An Enhanced Service Ring Protocol for Service Discovery in MANET

Nidhi Singh
Department of CS &IT
SHIATS, ALLAHABAD
U.P. India

Hari Mohan Singh
Department of CS & IT
SHIATS, ALLAHABAD
U.P. India

Shiv Pratap Pal
Kendriya Vidyalaya No.1
Akhnoor, J&K
India

ABSTRACT

Semantic service trading is a powerful mechanism to allow devices to share the resources and services in highly dynamic environment. Mobile devices are able to spontaneously form highly dynamic Ad-hoc networks allowing the communication even when no infrastructure exists. These types of networks enable the participating devices to share resources and services and thus overcome the often severe limitations of the individual device capabilities. The devices cooperate with each other by providing different types of services. These services must be discovered before they can be used. Locating a network service or a device on demand is a challenging task for enabling mobile and pervasive computing. There exists a variety of network protocols and architectures that enable an application to discover a network service with little manual configuration. In this paper our studies is focused on Service Rings Protocol and suggests caching to improve its performance. The modification increases the performance and efficiency of Service Ring protocol for service discovery in MANET by reducing the flooding and exchange of service discovery messages.

General Terms
Service Discovery , Service Ring , Caching, MANET et al.

Keywords
Enhanced Service Ring ,Service Discovery, Overlay Structure, Service Access Point .

1. INTRODUCTION

The last few years witnessed a rapid increase in the usage of mobile devices such as laptops, cell phones and personal data assistants (PDAs). The accompanied maturity in wireless technology such as 802.11 has made wireless networks almost as ubiquitous as traditional wired network. An important consequence of such developments has been the concept of Ad-hoc networks. These networks are characterized by their lack of infrastructure and ease of formation. Each participating device is mobile and the networks are formed temporarily. For example, a network is formed when a group of people comes together in a conference, in an emergency relief scenario or in airport lounge. As in the more traditional wired environments, one problem is service discovery. Mobile devices constituting the wireless ad-hoc network due to their small size and mobility are inherently scarce in resources, which necessitate the need to cooperate among them for performing operations that cannot be done alone. At application level this cooperation can take the form of services following the paradigm of so called service oriented architecture. To get benefit from the services offered by other devices, they have to be searched. Search and Service Discovery Protocols (SDPs) are used for this purpose. Numerous efficient Service Discovery protocols for MANET is discussed in the literature. In this paper our studies is focused on Service Ring Protocol and suggests caching to improve its performance. Some of the possible applications and scenarios for Ad-hoc network are as follows:

- Using wireless handheld device, the ability to locate a nearby restaurant serving favourite food.
- The ability to share with others entertainment sources such as music and games available on personal handheld device. Such a capability would be extremely useful while waiting for a flight in an airport lounge or other similar situations of killing time in a public place.

In all the above mentioned scenarios, the resources being shared or searched are packaged as service. As service is any tangible or intangible facility a device provides that can be useful for any other devices. Services may comprise of software as well as hardware. A software service, for example can be a simple file, such as an MP3 file or software implementation of an algorithm, such as converting one audio format to another. A hardware service , for example, can be a printer that a mobile device can use wirelessly. To benefit from these services, a device must be able to locate them in the network and invoke them. Service discovery protocol enables both the locating and invoking of services.

The feasibility of the above scenario requires not only the formation of Ad-hoc networks and packaging of resource as services, but also a discovery mechanism suited to the needs of Ad-hoc networks. Low level technologies necessary to form peer to peer ad-hoc network are available. The missing link is higher-level framework and protocol, which will enable devices to discover and advertise their services over ad-hoc networks.

Traditionally, service discovery protocols have focused on services provided by devices such as printers, fax-machines, cameras, audio-systems, etc. For wired networks, many service discovery techniques have been proposed example, Jini , UDDI , SLP, Service Ring. In all these techniques there is a Central Service Repository which facilitates services / user agent discovery within a community. This Central Repository is a special node which is also termed as SAP i.e. service access point or directory nodes in different architecture. It is obvious that, in order to obtain effective matches between services offered and requested, a semantically rich description and query language is necessary, especially when the service requests negotiate with the services available without the help of human users.

Service Rings form an overlay structure by grouping nodes that are physically close to each other and offer similar services. Each service ring has designated SAP that keeps the directory and has information about available services in the service rings. The SAP's are connected with other service rings SAP. When a node wants to search for a service , the query is routed through the ring structure, passing through SAPs of other rings and reaches only those sub-rings that can possibly offer the service. If this service information is cached by intermediate SAPs when query response is routed through the ring then it becomes available for later discovery. So caching service information will improvise the service discovery process. Caching services is a potentially important application for MANETs as it can improve mobile user's perceived quality of service.

Caching is a common technique used in various areas of computer science to improve performance, scalability and quality of service where some form of locality exists. When considering network based services offered to mobile clients, it is likely that multiple clients in the same MANET, or even the same region of MANET, will try to access the same service concurrently. This suggests that caching such services within the MANET would be beneficial. Thus, this paper deals with caching issue in Service Ring protocol. Here first of all Service Ring protocol in service discovery is examined and performance is evaluated. Thereafter we will present enhanced Service Ring with a cache based approach and then compare the performance with an existing approach.

2. RELATED WORK

There have been intensive research efforts in the field of service discovery protocol. There are number of platforms, architecture, multi-agent communication and collaboration. IETF SLP[12], Sun's Jini[17], UPnp[19] and Salutation are some of the front-runners in this area. The Service Location Protocol is one such protocol. SLP is a protocol for automatic resource discovery on IP based networks. It implements discovery mechanism on service attributes, which are essentially the different ways of describing a service. SLP service agents are responsible for advertising service handles to the directory agents thus making service available to the user agents. This protocol also supports a simple service registration leasing mechanism.

Another service discovery architecture is Jini. Jini is a distributed service –oriented architecture developed by Sun Microsystem. Jini lookup services maintains dynamic information about the available services in a Jini federation (a collection of Jini services). When a Jini service wants to join a Jini federation, it first discovers one or many Lookup Service from the local or remote networks. The service then uploads its service proxy (i.e. a set of Java classes) to the Jini Lookup Service. The service clients can use this proxy to contact the original service and invoke methods on the service.

Yet another solution to service discovery is Universal Plug and Play (UPnP) , supported by Microsoft. This architecture is designed to extend the original Microsoft Plug and Play peripheral model. UPnP works primarily at the lower layer network protocol suite (i.e. TCP/IP). UPnP uses the Simple Service Discovery Protoocl (SSDP)[7] 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 URL that identifies the advertising service and a URL to a file that provides description of the advertising services. 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.

Another framework that was developed for dynamic distributed systems is the Ronin Agent Framework. Ronin is a Jini based distributed agent development framework. It offers a simple but powerful agent description facility that allows agents to find each other and an infrastructure. In Ronin , agents are described using the Common Agent Attributes and the Domain Agent Attributes. The Common Agent Attributes defines the generic functionalities and capabilities of an agent in a domain-independent fashion. The Domain Agent Attributes defines the domain specific functionality of an agent.

Also there are several FIPA complaint platforms. One of such platforms is JADE distributed by TILAB. Another platform is FIPA-OS. Both JADE and FIPA-OS are convenient tools for agent application development however they are intended for a static network infrastructure and are not really meant to handle frequent disconnections and various other aspects of the dynamic nature of ad-hoc environments. The limitations of JADE and FIPA-OS were recognized and LEAP and MicroFIPA-OS were developed to extend the functionality and allow mobile devices to participate. MicroFIPA-OS and LEAP have smaller footprint than the original framework and fits well on the majority of the mobile devices like cell phones and PDAs. However the light weightness came at a price of functionality. The mobile devices running LEAP or MicroFIPA-OS host only small and most critical parts of the platform. This setup forces the devices to heavily depend on the static and resource rich infrastructure around them to advertise or host agent management systems and directory facilitators. Though, it is clear that these protocols were the steps in the right direction. This resulted in FIPA to introduce the concept of structured compounds between agents operating in ad-hoc environments to facilitate agent discovery and interoperability.

Service discovery by flooding like the Simple Service Discovery Protocol (SSDP) and the file sharing protocols like Gnutella and JXTA-Search. Hash-based Approaches like OceanStore, Globe, Chord[9], Freenet , Tapestry[3] and CAN. Semantic Routing[11] i.e. intelligently choosing nodes to forward a request by inspecting its semantics, like the Intentional Naming Scheme (INS). Allia[14] is another static directory based service discovery. Konark[18] uses another approach. In this a middleware is designed specifically for discovery and delivery of device independent services in ad-hoc networks. Service Ring is another approach based on semantic service discovery on transport layer overlay. It is based on Central Service Repositories in file sharing.

3. SERVICE DISCOVERY PROTOCOLS

Mobile devices constituting the wireless Ad-hoc network, due to their small size and mobility are inherently scarce in resources, which necessitate the need to cooperate among them for performing operations that cannot be done alone. At application level this cooperation can take the form of services following the paradigm of so called service oriented architecture. To get benefit from the services offered by other devices, they have to be searched. Search and Service Discovery Protocols (SDPs) are used for this purpose.

Service discovery for distributed environments is often designed for Internet based networks and thus does not take the characteristics of mobile Ad-hoc networks into account. Various Service Discovery Protocols are discussed in previous section. Now the question is how these service discovery protocols carry out the process of Service Discovery? For this there is a framework which provides an ordered way to execute service discovery. The framework is discussed in next part of this section.

3.1 Service Discovery Framework

The basic building blocks of service discovery framework used in MANETs is shown in figure 1. This framework consist of certain steps that needs to be followed during discovery process.

These steps that make a complete SDP are the description mechanism, registration mechanism, discovery mechanism consisting of search methods, selection mechanism, routing mechanism, and mobility support mechanism. A user, which can also be an application software acting on behalf of the end user, interacts with the SDP. The Figure 1, depicts the basic mechanism and the arrows represent how the flow of service discovery process takes place. It is shown that the user or any application program starts the process with description mechanism. This is the first step to which the user interacts. It describes a service in a format used to register or to search a service. A proper description facilitates searching of a service. The following section tends to discuss the service discovery elements.

i. Service Description

Service description is an abstraction of the facilities and characteristics of a service. The description of a service is necessary, if it is to be utilized by other devices or services. The nodes searches for the services only by looking at the descriptions of the services advertised by the service provider. A service may remain completely unknown to other devices in the network if it is not properly described. Description of a service includes service properties like ID of a service, server properties like load on a service provider, and network properties like maximum number of hops from the current host to a service provider.

ii. Registration

After describing the service using some format, next step is to store the service description somewhere so that when other nodes needs the service of this node can discover this node. Thus the Registration step helps finding information like where the service is stored and how long to store the service description. Also it includes information like no. of hops the information will travel. There are several ways to manage service description. When the node is registered only in local cache then to find the service search must cover all the nodes Or if the description is stored in some special designated nodes , then search nodes are called directory nodes and request will be routed to the special node only. Flooding is another way to store service description in which search request is broadcasted to all the nodes in the network.

iii. Discovery

SDP framework provides an ordered way to execute service discovery process. After Describing the service and registering the service information the next step is to facilitate the service to nodes which are looking for the stored service information. This step of SDP framework is responsible for the discovery of required service. Discovery can be done in two ways. First is to search the service and second is the selection of service using criterion-based algorithm or manually. These two mechanisms can be independent and integrated. For example, an SDP might first search for all instances of a service provider and then select a suitable service, or might perform the two functions simultaneously.

- **Search:** The search mechanism is responsible for searching services that have already been registered by the registration mechanism. In network that doesn't have special directory nodes to keep service description, search for a service depends on how the registration mechanism has registered the services. In SDPs**,** that register services in the local cache, search takes place in all nodes by flooding the search messages. Another search approach is to selectively forward search requests.

- **Selection:** The query requests from a client node to the network can result in many responses from the matching services. Selection from among the discovered services is then essential to invoke one of the services. Selection can be done manually by the user or automatically, Using criterion based algorithm. When a node does not have enough information to selectively forward a request, it simply broadcasts the request to its neighbouring nodes.

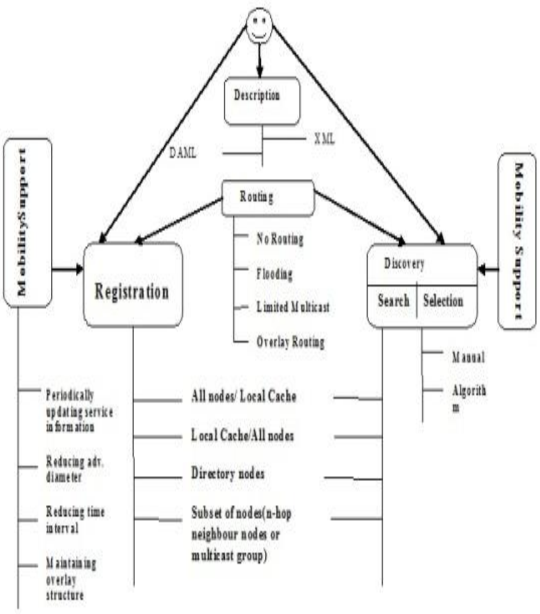

Figure 1: Service Discovery Protocol Framework

iv. Routing

The SDPs that registers or searches in all the network's nodes floods the query or registration messages. An SDP can scale well if the traffic generated in the network is limited. To limit the network traffic, SDPs employ different routing mechanisms. For examples, SDPs limit the flooding to n hops or use multicast routing to specially formed groups of service providers, or use selective forwarding based on the notion of potential. Routing is done on the basis of overlay network formed as in the Service Ring protocol.

v. Mobility Support

Mobility is a challenge for SDPs with directory nodes. One approach for handling mobility is to periodically update the service information. When a service provider advertises its service, it also announces timeout, which goes to state, when the service information will expire. The service provider must renounce the service information periodically. In SDPs that form overlay networks there are special algorithms for maintaining the structure of overlay, which gets faulty due to the mobility and logging in and logging out of nodes from the network. Service rings have special algorithms to maintain the consistency of the ring overlay. Each ring member knows only its successor and it predecessor. RingCheck messages are initiated periodically by the appropriate Service Access Points. The messages, circle through each ring to check its consistency.

3.2 Service Ring Protocol

It is a new approach for semantic service discovery based on transport layer overlay. A service ring ideally groups together devices that are both physically close to each other and offer similar services. Each ring possesses a designated Service Access Point (SAP) which knows a summary about all the services offered within its ring. SAP can be connected to a ring too, which leads to a hierarchical structure. Service rings are overlay structure on top of the transport layer of an Ad hoc network. These networks are accessed through their transport layer interface, and if this network is not partitioned then each node can send messages to other nodes using one of the Ad-hoc routing. An Ad-hoc network can be regarded as graph with every node ni representing a device and each edge ni nj representing the possibility to send messages between two devices. Yet the goal of every overlay structure is to minimize the number of messages that are necessary to perform a certain task(e.g. Service search) by exclusively using selected edges. Moreover only these chosen edges have to be maintained which are adapted to the topology of the network. We call these edges overlay links and denotes them as n_i * n_j.

3.2.1 Service Access Point

There is one special designated node in each ring. This node is termed as Service Access Point of the Service Ring. SAP consists of summary about all the services offered within its ring. SAPs consist of directory of information about services offered by other nodes within its ring. SAPs can be connected to other SAPs of the other rings which leads to a formation of service ring at next higher level. Thus in this way SAPs can form hierarchical structure. Thus in turn m level of service ring can be formed. The highest level of Service Ring is called World Ring.

3.2.2 Level- 1 Service Ring

In Figure 2, For L=1, here SAPs of R0 Service ring form a higher level service ring , if it fulfils these conditions:

a) Each device forms a S_0 service ring.
b) For every device n_i, n_{i+1} : n_i * n_{i+1} and forms an overlay link.

The SAPs of level 0 service ring R_1, R_2, R_3 and R_4 are forming the level 1 ring with node n_6 (node$_6$)as its Service access Point. With these terms, we can introduce the term valid Service Ring Overlay. Generally an overlay is valid if:

a) All devices are member in exactly one level, 0 service ring.
b) All SAPs of level 0 rings are members in a level-1 service ring.
c) There is one highest level 1 service ring, which is called the World Ring.

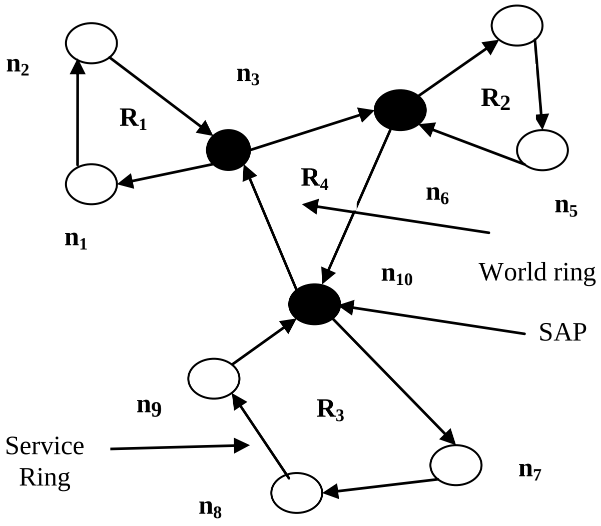

Figure 2: Level -1 Service Ring

To get an efficient overlay structure that helps to route service requests semantically, some additional criteria is required for semantically good service ring overlays. The first thing we need is to describe the service. Also there needs to exist a distance function dist on the service description, allowing to compare different service description and to allow how semantically similar they are. Service discovery process is carried out in the Service Ring following the steps already discussed in Service Discovery framework.

4. ENHANCED SERVICE RING PROTOCOL

In service ring protocol where similar services are arranged together, there may be number of higher level service ring consisting of number of nodes. Each node may offer some services. In this approach a special node was considered for maintaining information about other nodes forming an overlay link with this node in a service ring. Whenever any node request service to its SAP in the network, first it checks the availability of service in its own ring, if service is found then no further service discovery steps are followed. But if the request is not found then SAP forwards the request to next higher and closer SAP. In this way request is forwarded in hierarchical fashion to the Service Ring where service is located.

Lets consider a case when a node N_1 of service ring R_1 requests for a service S_{11} to its SAP (say node N_2) the service (S_{11}) is available at another Service Ring say R_2. Thus by service discovery process, SAP N_2 will forward the request to another SAP at the next higher level Service Ring and may find and avail the service and send reply to lower level SAP i.e. node N_2 and N_2 responds to node N_1. But if node N_1 or other nodes of same Service Ring R_1 i.e node N_3 , N_4 or N_5 requires the same service again, then again service discovery process has to be executed. For instance, in the service discovery scenario, SAP node N_2 will not learn about service S_{11} information and again it requires to go through the service discovery procedure.

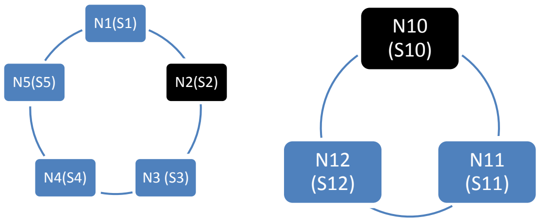

Figure 3 : Service Ring R1(left) Service Ring R2 (Right)

To improve performance of Service Ring, Caching may prove to be very useful. When considering network based services offered to mobile clients, it is likely that multiple clients in the same MANET, or even the same region of a MANET, will try to access the same service concurrently. This suggests that caching such services within the MANET would be more beneficial.

We propose a simple caching optimization in Service Ring protocol of service discovery. All intermediate nodes that are also participating as Clients or User Agents (UAs) in the service discovery process shall cache service information found in the Service Replies (SrvRplys) that they relayed. In this way, if one of these intermediate UAs wishes to use a service that was previously cached, it needs not issue a new Service Request (SrvRqst) and can use the cached service. UAs need not generate SrvRplys for cached service information. This will improve the Service Discovery process and searching will be more efficient. Therefore in the discovery scenario illustrated in Fig. 3, when N_1 requests service $\mathbf{S_{11}}$**, SAP** $\mathbf{N_2}$ or other SAP following the route to Service Ring R_2, re-broadcasts the SrvRqst as usual. According to enhanced Service Ring approach:

i) When other UAs N_3 , N_4 and N_5 of the same service ring R_1 needs the same service again they do not have to follow the long and time consuming service discovery process again. These nodes can get service information from their Service Ring.

ii) Also MANET comprises of numerous service rings . The nodes of other closer and neighbour Service Rings of R_1 may request for the same service. Instead of following long procedure of Service Discovery, they can obtain the reply from their neighbour SAP R_1.

iii) When a service request is generated at any level in the Service Ring and is routed to some higher level ServiceRing through intermediate SAPs in between, Then when a Service Reply is forwarded following the same route, these intermediate SAPs will also cache the service information. When nodes of these SAPs or neighbour SAPs needs the same service previously searched, can obtain the same from the cache. Hence the Caching will also improve availability of service information and Service Discovery process will be more effective.

The nodes offering similar services are arranged together and one special designated node SAP keeps all the information of nodes in the Service Ring. Whenever any new node wants to enter into Service Ring, node registers itself by sending message to SAP. So in large network different service rings will be formed and their SAPs will be connected with each other to form a higher level Service Ring. In this way, hierarchical structure of Service Ring will be formed. When a node wants to search for a service, it calls the service search algorithm and the query is routed through the ring structure but only descends in those sub-rings that could possibly offer the requested services.

5. EXPERIMENTAL RESULTS

5.1 Protocol Performance of Cached Service Ring

We have discussed earlier the Service Ring protocol with and without caching approach. Service Ring is nothing but collection of mobile nodes forming an overlay link. Thus if the level of service ring increases or decreases, the other parameters associated with it will also change.

Experimental parameters:

- Let L be the number of levels in Service Ring.
- Let R be the service Rings formed, the relation between Service Ring and no. of nodes can be established as:

$$L = \log_2 R \quad \ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots\ldots. (1)$$

We are not taking into consideration Propagation delay as it will be the same for any MANET. Hence

$$\textit{Delivery time} = \textit{transmission time taken in one hop}\ldots\ldots(2)$$

- Let T_s be the time to search request at any hop. i.e *Search time.*
- Let T_{ud} be the *Updation time* i.e. time taken to update the service information at each hop in caching approach.

As we have already discussed in previous section about existing approach of Service Ring in MANET. Whenever the UserAgent sends Srvrqst to its SAP, the SAP forwards the SrvRqst to higher level SAP from which it is forwarded again to the next higher level SAP until the request gets processed. In the same way SrvRply is forwarded to descendent sub-rings. The intermediate hops simply forward the response message to lower level SAP until it reaches the concerned SAP. There is no additional time taken in sending response to the UserAgent. The expression for the total time taken in Service Ring (Without Cache) can be written as:

$$T_{tot} = 2 * L * \textit{AVG delivery time} + L * \textit{search time}$$

$$T_{tot} = L * (2 * \textit{AVG delivery time} + \textit{search time}) \ldots\ldots(3)$$

While in the proposed Service Ring protocol there is an overhead of updation of information at each intermediate SAP during response of the Servreq. As each intermediate SAP will update this information in its cache, therefore the next time discovery will be fast and less time consuming. Thus the total time taken in Extended Service Ring (With Cache) can be written as:

$$\textit{Tresptot} = 2 * L * \textit{AVG Delivery time} + L * \textit{Search time} + L * \textit{updation time}$$

$$\text{Tresptot} = \text{L (2 * AVG delivery time + Search time + updation time)} \quad \ldots\ldots(4)$$

Or we can simply calculate the transmission in level Service Ring by assuming transmission time only . i.e the time taken in sending service request SrvReq from its local cache to the destined cache or SAP.

Let T_{local} be the time to reach its local SAP , then time used to reach its destination SAP at any level. So, the total updation time (T_{upd}) will be-

$T_{upd} = L * T_{local}$ ……………………………………………..(5)

We can evaluate time as some function of time i.e. O(Tupd.)
Thus in Worst case –

$$O(T_{upd}) = L * T_{local}$$

In Best case L=1 that is its own service ring hence

$$O(T_{upd}) = T_{local}$$

The best case result shows that when a service is already available in the Service Ring then node can obtain the query result from its own SAP. Therefore time taken will be equal to T_{local} .

Also with the help of derived expression/formula (1-5) we can find out the effect of cached approach over traditional Service Ring protocol. Assuming the few parameters as a constant the total response time can be evaluated.

Next, A graph is plotted between Average Response time vs No. of Levels in Service Ring. .

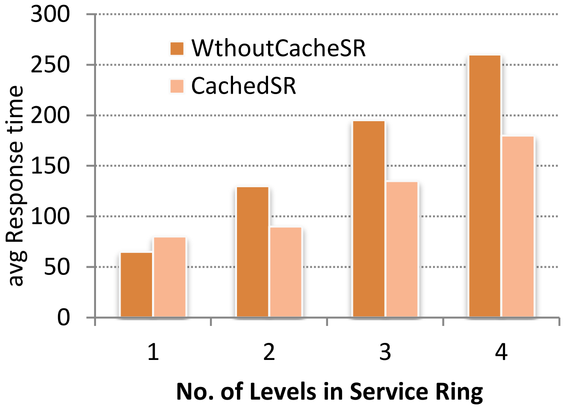

Figure 4: Average response time

The Figure 4 graph represents total response time . As we can see, in level -1 the cache approach in Service ring is taking more time in comparison to existing without cached approach because when first time Service Request is responded by any SAP , then it needs to be updated at each intermediate SAP along with the SAP of Client. But when Service Request for the same SAP is sent again next time , it will not take any updation time i.e. Updation time=0 while the value of search time will be reduced. Here we have assumed TotalDelivery time and Search time as constant in previous approach (without cache). While in Cached Service Ring approach updation time =0 for the same service request sent next time. We have assumed that query request is sent for the same service again for which the information is already served .We can say that caching has improved the Service Ring performance as it will result in less service discovery time and increased availability of service information at SAPs of different levels of Service Ring.

As in this proposed work, we have tried to improve the performance of traditional Service Ring , It can be observed and estimated that how the change in Cache Capacity is going to affect the Average Response time of Service ring. having different Cache Capacity. The average service discovery time with different cache capacity shown in Figure 5 and Figure 6.

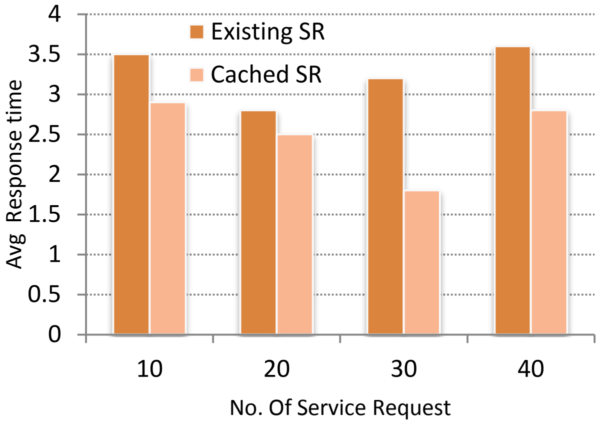

Figure-5. Average Service Discovery time (Cache capacity =10)

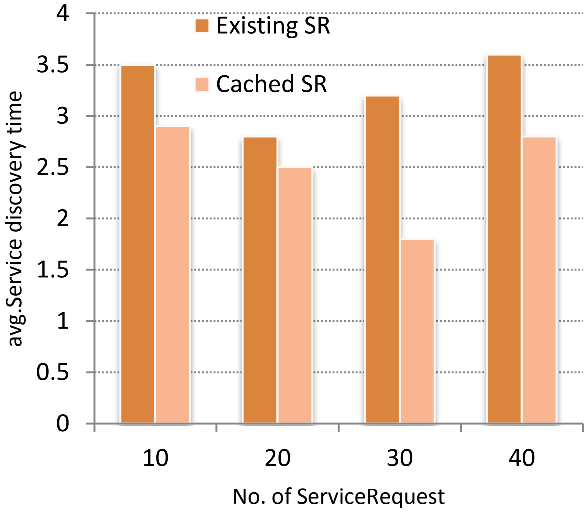

Figure 6. Average Service Discovery time (Cache capacity =15)

The above graphical representation results the average response time with change in cache capacity size. The average service discovery time takes less time in cached approach as compared to without cached protocol. With the change in cache capacity the avg. Service discovery time is evaluated.

An existing Service Ring protocol shows the same result in both cases while cached Service Ring reduces the service discovery time.

Thus the results shows that the proposed approach will improve the performance and also have the following advantages.

1. Reduces overhead of service discovery every time.
2. Increased Service availability and accessibility.

5.2 Conclusion

In this paper we have presented simple caching modification in Service Ring Protocol for service discovery in MANET where intermediate SAPs (Service Access Points) cache service information found in Service Replies (SrvRplys) which they relay. In the previous approach of Service Ring protocol the summary of information about nodes and services are kept only at the concerned SAP and not on the other or neighbor SAP which takes enough service discovery time in network. Hence caching is the solution which improves the performance as well as reduces the service discovery time. The modification increases the performance and efficiency of Service Ring protocol for service discovery in MANET by reducing the flooding and exchange of service discovery messages. The modification conforms to the general specification of the Service Ring protocol.

6. ACKNOWLEDGEMENT

We wish to thank Dr. Raghav Yadav for his support and valuable feedback on enhancement approach for Service Ring Protocol in Service Discovery Process. Also thankful to Dr. W. Jeberson for his feedback and to other anonymous reviewers.

7. REFERENCES

[1] A. Bakker, E. Amade, G. Ballintijn, I. Kuz, P. Verkaik, I. van der Wijk , M. van Steen, and A. S. Tanenbaum, “The globe distribution network,” in Proceedings of the USENIX Annual Conference, pp. 141–152.

[2] A. Ozg¨ur, O. L´evˆeque, and D. N. C. Tse, “Hierarchical cooperation achieves optimal capacity scaling in ad hoc networks,” IEEE Trans.Inf. Theory, vol. 53, no. 10, pp. 3549–3572, Oct. 2007.

[3] B. Y. Zhao, J. D. Kubiatowicz, and A. D. Joseph, “Tapestry: An Infrastructure for fault-tolerant wide-area location and routing,” UC Berkeley, Tech. Rep. UCB/CSD-01-1141, April 2001.

[4] D. Chakraborty et al., “Toward Distributed Service Discovery in Pervasive Computing Environments,” IEEE Trans. Mobile Computing, vol. 5, no. 2, 2006, pp. 97–112.

[6] E. Royer and C. Toh, “A review of current routing protocols for ad-hoc mobile wireless networks,” IEEE Personal Communications, April 1999.

[7] Internet Engineering Task Force, “Simple service discovery protocol - internet draft v1.03,” http://www.upnp.org/download/draft cai ssdp v1 03.txt.

[8] I. Stoica, R. Morris, D. Karger, M. F. Kaashoek, and H. Balakrishnan, “Chord: A scalable peer-to-peer lookup service for internet applications,” in *Proceedings of the 2001 conference on applications, technologies,architectures, and protocols for computer communications*. ACM Press, 2001, pp. 149–160.

[9] J. Tyan and Q. Mahmoud, “A Network Layer Based Architecture for Service Discovery in Mobile Ad Hoc Networks,” Proc. 17th Ann. IEEE Canadian Conf. Electrical and Computer Eng. (CCECE 04), IEEE Press, vol. 3, May 2004, pp.1379–1384.

[10] L. Cheng and I. Marsic. Service discovery and invocation for mobile ad hoc networkedappliances, December 2000

[11] M. Klein, B. Konig-Ries, and P. Obreiter, “Service Rings: A Semantic Overlay for Service Discovery in Ad Hoc Networks,” Proc. 14th Int’l Conf. Database and Expert Systems Applications (DEXA 03), IEEE CS Press, 2003, pp. 180–185.

[12]M. Paolucci, T. Kawmura, T. Payne, and K. Sycara, “Semantic matching of web services capabilities,” in First International Semantic Web Conference, 2002.

[13]Network Working Group, “Service location protocol - RFC 2165,” June 1997, http://www.ietf.org/rfc/rfc2165.txt.

[14]O. Ratsimor et al., “Allia: Alliance-Based Service Discovery for Ad Hoc Environments,” Proc. 2nd Int’l Workshop Mobile Commerce (WMC 02), ACM Press, 2002, pp. 1–9.

[15] U. Kozat and L. Tassiulas, “Network Layer Support for Service Discovery in Mobile Ad Hoc Networks,” Proc. IEEE Conf. Computer Comm. (INFOCOM 03), IEEE Press, 2003, pp. 1965–1975.