2010 10th IEEE International Conference on Computer and Information Technology (CIT 2010)

Reliable Hybrid Service Discovery Protocol in Mobile Ad-Hoc Grid

Askari Parichehreh
Islamic Azad University - Behshahr Branch
apmastermail@gmail.com

Bahman Javadi
Islamic Azad University– Behshahr Branch
javadi@aut.ac.ir

Mehdi Sarikhani
Islamic Azad University- Behshahr Branch
Sarikhan.mehdi@gmail.com

Fereshteh Momeni
Islamic Azad University- Behshahr Branch
momeni@iaubs.ac.ir

Abstract- **Mobile ad-hoc Grid is one of the modern distributed systems that have been introduced until now. It combines traditional grid capabilities with ad-hoc network flexibilities. Service discovery is one of important challenges in this environment. In this paper, we focused on hybrid service discovery protocol and proposed a reliable service discovery scheme based on neighbors communication link quality in each zone. Service query messages in our proposed algorithm are directed to neighbor nodes selectively along reliable links. Reliable link are distinguished by power of advertisement messages signal which are send between nodes in a zone. We show by mathematical proof and simulation that compared to traditional hybrid protocol, our reliable hybrid protocol achieves fewer messages overhead and better success service delivery ratio.**

Keywords—mobile ad-hoc grid; service discovery; hybrid protocol;

I. INTRODUCTION

Mobile ad-hoc grid is a computational environment that use parallel and distributed technology. Computing nodes in this system is established on mobile ad-hoc network, thus this technology combines capability of traditional grid with flexibility of mobile ad-hoc networks. However, the development of ad-hoc grids entails new challenges compared to traditional wired grids. Resource discovery, power consumption, QoS security and etc, are problems that have still to be resolved. In this paper we present in-depth the problem of resource discovery in mobile ad-hoc grids. Generally, we can classify service discovery mechanism in tree main categories:

1. Advertisement based service discovery
2. Query based service discovery
3. Hybrid service discovery

In the advertisement based mechanism or proactive model, servers advertise periodically, by broadcasting or multicasting location and attributes of resources along with services. So that the client can build a local database with all services which therefore available on the grid environment. Examples of P2P Advertisement mechanisms are the Konark passive push protocol [1], and the UPnP discovery service [2].

In query based mechanism or reactive model, client sends a query message to other nodes, by Broadcasting or multicasting and asks about services that match the same specific attributes. Any servers which having that service respond to the client's query by sending a description of the service, Examples of P2P query-based systems are the Bluetooth SDP[3], the Konark active pull protocol [1], and the service discovery mechanism proposal for on-demand ad-hoc networks[4,5].

Hybrid mechanism tries to find a good compromise between proactive and reactive methods. The basic idea of hybrid mechanism is to limit the advertisement message within small zone; a zone is defined for each node individually to reduce advertisement overhead and on the other hand enhance query response time. Example of hybrid service discovery protocol can be observed in [6].

Using the larger zone reduces response time but increase network traffic overhead. We found a solution to decrease the response time without increasing network traffic.

The remainder of the paper is organized as follows. First of all, we describe a background of our work and make a survey on service discovery protocols and in section II. We present our service discovery mechanism in section III, while mathematical proof and performance evaluation are presented in section IV and V respectively. Finally, section VI concludes this paper.

II. BACKQROUND AND RELATED WORK

A considerable amount of work has been spent into studying and designing resource discovery protocol in ad-hoc grid environments and solving of this issue. In this section, we present a background of our work and describe four important service discovery protocols which are implemented in real situations.

For evaluating a service discovery mechanism we have two basic parameters: (i) the clients are enabled to discover several servers with proper services. (ii) The servers which elected for job execution are enabled to deliver the requested service.

For reliable service discovery, we should focus on using the reliable links between neighbor nodes. Some of links between nodes may not be reliable because their connections maybe week and disconnected by nodes mobility. Thus, we should monitor movement and geographical position of nodes to capture their movement tendency. There are several ways to solve this problem, but each has some difficulties. One way to obtain node position is using GPS. But GPS is an extra hardware that should be set up on mobile nodes and suffers from poor signals in indoor environment. Another way is by measuring relative locations from fix-point position-known based stations, but this decentralized architecture is not suitable for ad-hoc grid dynamic transient environment. Due to this reason we try to find another solution for this problem that does not have any difficulties, some work in ad-hoc network has been carried out about this problem. But in ad-hoc grid environment and in hybrid protocol, we do not find any solution, so this matter motivates us for this research. In the following, we describe some important service discovery protocols.

A. Service location protocol (SLP)

This protocol was introduced by the IETF SVRLoc Working Group and initiated in 1997. SLP version 2 is the last version of SLP and was standardized in 1999. [13, 14]

SLP is a lightweight, open and scaleable protocol for service discovery in IP networks. It can use several architectures such as centralized or decentralized directory and even directory-less architecture. SLP also use tree agents: (i) user agents (US) that perform service discovery

978-0-7695-4108-2/10 $26.00 © 2010 IEEE
DOI 10.1109/CIT.2010.477

on behalf of clients or applications, (ii) service agents (SA), which advertise the service information and position, (iii) directory agents (DA) that service information are registered on it. The SLP uses the centralized architecture when DA is present, and otherwise uses the directory-less architecture.

B. Jini

Jini was introduced by Sun Microsystems in 1998. The Jini Community was initially established in January 1999 with the release of the first version of the Jini Technology Starter Kit from Sun Microsystems. The last version of the Jini Technology Starter Kit has been released in February 2004 [15, 16].

Like SLP, Jini can provide several architectures. Jini uses Lookup Service (LUS) as directory agent which maintains a database for all services in the network. When LUS is found the services will register with it and clients can query it using Remote Method Invocation (RMI). Then, LUS matches the query against the registered services information and provide response to the client. The client may receive several responses from several LUS; therefore the client may run filtering process in order to make a selection over the received messages.

C. Konark

Konark is a peer to peer service discovery mechanism, in which each node that provides services runs a SDP manager, SDP managers in each mobile device used for two basic functions: (i) registering and advertising local services and (ii) service discovery on behalf of the client's application.

To discover the service the Konark uses two different models: Active pull and passive push model. In active pull model, client multicasts a service discovery message and then the servers with the required service respond to the message. In passive push model, servers advertise periodically the services that are locally registered.

D. Hybrid service discovery protocol.

This protocol is similar to idea that introduced in Zone Routing Protocol [10-12]. A discovery zone is defined for any node individually, and is composed of the entire neighbors whose distance to them dose not exceeds a certain number of hops, that we named in R where R is the zone radius.

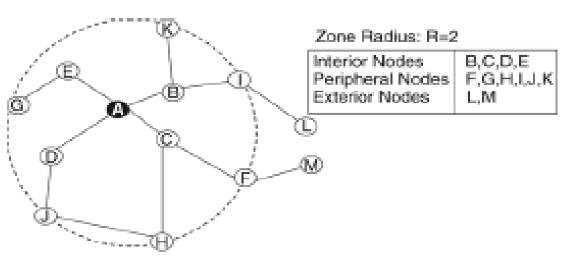


Figure 1. Classification of deferent nodes in a zone with R=2.

According to Fig 1 within the discovery zone of a given node, nodes can be classified into tree classes:

1- Interior nodes which its distance to given node is less than R.
2- Peripheral nodes which its distance to given node is equal to R.
3- Exterior nodes which its distance to given node is more than R

Service Discovery mechanism in this protocol uses hybrid peer to peer mechanism. For service discovery within the zone, we used advertisement mechanism and for service discovery between the zones the query mechanism is used. When the client node needs a special service, it broadcast or multicast a query message to other nodes and asks about services that are matched with the same specific attributes. Any servers having the special service, respond to the client's query by sending a description of the service [7]. Client may receive several replies from many servers, in this case client node can filter reply messages or use first arrival message to decrease response time.

III. Reliable hybrid service discovery protocol

At first, it should be mentioned that computational nodes in ad-hoc grid, links together with wireless connections. Any node in this environment sends messages with certain transmission power. Subject to distance between nodes, a signal is received at the receiver with certain power and quality. On the other hand, each mobile node based on its hardware and software configuration, has a receiving sensitivity. Receiving sensitivity is the power of the weakest signal that can reliably hear and demodulat. This is notable that the power of message has inverse relation with nodes distance. This means that signal powers of closer nodes is more than farther nodes. Thus, it can be said the receiving power by the receiving node is inversely proportional to square of the distance between the nodes.

$$\gamma = \frac{P_t}{4\pi(r^2)}$$

Therefore, when distance between neighbor nodes in the zone of hybrid service discovery protocol increase, reliability of links decrease and also probability of link broken increase.

Therefore, we use reliable links based on the power of signal. In our proposed protocol, we define reliable links as links that have tree criteria as follows:

i) The power of signal in both ends of reliable link has a value above receiver sensitivity threshold.
ii)The sequence of signal power in a sequence of time does not have decreasing tendency.
iii)The link should be live; this means that, advertisement and NDM messages have been changed between nodes.

A node with above link connection properties is called strongly connected node. But how can client distinguish reliable links and strongly connected nodes in its neighborhood to send its query messages in a reliable way?

Our reliable service discovery protocol performs hybrid approach that combines advertisement and query mechanism like that was mentioned in reference [8]. But we have to modify that protocol to achieve our reliable approach.

A. advetirment mechanism

In our approach similar to traditional hybrid scheme advertisement messages with service information (such as processor speed (CPU), the amount of free memory, and life time of battery) send in advertisement period in each zone, when a host receives an advertisement message, extract it and then register its information. But in opposite with traditional protocol, addition of service information, the receiver registers the link quality information like, messages signal power and timestamp, this process perform with *UpdateSlt(pkt, sig)* function as is shown in Fig 2. . The all of this information stores in service look-up table as one entry. This process should be taken in both end of link. Thus, both nodes can obtain the link quality from other side. After this, host node constructs an advertisement message with updated information and sends it to other nodes in its zone by *SendAdvertisePacketViaMac* function. For avoiding flooding advertisement messages, *Hop-count* set with *Zone-Radius*.

By advertisement mechanism, all nodes in each zone know about nodes services and link quality in their zone.

```
/* input adv packet pkt , signal info sig */
extraxt(pkt);
if(pkt != null){
   if(pkt->header == adv){
    updateSLT(pkt , sig);
    AdvertisePacket advpkt;
    advpkt = createadvPacket(SLT);
    advpkt -> Hop-count= ZoneRadius;
    advpkt -> destadd = anydest;
    SendAdvertisePacketViaMac(advpkt);
    }
 }
return;
}
```

Figure 2. advertisment mechanism pesudo code

N-ID	CPU	MEM	BAT	S-Power[1..n]	R-Time[1..n]

Figure 3. service look-up table structure

The structure of service look-up table can be seen in Fig 3. As is shown, each entry in this table records information about two subjects, first, service information of sender node and second link quality information. We should mention that *S-power* and *R-time* implement as an array and each entry of this array indicates to link quality information of a sender node. Instead of *S-power* and *R-time* array, we can use *Signal Power* field that set with the following equation:

$$SS_{comulative} = \alpha \cdot SS_{comulative} + (1-\alpha) SignalPower$$

In this equation *SScomulative* is the stored value in service lookup table while *SignalPower* is the signal strength of received messages. α is the constant value that is application specific. This method reduces advertisement message size.

B. Query mechanism

When a client needs a special service such as CPU, memory or special software which does not have that service, bordercast service query message to its own reliable border nodes through reliable links. Thus, bordercasting mechanism should be initiated based on reliable links.

The bordercasting mechanism forwards a query message from current node to its border nodes. But the entire border nodes are not as a reliable, so this process in our algorithm is modified and sends query messages only to reliable nodes through high quality communication links. The reliable border nodes can be recognized easily by using of *S-power* and *R-Time* filed in service look-up table. thus, it matter of course that a large amount of bandwidth and energy that is invested for these messages will be stored by this scheme as is shown in Fig 4. In this figure reliable links is shown by bolder lines. With a fingertip account, in Fig 4 we can see, the number of messages with consideration of reliable link is less than the traditional protocol.

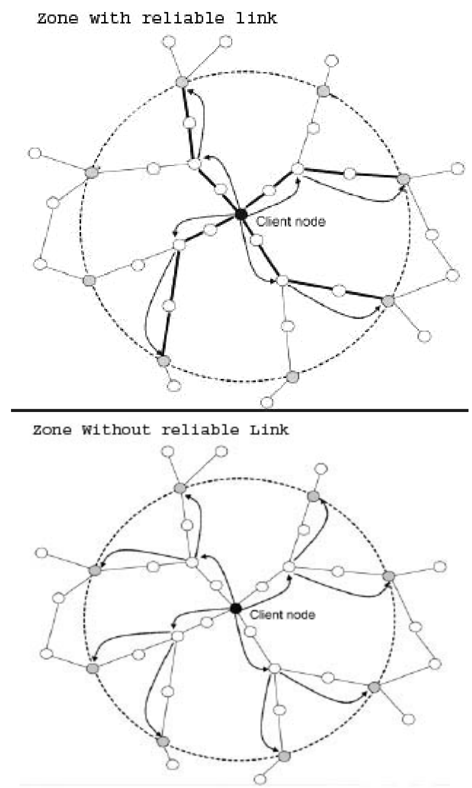


Figure 4. Bordercasting in reliable protocol versus traditional protocol

In the following, we explain query mechanism with more details and step by step.

A query message in which we used for query mechanism includes parameters such as:

- client node Identifier
- query message Identifier
- forwarding distance value
- List of required service such as processor and memory.
- A list of intermediate node which receives query messages and forwards it to border nodes, This list shows the traveling route of query message.

You can see query message structure in Fig 5.

N-ID	Seq	CPU	MEM	BAT	Forward-value	Route-List

Figure 5. Query message structure

```
/* input packet pkt  ,  signal info sig   */
extraxt(pkt);
if (pkt != null){
    if (pkt->header == Query){
        updateSLT(pkt,sig);
        if (Hop-count<=Initial-value || isstale()){
           if (ServiceExistatMe || ServiceExistatZone){
              int replyroute[];
              replypacket repkt;
              replyroute = reverse(pkt->srcadd,repkt);
              reppkt=CreateReply(pkt->srcAdd, repkt);
              ReplyViaMac(repkt);
              }
              else {
              BordercastThree BThree;
              BThree=ConstructBcastThree(BThree,pkt);
              AppendBThreeToQuery(BThee,pkt);
              BordercastViaMac(pkt);
           }
        if (Hop-count > InitialValue){
            SendTimeoutError(pkt->srcadd);
            Discardpkt(pkt);
          }
        }
    return;
 }
```

Figure 6. Query mechanism pseudo code

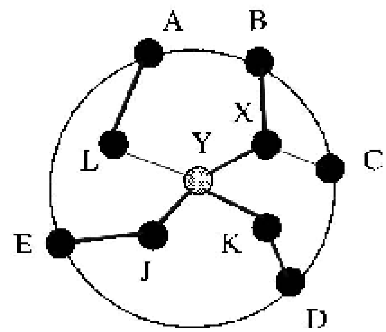
As is shown in Fig 6, when a node receives a query message:

i) register it's signal power by it's own service look-up table with *UpdateSLT(pkt, sig)* function.
ii) The hop count of the message should be checked and then, based on query control mechanism verify the query identifier in its local cache, if this message is replicated and handled previously, discard this query message, this process perform with *Isstale()* function .
iii) If the message is not replicated, extract its information and after registering its signal power, looks up the service look-up table and verify that required service does exist in its own zone or not, if so, create a response message and send it to client node. This message follow the route that query message traveled it, but in reverse mode, to get in the client node. This process perform with *reverse(),* and *CreateReply()* functions, respectively.
iv) After checking service look-up table if it revealed that service does not exist in the zone, bordercasting process will be started, but this process does not forward query message to all border bodes. It sends query message only to that nodes which have above average link quality.

C. *Bordercasting tree constraction*

In original protocol, bordercasting mechanism is used for directing query message outward, via multicast, to a set of border nodes, this operation requires construction of a bordercast tree. To achieve our goal, we modify this tree, and construct it based on quality of communication link reliability.

As it was explained previously, each node in advertisement period sends its own service information and link quality of its neighbors. Therefore client node has sufficient topological information to about intermediate and border node in order to construct bordercast tree. Therefore makes its bordercast tree and prunes some link which does not have high link quality. As is shown in Fig 7, bordercast tree in our protocol does not have information about unreliable links and their nodes such as C, L and A, however, this approach, increase query messages length and increase bandwidth consumption, but as we have shown in the following, it reduces the number of query messages, saves energy that is invested for these messages sending and also increases the success ratio of service delivery.



Query message	
Bordercast tree	
Y	X
Y	K
Y	J
X	B
K	D
J	E

Figure 7. Bordercast tree construction

D. *Service selection and filtering mechanism*

In traditional grid after service discovery and prior to job scheduling, filtering mechanism, filters the discovered services. Thus, we should implement a similar strategy in ad-hoc grid environment.

After query mechanism several nodes may provide required services and reply to the client with response message, for sake of filtering this messages and selecting better server, we can consider several parameters like geographical approximation of the server which can be obtained with hop-count value of response message. But we should point out that the most important issue in ad-hoc grid is disconnection operation and link broken probability, furthermore, the response time is very critical for user. Thus, we overlook the geographical distance and take the first response message as selected service and assign job to that server. We hope that server has the nearest geographical station, as compared to other servers.

This method requires no special tools or equipment for geography station approximation, and thus is easy to implement.

IV. MATHEMATICAL PROOF

We assume that the connectivity of two nodes is a memory less stochastic process for simplicity. Meaning that, a future connectivity of two nodes is independent of its history, but, depending only on the current state of the connectivity. In Fig 8, we describe the connectivity model between two nodes in ad-hoc grid.

We define markov chain contains with two state S={R, U}. Where the state R shows that the wireless link is reliable, but U state shows unreliability of wireless link connectivity. Transition from R to U has probability λ, and transition from U to R has probability μ.

As is shown in Fig 5, we have a continuous-time 2-state markov chain of process Xt with probability of μ, λ as follow:

$$P\{X(t+h)=R \mid X(t)=R\}=(1-\lambda)h \qquad (1)$$

$$P\{X(t+h)=R \mid X(t)=U\}=\mu\cdot\lambda \qquad (2)$$

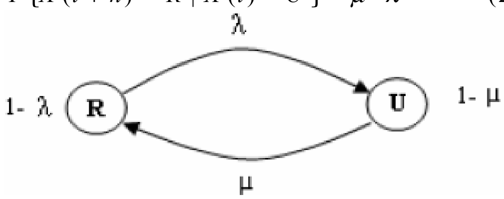


Figure 8. Connectivity states and their transition probability

The probability of $P_R(t)$ and $P_U(t)$ can be obtained as follow:

$$P_R(t+h)=P_R(t)P\{X(t+h)=R \mid X(t)=R\} + P_U(t)P\{X(t+h)=R \mid X(t)=U\} \qquad (3)$$

$$P_R(t+h)=P_R(t)\cdot(1-\lambda)h+P_U(t)\cdot\mu\lambda \qquad (4)$$

and,

$$\lim_{h\to 0}\frac{P_R(t+h)-P_R(t)}{h}=-\lambda P_R(t)+\mu P_U(t) \qquad (5)$$

In stationary state that P(t) is independent from time and $P'_R(t)=0$, we can write :

$$-\lambda\cdot P_R+\mu\cdot P_U=0 \;\Rightarrow\; P_R=\frac{\mu}{\lambda}\cdot P_U$$

In other side, we know that, $P_R+P_U=1$ then

$$P_R=\frac{(\frac{\mu}{\lambda})}{\left(1-\frac{\mu}{\lambda}\right)}$$

So that,

$$P_R=\frac{\mu}{\mu+\lambda} \text{ and, } P_U=\frac{\lambda}{\mu+\lambda} \qquad (6)$$

Therefoe we can say that the wireless link can be in reliable state with probability of P_R.

Let X is the number nodes in R state. It is clear that, X is an Binomial random variable with parameters N,P_R .

Therefore, if C as a client node has K reliable border nodes and N border nodes, the probability of k nodes reliability is equal to:

$$P(X=K)=\binom{N}{K}\left(\frac{\mu}{\mu+\lambda}\right)^K\left(\frac{\lambda}{\mu+\lambda}\right)^{N-K} \qquad (7)$$

$$K = 1,2,3,\ldots,N$$

Thus, the average number of query messages that in query mechanism for each intermediate node should send is equal to number of reliable neighbor nodes, so we have:

$$E(QueryPackets)=E(X)=N\cdot\left(\frac{\mu}{\mu+\lambda}\right) \qquad (8)$$

But, if we do not use reliable service discovery mechanism like traditional protocol, the number of query message that is used for bordercasting mechanism in intermediate node like C is equal to N. because node C sends the query messages to all reliable or unreliable nodes in its own zone. With compression of number of query messages in our protocol and traditional protocol it can be easily observed that the average number of messages in our protocol is less than traditional protocol, because the below relation always is true.

$$0 \prec \frac{\mu}{\mu+\lambda} \prec 1$$

V. PERFORMANCE EVALUATIONS

We evaluate our reliable hybrid service discovery protocol in Glomosim simulator [17]. We set number of nodes between 500 to 1000, and we used IEEE 802.11 b for MAC protocol that operates at a 2 Mbite/s data rate. The speed of node is set between 2m/s to 10 m/s. We defined 50 different services that any node can provide, 10 service in built-in form, and when it need other services it should send a query message to other nodes. We use several parameters in our simulation that are shown in table1.

TABLE 1 SIMULATION PARAMETERS

parameter	symbol	value
Number of nodes	N	500 - 1000
Network coverage area	A	1500m × 1500m
Node speed	V	2 - 10 m/s
Node mobility	M	Random waypoint
Zone	R	1-10
Data transfer rate	T	2Mbit/s

It is intuitive that in ad-hoc grid environment, if service discovery request are rather rare, a zone size can be smaller, this approach would be efficient at least in term of control overhead, although increase response time. But, where request for services is frequently high, a zone size can be increased. So, setting of zone radius is application specific parameter. We use several values between 1 to 10 for it in some scenarios and use fix value in other scenarios.

In our simulations each node moves according to the random waypoint mobility model. The total time of simulation is 500 second. Note that in a real situation, link quality is controlled by the signal sensitivity threshold that set through the device driver interface. But we assume that the link quality depends on the distance between the sending and receiving nodes. In our simulation, we set radius of nodes coverage area to 120m, if two nodes have distance more than 100 m, the link between them is considered as unreliable. And a link with more length of 120 m is considered as disconnected. We considered two parameters for comparison, success delivery ratio and query message traffic. We did not consider advertisement messages, because it was the same in two protocols.

At first, we evaluate the ratio of success service delivery in our protocol compared with traditional protocol, as is shown in Fig 9, the number of success service delivery in our algorithm is more than traditional algorithm, and we can say that this difference is caused by accuracy of service selection in our algorithm, because we use reliable link for service discovery and delivery but in traditional scheme link quality is not considered. Furthermore, the rate of service failure in our reliable service discovery protocol is less than the traditional hybrid protocol.

Then, we evaluated the relation of success service delivery ratio with node mobility. For this, we performed several scenarios, with different node mobility speed. We consider zone radius as a fix value(R=4). Simulation showed that success ratio of service delivery decreased when mobility increased. But as shown in Fig10, reduction gradient in our protocol is less than original protocol because we use reliable link where broken probability in them is low.

Another parameter that we considered in our simulation was number of query messages that bordercast to border nodes. We increased number of nodes from 500 to 1000 in our six time simulations. As is shown in Fig 11, number of query messages that bordercast to the border nodes is less than the number of query messages that is bordercast in original protocol.

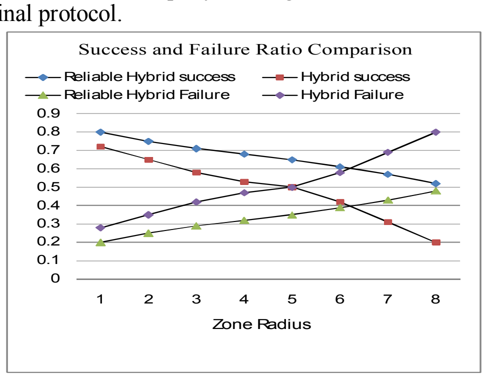


Figure 9. Success and failure ratio

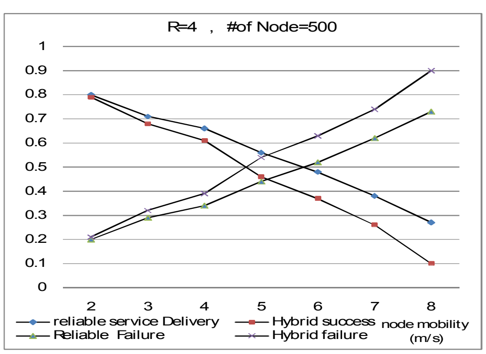


Figure 10. Service delivery versus mobility

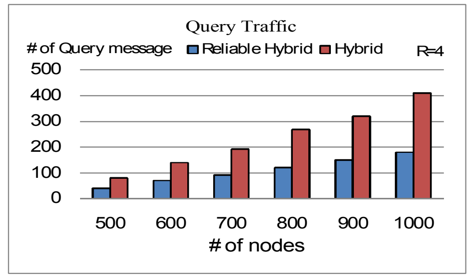


Figure 11. Query traffic comparison

VI. Conclusion

In this paper, we modified hybrid service discovery protocol and improve its reliability based on power of signal that exists on advertisement messages which sends in a zone. We modified the bordercasting mechanism and recognize reliable links and its strongly connected nodes in each zone, then forwarded the query messages only through these reliable links. We proofed mathematically that our reliable hybrid service discovery protocol has less traffic and bandwidth consumption, and showed success ratio of service delivery in our protocol in comparison with traditional protocol by simulation.

REFRENCE

[1] S. Helal, N. Desai, V. Verma, C. Lee, "Konark — A Service Discovery and Delivery Protocol for Ad-hoc Networks," the Third IEEE Conference on Wireless Communication Networks (WCNC), New Orleans, March 2003.

[2] B.A. Miller, T. Nixon, C. Tai" Home networking with universal plug and play," IEEE Communications Magazine 39 .2001

[3] Bluetooth. Service discovery protocol, Bluetooth Specification Version 1.1, Part E, Feb. 2001.

[4] R. Koodli, C. Perkins, "Service discovery in on-demand ad hoc networks," IETF Internet Draft (draft-koodli-manet servicediscovery- 00.txt), October 2002.

[5] C. Frank,"Consistency challenges of service discovery in mobile ad hoc networks," the 7th ACM Int. symposium on modelling, analysis and simulation of wireless and mobile systems, 2004

[6] R. Moreno, "A hybrid mechanism for resource/service discovery in ad-hoc grids," Dept. de Arquitectura de Computadores, Universidad Complutense de Madrid, Madrid, Spain. 2008

[7] T. Phan, L. Huang, C. Dulan. "Challenge: Integrating Mobile Wireless Devices into the Computational Grid,".2003, pp. 72–8

[8] A. Parichehreh, B. Javadi, A. T. Haghighat, "A Modify on Hybrid Service Discovery Protocol in Mobile Ad-Hoc Grid," ICFN 2010

[9] Z. Li,"Challenges Of Mobile Ad-Hoc Grids and Their Applications," 2005.

[10] Z. J. Hass, "A new routing protocol for the reconfigurable wireles s networks," Proceedings, IEEE 6th International Conference on Universal Personal Communications, 1997, pp. 562-566.

[11] Z. J. Hass and M. R. Pearlman, "The performance of a new routing protocol for the reconfigurable wireless networks," Proceedings, IEEE International Conference on Communications (ICC), 1998,

[12] M. R. Pearlman and Z. J. Hass, "Determining the optimal configuration for the zone routing protocol," IEEE Journal on Selected Areas in Communications, vol. 17, no. 8, , Aug. 1999.

[13] E. Guttman, C. Perkins, J. Veizades, and M. Day, "RFC 2608: Service location protocol, version 2," 1999, status: PROPOSED STANDARD.

[14] SRVLOC, "Service Location Protocol Working Group," 1997.

[15] Sun Microsystems, "Device Architecture Specification," Sun Microsystem, Inc, Tech. Rep., June 2003.

[16] Sun MicroSystems, "The Davis Project," 2003. [Online]. Available: http://davis.j

[17] Glomosim simulator homepage, http:// pl.cs.ucla.edu/projects /glomosim