

Experimental Implementation of Asynchronous Rendezvous Protocols using Microsoft Sora

Jiseon Lee, Chulmin Kim, Cheeha Kim
Department of Computer Science and Engineering
Pohang University of Science and Technology (POSTECH)
Pohang, Korea
{happyjskr,kimcm94,chkim}@postech.ac.kr

ABSTRACT

In opportunistic networks, mobile devices can communicate with each other via connections between them that are not always available. After a device discovers another device in its transmission range, communication is enabled. Without any knowledge about geographical positions of mobile devices, neighbor discovery is energy consumption process. For example, continuous scanning is the fastest way to find neighbors but most mobile devices are battery-powered and cannot afford persistent scanning. So, neighbor discovery with minimal energy consumption is very important. Most neighbor discovery schemes adopt periodic sleep-wake scheduling associated with rendezvous capability. This means that each device just explores neighbors while they wake-up. Many rendezvous protocols have been proposed to reduce energy consumption. In this paper, we implement three promising asynchronous rendezvous protocols using Microsoft Research Software Radio (Sora) board and present the experimental results. Our experiment provides experimental insight into the implementation of rendezvous protocols in real situations.

Categories and Subject Descriptors

D.5.3 [Network Performance Evaluation]: Network Experimentation

General Terms

Experimentation

Keywords

Opportunistic Networks, Neighbor Discovery, Protocols

1. INTRODUCTION

The wireless mobile device market has grown rapidly. Depending on the device characteristics, a variety of applications (e.g., social networking [1] and mobile gaming community [2], [3]) and networks (e.g., mobile ad hoc networks

(MANET) [4] and cognitive radio networks [5]) can be operated without any infrastructure. A network comprised of mobile devices that is not supported by any infrastructure is known as an opportunistic network.

In opportunistic networks, a mobile device cannot always communicate with other devices. Communication is possible only when a device is within the transmission range of another device and they become neighbors. However, no device has any knowledge about the geographical positions of other devices. Therefore, a device needs to find other devices to determine whether they are within transmission range. Continuous scanning is the best solution for finding neighbors rapidly, but this is an energy-consuming process. Most mobile devices are battery-powered and cannot afford persistent scanning.

Thus, neighbor discovery with minimal energy consumption is an important objective in opportunistic networks. Each device tries to minimize its energy consumption while satisfying the discovery latency time limit. Most neighbor discovery schemes use periodic sleep-wake scheduling with rendezvous capability [6]. These schemes allow devices to sleep to save energy and they wake up at certain times to explore their neighbors at that time. The rendezvous protocol makes mobile devices know the time to wake up. Many rendezvous protocols have been proposed to reduce energy consumption [6][7][8][9][10].

Rendezvous protocols in opportunistic networks need to discover each other even when the periodic sleep-wake scheduling interval is asynchronous [10]. This condition means that each mobile device can start its rendezvous protocol operation individually. Existing rendezvous protocols that satisfy this condition can be categorized as stochastic-, quorum-, and prime-based techniques. Most rendezvous protocols are based on a time slot where time is divided into slots of equal size, which is the unit for transmission or sleep by all devices. A unit of the wake time is known as an active slot. If the active slots of two devices overlap, a rendezvous may occur and they can communicate. The Birthday protocol [7], a typical stochastic-based protocol, selects the active slot probabilistically. The slot state of each device is determined by a given probability. If the given probability is high, the rendezvous probability is also high but it cannot guarantee the discovery latency bound. The active slot in the quorum-based protocol [8] is determined by a grid square where time is divided into sets of m^2 contiguous intervals, which are arranged as a two dimensional $m \times m$ array. Each device selects a row and a column in the array randomly, so all devices can meet at least two points of the choices

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.

SRIF '13, August 12, 2013, Hong Kong, China.

Copyright 2013 ACM 978-1-4503-2181-5/13/08 ...\$15.00.

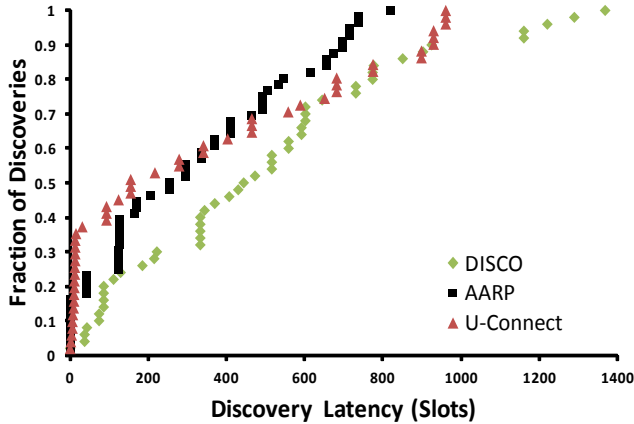


Figure 2: The Distribution of Discovery Latency - Symmetric case: 5%

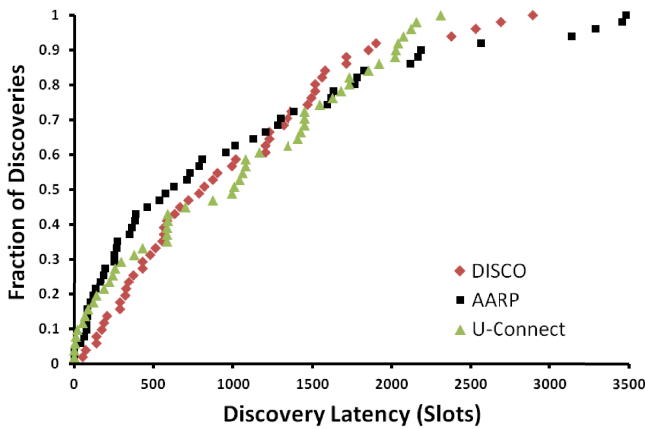


Figure 3: The Distribution of Discovery Latency - Asymmetric case: 1% and 10%

in an office area (Fig. 1). The Sora radio control board (RCB) was in a PC. Thus, two PCs were used as devices to implement the rendezvous protocols and there were antennas on the backs of the PCs.

We evaluated the discovery latency of the devices. Given the same amount of energy, the discovery latency is different for the protocols. The faster discover latency, the better a rendezvous protocol. In this experiment, two devices started the rendezvous protocol randomly (i.e., asynchronously) and we conducted the experiment 50 times to determine the distribution.

3.1 Performance Evaluation

Symmetric discovery latency is represented using CDF (Fig. 2). We set the energy consumption rate to 5% to normalize energy consumption as the same. It means that a device is awake for 5 active slots in a duty cycle of 100 slots. The rendezvous protocols select the appropriate parameters to satisfy this rate (DISCO: $P_1 = 37$, $P_2 = 43$; U-connect: $P = 31$; AARP: $P = 41$). DISCO has an even distribution but its worst-case discovery latency is the longest. With U-connect, 50% of the distribution is within 200 slots but the latency is longer than that of the other methods. AARP

has the shortest worst-case discovery latency, and its average rendezvous point is also shorter than the others (DISCO: 491, U-connect: 336, AARP: 308).

In the asymmetric case (Fig. 3), we consider the case where one device consumes 1% energy and the other consumes 10% energy. To achieve this for DISCO, 1% energy consumption device picks (191-211) prime numbers and the other picks (17-23) prime numbers. Likewise, for U-connect, one device consuming 1% energy selects prime number 155 and the other selects 17. For AARP, device consuming 1% energy picks 199 and the other 19. All the protocols have a similar trend but AARP had a high probability of early discovery.

3.2 Implementation issues

In practice, an aligned time slot is almost impossible. Without prior information and global synchronization, each device cannot know the start point of a time slot initiated by other devices. Thus, each device starts in its own time and they wake up at different times even if they have overlapping slots. This issue is known as slot non-alignment [9]. To address the slot non-alignment issue, we doubled the slot duration to provide an adequate duration for a rendezvous. This was proved in [12] and we verified it experimentally.

Another issue is interference. During the neighbor discovery experiment, a rendezvous never occurred in some cases because Sora experienced interference at the rendezvous time. The development of wireless communication means the current channel state can be full of them. In the experiment, many WLAN access points (APs) and wireless devices co-exist with Sora because the building contains other offices. Thus, it was hard to escape interference in most places. This is a normal communication environment at present so the results of our experimental implementation were realistic. However, previous studies assume an ideal channel state and they try to minimize the active slots within the duty cycle. If interference occurs at the rendezvous point in this situation, their rendezvous will be less likely to occur in one duty cycle or never at all. We may conclude from this observation that when designing a rendezvous protocol, it is necessary to count interference factor.

4. CONCLUSIONS

In this paper, we conducted an experimental implementation of asynchronous rendezvous protocols and showed the distribution of the discovery latency measured using Microsoft Research Sora boards. The worst-case discovery latency agreed with the calculated results but distribution presents practical expectation of discovery latency. Our experimental results highlight some practical issues such as tendency of the discovery latency and average discover time shown in Figure 2 and 3 and these distribution results may provide insight related to implementation. Depending on users' requirement such as high priority on early discovery latency or average discovery latency or bound of the discovery latency, preference of rendezvous protocol will be different and they can get insight from our experimental results.

5. ACKNOWLEDGMENTS

"This research was supported by the MKE(The Ministry of Knowledge Economy), Korea and Microsoft Research, under IT/SW Creative research program supervised by the NIPA(National IT Industry Promotion Agency)" (NIPA-2012-

H0503-12-1031)

“This research was supported by the MSIP(Ministry of Science, ICT&Future Planning), Korea, under the ITRC(Information Technology Research Center) support program supervised by the NIPA(National IT Industry Promotion Agency)” (NIPA-2013-H0301-13-3002)

6. REFERENCES

- [1] E. Miluzzo, N. D. Lane, K. Fodor, R. Peterson, H. Lu, M. Musolesi, S.B. Elisenman, X.Zheng, and A.T. Campbell, “Sensing meets mobile social networks: the design, implementation and evaluation of the cenceme application,” In Proceedings of the 6th ACM conference on Embedded network sensor systems, pp. 337-350, ACM, 2008.
- [2] “Nintendo 3ds - stretpass,” <http://www.nintendo.com/3ds/hardware>.
- [3] “Sony ps vita - near,” <http://us.playstation.com/psvita>.
- [4] J. Macker and S. Corson, “Mobile Ad Hoc Networks (MANETs),” <http://www.ietf.org/proceedings/01dec/183.htm>, December 2001.
- [5] S. Haykin, “Cognitive radio: Brain-empowered wireless communications,” *IEEE J. Select. Areas Commun.*, 23 (2) pp. 201-220, 2005.
- [6] H. Ko, S. Oh, and C. Kim, “Adaptive, asynchronous rendezvous protocol for opportunistic networks,” *Electronics Letters*, Vol.48, No.8, pp. 462-464, April 2012.
- [7] M. J. McGlynn and S. A. Borbash, “Birthday protocols for low energy deployment and flexible neighbor discovery in ad hoc wireless networks,” In Proceedings of MobiHoc’01. ACM, pp. 137-145, 2001.
- [8] Y.-C. Tseng, C.-S. Hsu, and T.-Y. Hsieh, “Power-saving protocols for IEEE 802.11-based multi-hop ad hoc networks,” In INFOCOM, 2002.
- [9] P. Dutta and D. Culler, “Practical asynchronous neighbor discovery and rendezvous for mobile sensing applications,” In SenSys ’08: Proceedings of the 6th ACM Conference on Embedded Network Sensor Systems, pp. 71-84, 2008.
- [10] A. Kandhalu, K. Lakshmanan, and R. R. Rajkumar, “U-connect: a low-latency energy-efficient asynchronous neighbor discovery protocol,” In Proceedings of IPSN ’10, pp. 350-361, 2010.
- [11] H. L. Ivan Niven and Herbert S. Zuckerman, “An Introduction to the Theory of Numbers,” John Wiley and Sons, 1991.
- [12] D. M. Yang, J. M. Shin, and C. H. Kim, “Deterministic rendezvous scheme in multi-channel access networks,” *Electronics Letters*, Vol. 46, No.20, pp. 1402-1404, 2010.