast protocols, QoS and embedded real-time systems.