

Device And Service Discovery in Mobile Ad-hoc Networks

January 16, 2007

Master 2 SDRP: Initiation à la Recherche

Student: Jérémie Albert
Supervisor: Serge Chaumette

Université Bordeaux 1
LaBRI

1 Introduction

When a wireless device arrives in an area where there are some other wireless devices, how can it discover them, communicate with them, use a service provided by one of them or provide itself a new service? This is one of the main challenges of mobile ad-hoc networks. Therefore, in this work, we consider mobile ad-hoc networks and we study discovery issues related to mobility. In this paper, we proceed in two steps. First, we deal with service discovery issues based on a paper entitled “A Classification of Service Discovery Protocols” [9]. Second, we do a bibliographic study about device discovery systems.

In a device or service discovery, several steps have to be performed. First, the device has to discover its neighbors, be discovered by them, and store information about them. Second, it has to maintain these information. Then, it uses or proposes a service to the community.

2 Service discovery in MANets

2.1 Service description

Service discovery protocols need that a service description exists. Service description enables the users to identify a specific service. The most used description format is the attribute-value structure. For instance, in Bluetooth Service Discovery Protocol (SDP) [2], service information is contained within a service record, which consists of a list of attributes. Many of the protocols use Extensible Markup Language (XML) [18] to describe services. GSD [1] chooses DARPA Agent Markup Language with Ontology Inference Language (DAML+OIL) [15].

2.2 Storage of service information

Information about available services depend on a storage system. In mobile ad-hoc networks, storage may be non-existent, for example if the network is composed of resource-poor devices. The most adapted storage system for a MANet is thus an unstructured distributed system. Indeed, in this type of network, communication is based on broadcast and multicast mechanisms. Several methods are used to obtain and maintain the service data:

- service providers flood the network with service advertisements;
- clients flood the network with discovery messages;
- nodes cache the service advertisements;
- nodes overhear in the network traffic and cache the interesting data.

There are several examples of protocols in which every interested node maintains its own view of the services and devices available in the network like GSD [1], or those described by Wu and Zitterbart [17], or Varshavsky and al [14].

Another important issue is the complexity of the view that each device has to maintain. Contrary to UPnP [4], GSD [1] allows each node to store the service descriptions available in a maximum number of hops.

2.3 Search methods

In unstructured distributed storage systems, nodes have to manage themselves how to find appropriate services (possibly by flooding). The more organized and distributed information is, the less search effort is required, but a complex storage mechanism makes the information consistency difficult to maintain. Therefore in very mobile networks flooding may be the only option for service lookup. There are two ways to obtain information about services:

- Passive Discovery (or Push Model) - services or brokers announce or advertise their presence
- Active Discovery (or Pull Model) - an application needs information about services or brokers and sends discovery messages

Most protocols implement both methods.

2.4 Maintenance

Maintenance is regarded as a permanent adjustment of the service information stored in nodes. In a MANet, each node stores the service information it provides and it is interested in. The maintenance against changes in service description has to be managed by advertising the service with a different description. The nodes which have an older description can then update it and those which do not have it yet are informed of the availability of this service.

The maintenance against topology changes is a real issue because of mobility. Protocols dealing with this problem include Service Rings [7] and LANES [6]. Service Rings [7] groups together devices that are both physically close to each other and offer similar services. LANES [6] groups together devices into a lane where each node knows its predecessor and its successor in order to help routing for sending messages.

2.5 Service selection

If several servers offer the same service, we have to determine which one is the most interesting to use. It can be chosen by the user or by an optimisation algorithm implemented on the client side. In the last case, a very important issue is the metric used to define the best offer. For instance, Varshavsky and al [14] consider the hop count from the client to the server.

2.6 Service usage

After discovery, the client can use the selected service by connecting to the server. Commands can be transmitted using Remote Procedure Call (RPC) like Java Remote Method Invocation (Java RMI) [11] or Simple Object Access Protocol (SOAP) [13].

2.7 Security and privacy

Security includes authentication, authorization, trust, confidentiality, integrity, non-repudiation and privacy. All these terms are defined in RFC2828 [12]. In MANet, security is so important that it has to be a part of the main design strategy. It is out of the scope of this work.

3 Device Discovery in MANets

3.1 Device Discovery Issues

3.1.1 Network volatility

In mobile ad-hoc networks, a node can become unreachable at any time because of mobility. Anticipation systems can hardly exist so we must find what we have to consider to detect a soon to happen disconnection. Signal power seems to be the most significant consideration because a low signal power device will statistically become more quickly out of reach than a high power device.

3.1.2 Fault tolerance

A node can stop all communications at any time because of a critical system error. We have to think about the way a node could make the difference between a node which has crashed and a node which is out of reach. Elhadef, Boukerche and Elkadiki [3] propose to assign tasks to pairs of mobiles and the outcomes of these are compared in order to diagnose the state of the wireless ad-hoc network.

3.1.3 Resource awareness

Mobile ad-hoc networks are most often composed of devices which have very heterogeneous resources. Indeed, the devices can be computers, mobile-phones, sensors or even a mix of different categories. Some protocols support integration of resource-poor devices. They usually delegate the work load to additional devices. For instance, Splendor [19] uses proxies to achieve privacy for service providers, offloading computational work and enabling mobile services to do authentication and authorization easily.

3.2 Wireless networks technologies support and device discovery

3.2.1 Wireless technologies presentation

Bluetooth. In 1998, Ericsson, IBM, Intel, Motorola, Nokia and Toshiba formed a consortium and adopted the code name Bluetooth for their proposed open specification. Bluetooth provides a way to connect and exchange information between devices such as mobile phones, laptops, PCs, printers, sensors via an unlicensed short-range radio frequency. It uses the microwave radio frequency spectrum in the 2.4 GHz to 2.4835 GHz range.

Wi-Fi. Wi-Fi is a brand originally licensed by the Wi-Fi Alliance to describe the underlying technology of wireless local area networks (WLAN) based on the IEEE 802.11 specifications. It was developed to be used for mobile computing devices in LANs. It uses both single carrier direct-sequence spread spectrum and multi-carrier OFDM (Orthogonal Frequency Division Multiplexing) radio technologies. It uses the same microwave radio frequency spectrum as Bluetooth.

3.2.2 Limitations wireless technologies

Bluetooth. The inquiry substrate has two 16-channel subsets called train. Ten milliseconds are necessary for each train to complete. By specification, each train must be repeated 256 times in order to allow sufficient time to collect all inquiry responses. The specification also stipulates that a minimum of three train switches must occur, meaning that each train must be iterated twice. The inquiry device runs two iterations of both trains, 256 times per iteration in order to ensure that all

listening devices in range will be on a common frequency and in the inquiry scan substrate during at least one inquiry time slot. The resulting total is 10.24 seconds, as shown in Figure 1.

$$2 \text{ trains} * 2 \text{ iterations} * 256 \text{ times} * 0.01 \text{ seconds} = 10.24 \text{ seconds}$$

Figure 1: Minimum Time Required for Bluetooth Device Discovery

In a noisy or error-prone environment where both devices are on the same frequency at the same time, packets transmitted may be corrupted therefore there is no guarantee of successful inquiry. In such situations, the inquiry time may be much longer than the default time of 10.24 seconds.

In Bluetooth networks, nodes are necessarily masters or slaves. A master cannot have more than 7 active or 255 passive slaves. A piconet is a set of bluetooth devices which have the same master. A slave can only communicate with the master so if the master gets out of the network, a slave has to become the new master of the piconet.

Bluetooth has a short-range radio frequency, needs less power than Wi-Fi and allows a passive listening mode.

Bluetooth security is handled by SAFER+ (Secure And Fast Encryption Routine). It was submitted as a candidate for the Advanced Encryption Standard and has a block size of 128 bits. In the Bluetooth standard, it is used for authentication and key generation.

Wi-Fi. Contrary to Bluetooth, Wi-Fi is an average range system, approximately 50 meters indoors and 100 meters outdoors. However, Wi-Fi communication requires much more power. Security is handled by keys. Wi-Fi networks using Wireless Equivalent Privacy (WEP) keys are not considered as secure networks. Those using Wi-Fi Protected Access 2 (WPA2) keys cannot be broken today in a reasonable time.

3.2.3 Solutions to make discovery more efficient

Decrease of Bluetooth inquiry time using IrDA. Spending 10.24 seconds to discover devices that are in range is unacceptable in many situations. A solution has been proposed to accelerate Bluetooth inquiry using IrDA [16]. In IrDA, device discoveries may use 1, 6, 8, or 16 time slots. Each time slot must last at least 25 milliseconds, with each response beginning within 10 milliseconds and completing within 70 milliseconds of the end of the packet sent by the device performing discovery. The device discovery time is thus 1.12 seconds as shown in Figure 2.

$$16 \text{ timeslots} * 0.070 \text{ seconds} = 1.12 \text{ seconds}$$

Figure 2: Time Required for IrDA Device Discovery

Decrease of Bluetooth inquiry time with a synchronized round based algorithm. When a given Bluetooth device is searching for other devices and services, it is not discoverable by other devices. NEC Laboratories propose an algorithm [5] implemented at the application level to resolve the Bluetooth discovery problem so that it can be efficiently used in a dynamic setting as required by mobile p2p applications:

- the device performs a discovery,
- the device moves into a notification phase and informs its neighbors (which were discovered within the current round) about the neighbors it has discovered,

- the device moves into a receiving phase and collects information from its neighbors that are in the notification phase and filters its set of “possible” neighbors using a “ping”.

This synchronized round based discovery algorithm performs significantly better than the random waiting time algorithm. For example, with 3 devices, this algorithm needs only 1.26 inquiries (approximately 50 seconds) on average to find 3 other devices whereas the random waiting algorithm needs 8.9 inquiries (approximately 150 seconds).

3.3 Building discovery paradigms based on the underlying technology

The underlying technology of discovery paradigms relies on suppositions. Thus, for example, discovery with rendezvous relies on neighborhood awareness.

3.3.1 Discovery with rendezvous

The rendezvous algorithm allows two nodes to become connected with each other. An example of a randomized rendezvous algorithm is given by Yves Metivier, Nesser Saheb and Akka Zemmari [10] in Figure 3.

1. the node n selects one of its neighbors $c(n)$ chosen at random;
 2. the node n sends 1 to $c(n)$;
 3. the node n sends 0 to its neighbors different from $c(n)$;
 4. the node n receives messages from all its neighbors.
- There is a rendezvous between n and $c(n)$ if n receives 1 from $c(n)$.

Figure 3: A randomized algorithm to obtain rendezvous

3.3.2 Discovery based on mobile agents

In order to collect or deliver information about the network, neighbors or services, we can use mobile agents. A mobile agent moves from node to node and is executed on each node to do some predefined operations. For example, it can deliver services advertisements to neighbors in a predefined maximum number of hops and/or according to some specifications. Mobile agents systems need a middleware executing on each node to receive and execute the mobile agents.

Tuples On The Air (TOTA) [8] is a middleware infrastructure explicitly conceived as a support for distributed computing in dynamic network scenarios. The middleware propagates tuples across a network on the basis of application-specific patterns and adaptively re-shapes the resulting distributed structures accordingly to changes in the network scenario.

4 Conclusion

In this paper, we have presented some device and service discovery protocols and issues in mobile ad-hoc networks. We have established the large diversity of non-compatible protocols. All discovery protocols which have been presented are high-level protocols which rely on many suppositions about the network. Furthermore, the underlying technology of wireless networks still has many deficiencies like discovery time and radio frequency sharing conflicts. Therefore, it might be worth redefining communication primitives in order not to depend on networks suppositions. This will be the future direction of this work.

References

- [1] Dipanjan Chakraborty, Anupam Joshi, Tim Finin, and Yelena Yesha. Gsd: A novel group-based service discovery protocol for manets. In *4th IEEE Conference on Mobile and Wireless Communications Networks (MWCN)*. IEEE, September 2002.

- [2] Bluetooth Consortium. Specification of the bluetooth system core version 1.0 b: Part e. service discovery protocol (SDP), November 1999.
- [3] Mourad Elhadef, Azzedine Boukerche, and Hisham Elkadiki. Performance analysis of a distributed comparison-based self-diagnosis protocol for wireless ad-hoc networks. Technical report, University of Ottawa, Canada, 2006.
- [4] UPnP Forum. UPnP device architecture version 1.0, June 2000. available online at <http://www.upnp.org/>.
- [5] Sidath B. Handurukande, Samrat Ganguly, and Sudeept Bhatnagar. *Fast Bluetooth Service Discovery for Mobile Peer-to-Peer Applications*. available on www.sigmobile.org/mobisys/posters/Handurukande.pdf.
- [6] Michael Klein, Birgitta Konig-Ries, and Philipp Obreiter. Lanes - a lightweighth overlay for service discovery in mobile ad hoc networks. Technical report, 2003-6, University of Karlsruhe, 2003.
- [7] Michael Klein, Birgitta Konig-Ries, and Philipp Obreiter. Service rings - a semantic overlay for service discovery in ad hoc networks. In *DEXA Workshops*, pages 180–185. 2003.
- [8] Marco Mamei, Franco Zambonelli, and Letizia Leonardi. *Tuples On The Air: a Middleware for Context-Aware Computing in Dynamic Networks*. Los Alamitos, CA, USA, 2003.
- [9] Raluca Marin-Perianu, Pieter Hartel, and Hans Scholten. A classification of service discovery protocols. June 2005.
- [10] Yves Metivier, Nasser Saheb, and Akka Zemhari. Analysis of a randomized rendezvous algorithm. *Information and Computation 184 (2003) 109-128*, April 2002. available at www.computerscienceweb.com.
- [11] Java Remote Method Invocation (Java RMI), 2007. available online at <http://java.sun.com/>.
- [12] R. Shirey. Request for comments: 2828. Internet Security Glossary, May 2000.
- [13] Simple Object Access Protocol (SOAP), 2007. available online at <http://www.w3.org/TR/soap/>.
- [14] Alex Varshavsky, Bradley Reid, and Eyal de Lara. The need for cross-layer service discovery in MANETs, January 2004.
- [15] The DARPA Agent Markup Language with Ontology Inference Layer, 2006. available online at <http://www.daml.org/> and <http://www.ontoknowledge.org/oil/>.
- [16] Ryan Woodings, Derek Joos, Trevor Clifton, and Charles D. Knutson. *Rapid Heterogeneous Connection Establishment: Accelerating Bluetooth Inquiry Using IrDA*. Brigham Young University Provo, Utah 84602.
- [17] J. Wu and M. Zitterbart. Service awareness in mobile ad hoc networks. Boulder, Colorado, USA, March 2001. Paper Digest of the 11th IEEE Workshop on Local and Metropolitan Area Networks (LANMAN).
- [18] Extensible Markup Language (XML), 2007. available online at <http://www.w3.org/TR/xml11/>.
- [19] Feng Zhu, Matt W. Mutka, and Lionel M. Ni. Splendor: A secure, private, and location-aware service discovery protocol supporting mobile services. In *PerCom*, pages 235–242. ACM Press, New York, NY, USA, 2003.