

AVERT: Adaptive SerVice and Route Discovery ProTocol for MANETs

Christopher N. Ververidis and George C. Polyzos, *Member, IEEE*

Abstract—In the past few years there has been increasing research interest in service discovery protocols for Mobile Ad Hoc Networks (MANET). The most promising approaches so far address energy efficiency by merging the service discovery process with the routing process. Actually the service information is piggybacked into routing messages, so that a node is simultaneously informed of available services and of routes towards the corresponding service providers. This cross layering leads to improved adaptation to the network conditions and at the same time to significant energy savings. Those savings are infeasible if the two processes are implemented separately, because then each one would have to use its own messages and create additional (if not) redundant network overhead. In this paper we propose such a hybrid adaptive protocol for energy consumption-averse users. This protocol, named AVERT, is based on the Independent Zone Routing framework (IZR) and has the ability to allow each node to adapt its zone range (similarly to the way it is done in IZR). We also add a mechanism to adapt the sending rate of proactive messages on each node based only on local traffic monitoring. Through simulations we show that using this mechanism, the energy efficiency achieved is substantially higher compared to similar hybrid service and route discovery protocols.

Index Terms— Cross-Layer, Energy Efficiency, Mobile Ad Hoc Networks, Service Discovery and Advertisement,

I. INTRODUCTION

MOBILE Ad Hoc Networks (MANETs) have attracted significant research efforts of the networking community in the past. However, MANETs have not experienced the forecasted popularity in terms of mass usage and applications. A possible reason for that is that the main bulk of research aimed at solving the severe problems at the lower levels of the protocol stack (network layer, link layer) that such networks experience. However, since MANETs were envisioned to allow mobile users to access services and data without relying on any infrastructure, solving connectivity problems was not enough; it became crucial to also develop protocols for discovering data and services inside MANETs.

C. N. Ververidis is with the Computer Science Department, Athens University of Economics and Business, Athens, 10434 GREECE (phone: +30-210-8203646; fax: +30-210-8203860; e-mail: chris@aub.gr).

G. C. Polyzos is with the Computer Science Department, Athens University of Economics and Business, Athens, 10434 GREECE (e-mail: polyzos@aub.gr).

In the past few years there has been growing interest in developing energy efficient service discovery protocols. The most efficient of those protocols were proved to be cross layer protocols. These protocols are based on integrating the routing process with the service discovery process. The motivation for integrating routing and service discovery is to reduce the redundancy experienced when each of those two processes uses its own messages. Cross layer service discovery exploits the capability of simultaneously acquiring service information along with routing information by piggybacking service information onto routing messages. This way, redundant transmissions of service discovery packets at the application layer are avoided and energy is saved.

The idea of providing routing layer support for service discovery was first introduced by Koodli and Perkins in [1] who extended a reactive routing protocol, namely Ad Hoc On Demand distance Vector protocol (AODV). In [2] the authors have extended the Ad hoc On-Demand Distance Vector (AODV) routing protocol with service discovery functionality and have experimentally compared it with NOM [3] (a pure application based service discovery protocol). Their findings show that the integrated protocol produces 30% to 50% less control overhead and has 2 to 7 times lower service acquisition latency than the application layer based protocol (depending on simulation parameters). In [4] in order to compare a *reactive* routing and service discovery protocol and a *proactive* routing and service discovery protocol, DSR and the Destination-Sequenced Distance Vector protocol (DSDV) were extended to provide service discovery functionality. Those approaches were compared against SLP, implemented at the application layer. The extended DSR protocol proves to have the least messaging overhead among the three, with second best the extended DSDV protocol. DSDV is not the only proactive routing protocol extended with service discovery functionality. In [5] and [6] researchers have also extended the Optimized Link State Routing (OLSR) proactive routing protocol to support service discovery. Comparing the proactive, reactive and hybrid integrated protocols one can see that the most energy efficient and also effective protocols are the hybrid ones [7]. Hence, before we proceed in describing AVERT, which is a hybrid service and route discovery protocol, we will briefly present existing hybrid integrated protocols.

In our previous work [8] we have added service discovery extensions in a *hybrid routing protocol*, and we explicitly measured energy consumption. In hybrid routing protocols

each node proactively advertises the routes and services it is aware of by periodically sending control messages to its neighbors up to a fixed number of hops away (this is called the node's zone). Information for routes or services outside this zone may be gathered only upon request (reactively). Experimental results show that our protocol, namely E-ZRP, clearly outperforms application layer based service discovery protocols demonstrating energy savings of up to 55%. The proposed hybrid integrated protocol in [9] resembles the one proposed in [8] but adds the functionality of dynamically adjusting the size of a node's zone depending on service usage frequency. The higher the popularity of a node's services, the larger the zone where proactive announcements should be propagated. In [10] the same concept is followed, with the difference that the zone size determination is based on the transmission power/range selected by a node. Finally [11] presents another hybrid integrated protocol (SPIZ), where an autonomous and adaptive zone radius determination mechanism (based on multiple criteria such as call rate, mobility, service popularity etc.) is provided. In this paper we propose AVERT, a hybrid service and route discovery protocol that differs from all previously mentioned protocols in that it not only allows adaptation of zone radius but also adaptation of the rate of proactive messages sent by nodes, based only on local traffic monitoring on each node. Also we evaluate AVERT in terms of energy consumption, an analysis not found in the aforementioned approaches (except [8]), which only focus on packet overhead as an implicit indication of energy consumption.

II. THE DESIGN OF AVERT

In this section we propose AVERT, a hybrid service and route discovery protocol that differs from other service discovery protocols based on ZRP, in that it not only allows adaptation of zone radius but also adaptation of the rate of proactive messages sent by nodes, based only on local traffic monitoring on each node. In the following paragraphs we present our motivation for creating AVERT and also evaluate AVERT in terms of energy consumption, contrasting it to similar service discovery protocols.

Hybrid routing protocols have been proven to operate more efficiently than proactive or reactive protocols in MANETs, the main reason being their flexibility to adapt to changing network conditions. This was the case for ZRP, which uses a proactive protocol (IARP - Intra Zone Routing Protocol) for local routes and a reactive protocol (IERP - Inter Zone Routing Protocol) for global routes. Actually the proactive protocol serves as a basis for the global reactive protocol to discover distant routes more effectively. In ZRP, a node's zone is defined as the distance in number of hops to which the proactive route advertisement packets are allowed to propagate. Those advertisement packets are used by every receiving node to build its IARP routing table, which stores entries for all the available routes inside the node's zone.

Now, in the case that a node requests a route towards a destination that is not located inside its zone, then the requesting node initiates the IERP protocol. The IERP protocol uses a mechanism called bordercasting in order to find an available route towards the queried destination. With bordercasting the query is not flooded to all the neighbors of the originating node but it is sent only to those nodes that located on the border of the requesting node's zone (called the node's border nodes). Those border nodes check their IARP tables to see if they host any route towards the queried destination, and if they do they unicasts an IERP reply message to the requesting node informing it of the available path. If the border nodes do not find any related entry they just rebroadcast the query to their own border nodes.

In order for ZRP (respectively E-ZRP) to operate efficiently in any MANET, it is implied that all nodes fix their zone radius to the best value, assuming that they know a priori the call rate and the mobility rate in the MANET¹. Moreover, all nodes following ZRP have to use the same zone radius, which is problematic when different areas in the MANET present different call to mobility ratios. In order to remove these assumptions the authors in [12] have proposed IZR. IZR is a sophisticated version of ZRP, which allows every node to have a different zone radius and also to dynamically tune it on-the-fly by monitoring local traffic. In the next paragraphs we briefly describe the operation of IZR and the extensions made to it in order to build the AVERT route and service discovery protocol.

The basic difference of IZR compared to ZRP is its mechanism for adapting a node's zone radius to changing conditions in the MANET. The traffic produced by either IERP or IARP is largely dependent on the zone radius. The larger the zone the more IARP traffic is created (for updating a larger set of nodes) and the less IERP traffic is needed, since more destinations are inside the local zone and there are less queries for out-of-zone nodes. Actually in [13] it is experimentally shown that the total traffic (IERP and IARP) is a convex function of the zone radius. Taking this into account two distributed zone configuration algorithms are used by IZR.

The first algorithm, called Min-searching, is utilized for finding the zone radius, which corresponds to a local traffic minimum (which is also a global minimum due to convexity). Periodically each node measures the amount of routing traffic (IERP and IARP) that passes through it and chooses to increase or decrease the zone radius. If the decision in a previous measurement period was to increase (respectively decrease) the zone radius, and the amount of routing traffic measured in the next period has decreased then the node further increases (respectively decreases) the zone radius. If the traffic has increased compared to the previous period, then the zone radius is changed in the inverse direction of the one that was followed in the previous period. The algorithm stops

¹ It has been shown that small values for the zone radius give better performance when the call to mobility ratio is small and vice versa.

when the traffic of the previous period is less than the traffic of the period before that and also less than the current period. This is considered to be the best zone radius for achieving a short-term minimum routing traffic overhead.

However, this minimum corresponds to current network conditions and should be continuously adapted. This adaptation is done using the Adaptive Traffic Estimation (ATE) algorithm described in the next paragraph. Having reached a temporary minimum, ATE takes control and tries to adapt the zone radius by increasing or decreasing it in order to match the changing network conditions. Having as a reference the ratio IERP traffic/IARP traffic corresponding to the minimum discovered by Min-searching, it periodically measures the current traffic ratio and if it is increased by more than a factor of H , then it chooses to increase the zone radius by one hop. An increased ratio means that the IERP traffic dominates the routing traffic and hence the zone radius is smaller than it should be, given the current network conditions, and must be increased. Increasing the zone radius would lead to more efficient bordercasting and less IERP traffic. In the opposite case, where the current ratio is measured to be less by a factor of more than H than the reference ratio, then ATE decides to decrease the zone radius. A decreased traffic ratio means that the IARP traffic is now dominating and hence the zone radius is bigger than it should be. In case that a very large change is detected in the current IERP/IARP traffic ratio, then the Min Searching mechanism is re-initiated in order to find the new optimal zone radius (and the respective IERP/IARP traffic ratio). The re-invocation of the Min Searching mechanism can also be done periodically.

Experiments in [12] have shown that the IZR with dynamic zone radius configuration leads to more than 60% reduction in routing control traffic compared to the optimal setting of regular Zone Routing [13]. The basic difference between those two approaches is that in [13] all nodes have the same zone radius, which does not change, while in [12] each node may have a different zone radius and also adapt it using the two aforementioned algorithms.

The extensions made to the packets used in IZR in order to include service discovery information are essentially the same to the extensions done in the E-ZRP packets (see [8]). However AVERT builds on IZR in order to be even more efficient than E-ZRP but also introduces an adaptation mechanism for controlling the frequency of proactive traffic, in order to achieve higher energy savings when possible. To be more precise, the adaptation mechanism, called Broadcasting Frequency Optimizer (BFO), adapts the frequency with which the service aware NDP and IARP messages are broadcasted. The basic idea of BFO is that nodes that are not currently engaged in service invocation, discovery or provision (either as clients, providers or intermediates), can decrease their rate of sending proactive traffic (namely NDP and IARP packets) in order to conserve energy. BFO runs periodically on every node. In each period the node measures

only the data traffic² that passes through it and compares it to the data traffic passed through it in the previous period. If the current traffic is found to be lower than the traffic of the previous period, the node increases the time intervals between two subsequent broadcasts of IARP and NDP packets by T seconds. In the opposite case, it decreases both these intervals by T seconds. In order to avoid the two extremes of setting the broadcast interval to arbitrarily high values or to zero, a maximum allowable and a minimum allowable value for the broadcast interval are taken also into account such that they are never violated. When nodes increase their IARP broadcast interval the routing entries become stale more easily, hence nodes have to issue IERP queries more frequently in order to find routes and services. The obvious outcome is that such nodes that seem to decrease their involvement in service discovery-invocation and routing, decrease their outgoing traffic, thus saving energy to themselves and to their neighbors. Also, the increase of the IARP broadcast interval is accompanied by an increase in the NDP broadcast interval, which means that fewer changes in the 1-hop neighborhood of nodes are detected and hence the amount of expedited IARP messages can also be decreased, thus further decreasing the total proactive traffic. Now in case that the node becomes more involved in creating, receiving or relaying traffic, BFO decreases the broadcast interval of both IARP and NDP messages, such that the node informs its neighbors frequently about its state. This is done because it is crucial for itself and for the connected nodes to maintain accurate connectivity information.

We could say that AVERT uses MinSearching and ATE to decrease the total traffic in the network, and also employs BFO in order for every node to decrease its own outgoing traffic (sending packets costs more energy than receiving). Also BFO tries to do this in a harmless way for other nodes. It decreases the proactive traffic in cases that the node seems not to be too involved in sending or receiving traffic. In Figure 1 we show how the broadcast intervals for proactive traffic affect a node's outgoing traffic.

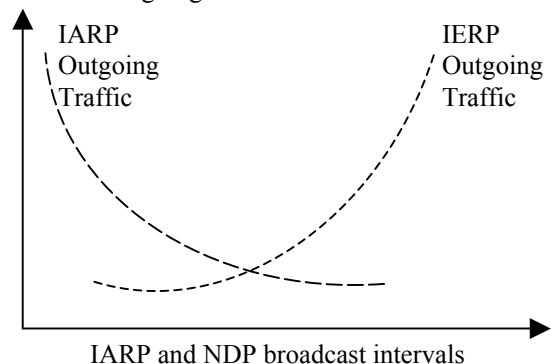


Fig. 1. Effect of IARP and NDP broadcast interval size on IARP and IERP outgoing traffic.

It is intuitive that IARP outgoing traffic is a non-increasing

function of the IARP and NDP broadcast intervals, since increasing these intervals means sending packets more sparsely. On the other hand this has the effect that IARP entries and link state get outdated more easily, hence a node may increase the usage of IERP requests for finding routes. Hence IERP traffic is a non-decreasing function of IARP and NDP broadcast intervals. Now, BFO tries to optimize the outgoing traffic by changing the broadcast interval accordingly, and it does so with the aim of not disrupting the service and routing processes that currently go through the node.

In general, we could say that BFO is not affected by MinSearching or ATE since it measures only the data traffic on every node and not the control traffic. However, it may affect those two mechanisms, since it controls the amounts of proactive traffic through the adaptation of the broadcasting intervals. In the next section we experimentally investigate the effects of coexistence of the BFO mechanism with the MinSearching and ATE mechanisms, as revealed by the service success ratios and the energy consumption achieved when using those two mechanisms with and without BFO. We also investigate the energy savings obtained from using AVERT against using IZR and SPIZ.

III. PERFORMANCE EVALUATION OF AVERT

The Qualnet [15] simulation environment was used to simulate AVERT. The initial broadcast intervals for IARP and NDP have been set to 10 seconds, which is also the lowest allowable interval, and the maximum allowable intervals have been set to 100 seconds. The MinSearching and ATE mechanisms are run every 200 seconds and the BFO mechanism runs every 100 seconds. The simulation time for every experiment is set to 10000 seconds. The initial zone range has been set to 5 hops for all nodes. In the following experiments the network consists of 20 nodes uniformly spread over an area of 2000x2000 meter². All nodes move following the Random Waypoint Mobility model (RWP) with constant speed of 3,5 m/s and no pause time. The wireless transmission range is set to 380 meters. Each server may host only one out of three possible service types (for the case of 3 servers in the network, service assignment to servers is done such that each type of service is hosted by exactly 1 server). Every 100 seconds, each client selects with probability 1/3 one out of the three available service types and tries to establish a service session with anyone of the servers that hosts the requested service type. Each service session involves the transfer of one item of 200KB size using FTP³.

The AVERT protocol is compared with the IZR and the SPIZ protocols. The primary performance metric that concerns us is the energy consumption. However, and in order

to reveal the possible costs especially of the BFO mechanism we should also take into account the completed service sessions. Defining the correct performance metric is however not trivial. As a first approach, we assume that the optimal operating point (in terms of broadcasting frequency) for all protocols is at the point where the ratio of Success Ratio (successfully delivered services) to the Total Energy expended is maximized. We have observed through simulations that this simple metric is maximized (for all protocols) when they set their broadcast timers to the maximum allowable (100 seconds). At this point however the protocols are confined to unacceptable success ratios (52%-65% of the maximum achievable, which can be much less than 100%). The problems stem from the fact that an increase by x% in the success ratio for any of the tested protocols requires a much larger than x% increase in the expended energy. This also means that using the aforementioned simple metric, a protocol that can achieve higher success ratio compared to another at a reasonable extra energy cost could be characterized as less performant. To cope with this situation a better performance metric would take into account not the success ratio and the total energy expended but the success ratio and the number of successfully delivered services per unit of energy expended. In this context, we derive the following metric (service efficiency σ) for comparing the three aforementioned protocols:

$$\sigma = \text{Percentage of Completed Services} \cdot \frac{\text{Completed Services}}{\text{Total Energy}}$$

This formula is helpful in characterizing the performance of the service discovery protocol, since it accounts simultaneously for the success ratio achieved and also accounts for the extra effort (in terms of energy consumption) required for achieving acceptable success ratios (65%-85% of the maximum achievable).

In our experiments we compare the AVERT protocol with allowable broadcast intervals of 10 seconds up to 100 seconds against IZR and SPIZ with broadcast intervals ranging from 10 to 100 seconds. We conduct three sets of experiments; in the first set we assume a scenario (Scenario-1) with high client to server ratio (5.6 to 1), in the second set we assume a scenario (Scenario-2) with medium client to server ratio (1 to 1) and in the third set we assume a scenario (Scenario-3) with low client to server ratio (0.17 to 1). We use these different scenarios in order to see the impact of the data traffic on the performance of BFO. The data traffic is higher as the ratio of clients to servers increases (keeping the node population fixed), since there exist more clients in the network requesting services. The results represent average values obtained over 10 runs for each experiment. We should note here that this is also the first (implicit) performance comparison of SPIZ against that of IZR, since in [11] the authors of SPIZ compared the performance of SPIZ only against a service extended (non-adaptive) ZRP-based service discovery protocol using zone radius of 1 or 2 hops.

Before proceeding to the experiments mentioned above we

² By data traffic we mean the IARP or IERP packets that carry service invocation data and not the IARP advertisement packets or the IERP query and reply packets.

³ We assume that once the FTP session with a server has been established, then if the server gets disconnected from the client before the completion of the item transfer, the client cannot transfer the session to another server.

investigate (see Figure 2 and 3) what is the optimal value for T, which represents how many seconds the BFO will increase or decrease the broadcast intervals each time.

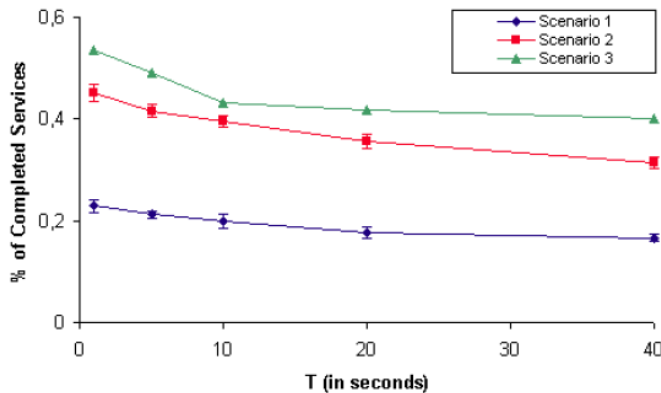


Fig. 2. Effect of the T parameter on the Percentage of Completed Services for AVERT.

Adjusting the broadcast intervals with larger step T leads to possibly greater energy gains but at the cost of decreased success ratios for AVERT. This is because if during a service session all routes toward the destination expire due to infrequent broadcasting, the client or server must try to discover again the route toward each other. In the mean time the application's tolerance may be exhausted and the service session may break before completion.

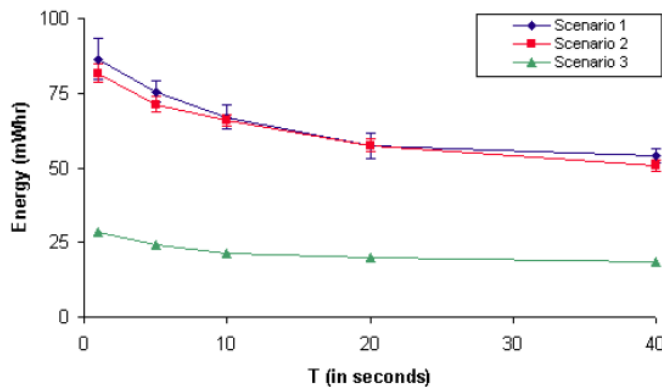


Fig. 3. Effect of the T parameter on the Energy Consumption per node for AVERT.

However, changing the broadcast intervals by smaller values (e.g. when $T=1$ second up to $T=5$ seconds), the nodes may gradually reach the optimal broadcasting interval based on current conditions. This means that in case of wrong estimations by nodes, before the change begins to affect in undesired ways the current service sessions in the network, the nodes are given the chance to re-adapt the broadcasting intervals. In the case that T has large values this is more difficult to happen, since increasing (or decreasing) the broadcast intervals a lot may severely impact routes before a

node can react to correct the situation. Also smaller T means that more fine-grained adaptation can take place. What validates this is that the value of σ decreases as T increases, as shown in Figure 4.

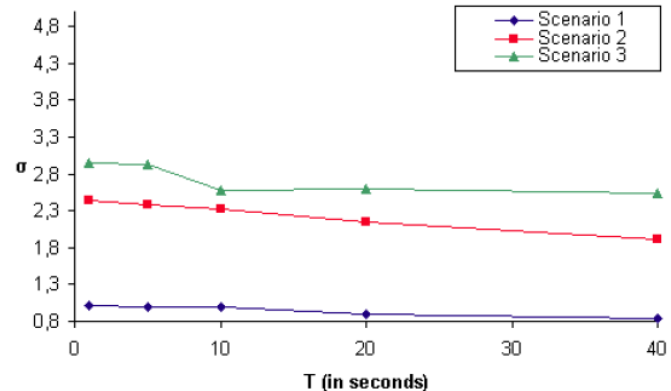


Fig. 4. Effect of the T parameter on σ for AVERT.

Also it is worth mentioning that as the ratio of clients to servers decreases the service success ratio increases since there are more available servers, possibly located closer to the requesting clients. Moreover, when the number of clients is low, there is less congestion in the network since the data traffic due to service invocation is less and also localized around nodes.

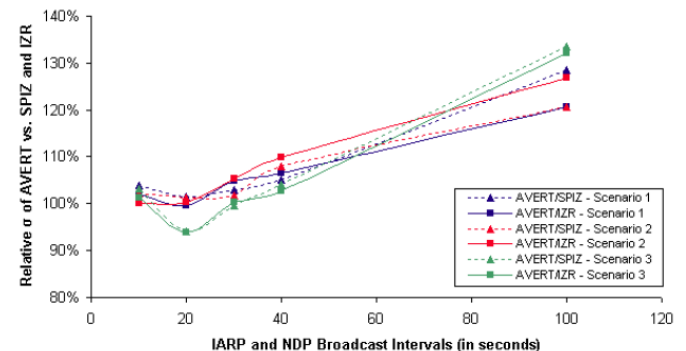


Fig. 5. Performance gains of AVERT against IZR and SPIZ.

Proceeding to the comparison of AVERT against SPIZ and IZR we fix T to 1 second, and compare the protocols based on the achieved σ ratios. In Figure 5 the y-axis represents the relative gains in the σ achieved by AVERT against the σ of the other protocols ($\sigma(\text{AVERT})/\sigma(\{\text{IZR}, \text{SPIZ}\})$), while the x-axis represents different broadcast intervals for IARP and NDP in the range of 10 to 100 seconds. Under the two scenarios tested, AVERT, using the BFO mechanism for determining a near optimal value for the NDP and IARP Broadcast intervals, shows performance gains of up to 35% (depending on the values for the NDP and IARP Broadcast intervals chosen by IZR and SPIZ). Since the two latter protocols cannot adapt their rate of broadcasting IARP and NDP messages to the conditions in the MANET, they are confined to use a “hard coded” value for these rates. However, this value cannot achieve the maximum performance under all

MANET scenarios, or even within the same scenario assuming that network conditions change dramatically during the lifetime of the MANET. For example, in Figure 6 we plot how the broadcast intervals of NDP and IARP impact the performance of IZR as the frequency of requesting services changes (under Scenario-3). It is obvious from the figure that as the service usage frequency increases, the performance is optimized using shorter broadcast intervals.

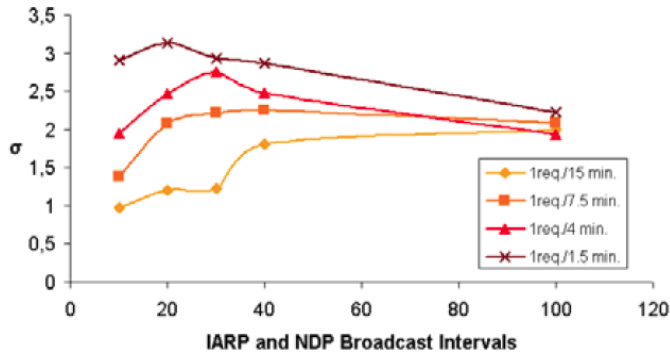


Fig. 6. Performance of SPIZ under different service request frequencies.

Returning to the results shown in Figure 5, the performance of AVERT is presented to be slightly worse than that of SPIZ and IZR only in Scenario-3. Actually in this scenario the BFO mechanism of AVERT does not have adequate feedback from data traffic (data traffic is low) and hence cannot tune the broadcasting frequency optimally. This is reflected especially when comparing AVERT with IZR and SPIZ when the latter protocols use relatively low broadcasting intervals.

IV. CONCLUSIONS AND FUTURE WORK

Considering the results above we conclude that choosing a small value of T is more effective, allowing smooth adaptation of the NDP and IARP broadcast intervals to current network conditions. Comparing AVERT to the non adaptive protocols IZR and SPIZ shows that employing a method for determining the optimal NDP and IARP broadcast intervals in real time can lead to significant performance improvement both in terms of successful service invocations and energy consumption. In our future work we plan to extend our performance evaluation of AVERT, by considering scenarios of higher mobility and also of higher density.

REFERENCES

- [1] R. Koodli and C. E. Perkins, "Service discovery in on-demand ad hoc networks," IETF Internet Draft, draft-koodli-manet-servicediscovery-00.txt, October 2002.
- [2] J. Antonio García-Macías and Dante Arias Torres, "Service Discovery in Mobile Ad hoc Networks: Better at the Network Layer?," Proc. IEEE Intl. Workshop on Wireless and Sensor Networks (WSNET'05), Oslo, Norway, June 2005, pp. 452-457.
- [3] D. Doval and D. O'Mahony, "Nom: Resource Location and Discovery for Ad hoc Mobile Networks," Proc. 1st Annual Mediterranean Ad hoc Networking Workshop, Med-hoc-Net 2002, Sardegna, Italy, Sept 4-6, 2002.
- [4] Engelstad, P.E., Zheng, Y., Koodli, R., Perkins, C.E., "*Service Discovery Architectures for On-Demand Ad Hoc Networks*", International Journal of Ad Hoc and Sensor Wireless Networks, Old City Publishing (OCP Science), Vol. 2. Number 1, March 2006, pp. 27-58
- [5] J. L. Jodra, M. Vara, J. M. Cabero, J. Bagazgoitia and J. L. Jodra, "Service Discovery Mechanism Over OLSR for Mobile Ad hoc Networks," Proc. 20th International Conference on Advanced Information Networking and Applications - Volume 2 (AINA'06), 2006, pp. 534-542.
- [6] L. Li and L. Lamont, "A Lightweight Service Discovery Mechanism for Mobile Ad Hoc Pervasive Environment Using Cross-Layer Design," Proc. 3rd IEEE International Conference on Pervasive Computing and communications Workshops, March 2005, pp.55-59.
- [7] Christopher N. Ververidis, George C. Polyzos, "Service Discovery for Mobile Ad Hoc Networks: A Survey of Issues and Techniques," to appear in IEEE Communications Surveys and Tutorials (2008).
- [8] Christopher N. Ververidis and G. C. Polyzos, "Extended ZRP: Performance Evaluation of a Routing Layer Based Service Discovery Protocol for Mobile Ad Hoc Networks," Proc. 2nd Annual International Conference on Mobile and Ubiquitous Systems: Networking and Services (MobiQuitous'05), San Diego, California, July 2005, pp. 114-123.
- [9] Chang-Seok Oh, Young-Bae Ko and Young-Sung Roh, "An Integrated Approach for Efficient Routing and Service Discovery in Mobile Ad Hoc Networks," Proc. IEEE Consumer Communications and Networking Conference (CCNC'05), Las Vegas, Nevada, Jan. 2005.
- [10] R. Harbird, S. Halies and C. Mascolo, "Adaptive resource discovery for ubiquitous computing," Proc. 2nd Workshop on Middleware for Pervasive and Ad hoc Computing, Toronto, Canada, October 2004, pp.155-160.
- [11] D. Noh and H. Shin, "SPIZ: An Effective Service Discovery Protocol for Mobile Ad Hoc Networks," EURASIP Journal on Wireless Communications and Networking, vol. 2007, Article ID 25167, 13 pages, 2007, doi: 10.1155/2007/25167.
- [12] P. Samar, M. Pearlman, and Z.J. Hass, "Independent Zone Routing: An Adaptive Hybrid Routing Framework for Ad Hoc Wireless Networks," IEEE/ACM Transactions on Networking, 12(4), 2004, pp. 595-608.
- [13] M. Pearlman and Z.J. Hass, "Determining the Optimal Configuration for the Zone Routing Protocol," IEEE Journal on Selected Areas in Communications, 17(8), 1999, pp. 1395-1414.
- [14] Christopher N. Ververidis, George C. Polyzos, "A Routing Layer Based Approach for Energy Efficient Service Discovery in Mobile Ad Hoc Networks," to appear in Wireless Communications and Mobile Computing Journal, Wiley, (2008).
- [15] Scalable Networks, The Qualnet Simulator, <http://www.scalable-networks.com>.