

Finding Mobile Applications in Cellular Device-to-Device Communications: Hash Function and Bloom Filter-Based Approach

P.Sumanjali , Ch.Sudhakar Reddy

Department Of Information Technology, G.Narayanamma Institute Of Science &Technology Shaikpet,Hyderabad.
Sumanjalipandem@Gmail.Com

Abstract: *The rapid growth of mobile computing technology and wireless communication have significantly increased the mobile users worldwide. We propose a code-based discovery protocol for cellular device-to-device (D2D) communications. To realize proximity based services such as mobile social networks and mobile marketing using D2D communications, each device should first discover nearby devices, which have mobile applications of interest, by using a discovery protocol. The proposed discovery protocol makes use of a short discovery code that contains compressed information of mobile applications in a device. A discovery code is generated by using either a hash function or a Bloom filter. When a device receives a discovery code broadcast by another device, the device can approximately find out the mobile applications in the other device. The proposed protocol is capable of quickly discovering massive number of devices while consuming a relatively small amount of radio resources. We analyze the performance of the proposed protocol under the random direction mobility model and a real mobility trace. By simulations, we show that the analytical results well match the simulation results and that the proposed protocol greatly outperforms a simple non-filtering protocol*

Keywords: Bloom Filter, D2D Communication, Discovery Protocol, Hash Function, Radio Resource.

1. Introduction

Remote sensor organize (WSN) is conveyed there is typically a need to refresh surrey old little projects or parameters put away in the sensor hubs. This can be accomplished by the information disclosure and scattering convention, which offices a source to infuse little projects, orders, questions and setup parameters to sensor hubs. Note that it is different from the code dissemination protocols which distribute large binaries to reprogram the whole network of sensors. For example, efficiently disseminating a binary file of tens of kilobytes requires a code dissemination protocol.

While disseminating several two-byte configuration parameter requires data discovery and dissemination protocol. Considering the sensor hubs could be appropriated in an unforgiving situation, remotely scattering such little information to the sensor hubs through the remote channel is a more favoured and down to earth approach than manual intercession. Persuade by the above perception, this paper as the accompanying principle commitment 1 the need of conveyed information disclosure and spread convention is not totally new, but rather past work did not address this need we study the functional requirement of such protocol, and said there design objective.

In this paper, we propose a D2D discovery protocol that discovers nearby devices in the context of cellular D2D communications. The mobile application discovery protocol in a cellular environment should be efficient in radio resource usage, energy-efficient to maximize the battery life, and capable of quickly discovering massive number of devices. However, it is challenging to achieve these goals in the environment in which there exist a number of devices

with many mobile applications as in Fig. 1. In a dense urban scenario, it is possible that there are hundreds (or even thousands) of devices, each with tens of applications, are present in one device's proximity. We aim to design a discovery protocol that is able to efficiently pinpoint the devices with matching applications among a large number of devices.

We design a code-based D2D discovery protocol that makes use of a discovery code containing compressed information of mobile applications in a device. The proposed protocol can minimize the amount of application information broadcast for the discovery procedure, by using a filtering capability of a discovery code. To generate a discovery code, the proposed protocol uses either a hash function or a Bloom filter [8].

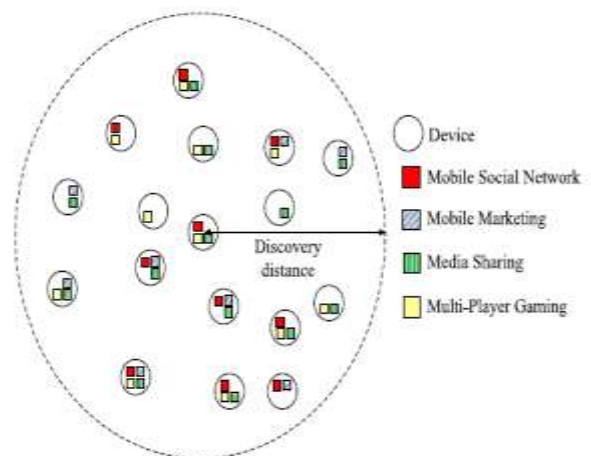


Figure 1- Example of a mobile application discovery scenario for D2D communications

2. Related Work

The administration and neighbour disclosure in remote condition has been looked into by a substantial assortment of works in the unique circumstance of specially appointed systems (e.g., [12], [13], [14], [15], [17], [20], [21], [22], [23], [24], [25], [26], [27], [28]), sharp systems (e.g., [17], [30]), and the D2D correspondence (e.g., [9], [10], [11])

In spite of the fact that various disclosure conventions have been proposed, cell D2D interchanges have extraordinary prerequisites for the disclosure to understand the vicinity based administration, which can't be met by the current conventions. The revelation convention for cell D2D interchanges ought to be firmly incorporated with the physical and MAC layer. In impromptu systems utilizing an irregular get to convention, a revelation edge can be dealt with as an ordinary information outline by bring down layers. In this way, it is not required to adjust the physical and MAC layers for sending revelation outlines.

Nonetheless, in OFDMA-based cell systems, for example, LTE/LTE-Advanced systems, the base station ought to dispense isolate time-recurrence radio assets devoted for transmitting the revelation flag [5]. In this manner, the execution what's more, the conduct of the D2D revelation convention rely on upon how the radio assets are assigned by lower layers.

The current disclosure conventions for impromptu systems (e.g., [12], [13], [14], [15], [17], [20], [21], [22], [23], [24], [25], [26], [27], [28]) can't be connected to the cell D2D correspondence as it seems to be, since they don't consider the physical also, MAC layer viewpoints by any stretch of the imagination. Then again, there are physical and MAC layer revelation conventions for D2D correspondence (e.g., [9], [10], [11]). The creators of [9] propose a D2D revelation convention for FlashLinQ [31], in which each gadget intermittently communicates a disclosure flag. In [10], the creators propose a joint iterative deciphering answer for multi-client recognition of the identifier sent by every gadget. In [11], a reference point based D2D revelation conspire is proposed for the LTE framework. Be that as it may, every one of those conventions concentrate just on the physical and MAC layers, not considering the application layer by any stretch of the imagination. Then again, the proposed revelation convention firmly incorporates the physical, MAC, and application layer parts of the revelation.

One of the key elements of the D2D revelation for the closeness based administration is the "vicinity based trigger." In the event that two gadgets come into nearness, every gadget ought to be ready to right away distinguish the other gadget and to trigger an activity for the versatile application. For instance, a client of versatile interpersonal organizations can be quickly alarmed when a companion comes into closeness. This component requires ceaseless promotion of utilization data, which expends a ton of radio assets. This closeness based trigger has not been considered by the disclosure conventions for specially appointed systems.

3. MOBILE APPLICATION DISCOVERY PROTOCOL

3.1. Motivation of the Proposed Code-Based Discovery Protocol

The proposed protocol aims to make a device discover all the common applications in all devices within the discovery distance d . For a given device, the common applications in the nearby devices can change over time as devices move into or out of the discovery distance or applications in the nearby devices become active or inactive. The proposed protocol is designed to track the changes in the common applications in the nearby devices for each discovery period. In the proposed code-based discovery protocol, each device can find out whether there is any nearby device with a given application or not, based on the discovery code received during the code exchange period. Then, the device broadcasts the application information message for an application only when there is a nearby device with that application. In this way, the proposed protocol can minimize the unnecessary transmission of an application information message discovery code.

In the proposed code-based discovery protocol, a discovery code is used to filter out the applications not in common. A discovery code contains compressed information of the set of the names of all applications in a device. When device m receives a discovery code from device n , device m can test if a certain application is in device n or not. Suppose that discovery code c is generated from a application name set

3.2. Hash Function-Based Discovery Code

We can use a hash function to generate a discovery code. A hash function maps long variable-length data to a short fixed-length hash value. Let h denote a hash function mapping a variable-length character string to a binary sequence of d bits. A hash value obtained by the hash function h should be evenly distributed over the range of 0 to $2^d - 1$. For the hash function h with such a property, we can use a cryptographic hash function such as SHA-1 and MD-5. Since the hash function h maps a larger set of data (i.e., character string) to a smaller set of data (i.e., binary sequence), a collision can happen in the case that two different character strings are mapped to the same binary sequence

3.3. Bloom Filter

A Bloom filter is a space efficient data structure for probabilistic represent of set of items $S = \{s_1, s_2, s_3 \dots, s_n\}$ with an array of n bits with m independent hash functions $h = \{h_1, h_2, h_3 \dots, h_m\}$. of n elements to support membership queries. The idea is to allocate a vector v with m bits, initially all set to 0, and then choose k independent hash functions, h_1, h_2, \dots, h_k , each with range $\{1, \dots, m\}$. For each element $a \in S$, the bits at the positions $h_1(a), h_2(a), \dots, h_k(a)$ in v are set to 1 (A particular bit might be set to 1 multiple times).

4. Description Of The Proposed Code Based Discovery Protocol

For the proposed protocol, each device maintains a discovery table, in which the application information about other devices are stored. A discovery table has five columns, which are device identifier (ID), application version (VER), update required (UR), proximity (PROX) and application information (INFO). Each row of a discovery table represents the information of applications in one device. Let $ID_i(m)$, $VER_i(m)$, $UR_i(m)$, $PROX_i(m)$, and $INFO_i(m)$ denote each column of i th row of the discovery table in device m . The application version of a device indicates a version of the set of applications within a device. Let $rk(m)$ denote the application version of device m at discovery period k . The application version $rk(m)$ is an integer of w bits, the value of which ranges from 0 to $2^w - 1$. When a device is turned on, the device has an empty discovery table (i.e., no row in the discovery table) and the application version can start from zero. The application version $rk(m)$ increases by one when the set of application information in device m changes. That is, in the case that the set of application information in device m changes in discovery period k , the application version is $rk(m) = (rk(m) - 1 + 1) \bmod 2^w$. When the application version changes, the device resets the discovery table.

The CDBP protocol was designed to provide authenticated broadcast from a base station towards all nodes of a wireless network. The protocol is based on a delayed disclosure of symmetric keys, and it requires the network to be loosely time synchronised. The protocol computes a MAC for every packet to be broadcast, by using different keys. The transmission time is split into time intervals of Δ_{int} time units each, and each key is tied to one of them. The keys belong to a key chain k_0, k_1, \dots, k_n generated by Base station BS by means of a public one-way function F .

In order to generate this chain, Base station BS randomly chooses the last key k_n and repeatedly applies F to compute all the other keys, whereby $k_i := F(k_{i+1})$, for $0 \leq i \leq n-1$. The key-chain mechanism together with the one-way function F , provides two major advantages: (i) a key k_i can be used to generate the beginning of the chain k_0, \dots, k_{i-1} , by simply applying F as many times as necessary, but it cannot be used to generate any of the subsequent keys; (ii) any of the keys k_0, \dots, k_{i-1} can be used to authenticate k_i . Each node m_j is pre-loaded with a master key $k_{bs:m_j}$ for unicast communications with BS.

The CDBP protocol is constituted by two main phases: bootstrapping new receivers and authenticated broadcast. The former establishes the node's initial setting in order to start receiving the authenticated packets, the latter describes the transmission of authenticated information.

5. Performance Evaluation

5.1. Simulation Model and Parameters

We use Network Simulator Version-2 (NS2) [15] to simulate our proposed algorithm. In our simulation, the channel capacity of mobile hosts is set to the same value: 2 Mbps. We use the distributed coordination function (DCF) of IEEE 802.11 for wireless LANs as the MAC layer protocol. It has the functionality to notify the network layer about link breakage.

In our recreation, versatile hubs move in a 1000 meter x 1000 meter district for 25 seconds reenactment time. All hubs have a similar transmission scope of 250 meters. The mimicked movement is Constant Bit Rate (CBR). Our reproduction settings and parameters are condensed in table 1

No. of Nodes	50
Area Size	1000 X 1000
Mac	802.11
Radio Range	250m
Simulation Time	25 sec
Traffic Source	CBR
Packet Size	512
Receiving Power	0.395
Sending power	0.660
Idle Power	0.035
Initial Energy	10.3 J
Rate	50,100,150,200 and 250Kb

Table 1: Simulation Settings

5.2. Performance Metrics

We assess principally the execution as indicated by the accompanying measurements.

Average Packet Delivery Ratio: It is the proportion of the number of bundles got effectively and the aggregate number of parcels transmitted

Average Routing overhead: It is the average number of routing packets by nodes.

Delay: It is the time taken by the packets to reach the receiver.

Average Throughput: It is the amount of successful message delivery over a node

6. Results and Discussion

We investigate the performance of proposed ESRP protocol using the ns-2 simulator with the necessary extension and compare it with AODV. In our simulations, we use a fixed transmission range of 250 meters, which is supported by most of real time and current network interface cards. We used the "random waypoint" with speed of nodes is uniformly distributed between 0 to maximum speed of 20 m/s with pause time value of 60 sec. All mobile nodes to be equipped with IEEE 802.11 network interface card and data

rates of 2.5Mbps. The initial energy of all the nodes is 10J. The transmission power is 600mW and the receiving power is 300mW. Finally, source nodes generate CBR (constant bit rate) traffic. Traffic sessions are generated randomly on selected different source- destinations with a packet size of 512 bytes.

Every node in a network has to run our algorithm whenever it becomes an intermediate node to forward the information of source nodes. Each simulation was run for the duration of 200 seconds and sampled data we collected from simulation is average of 4 times.

Our aim is mainly to improve the reliability and packet delivery ration.

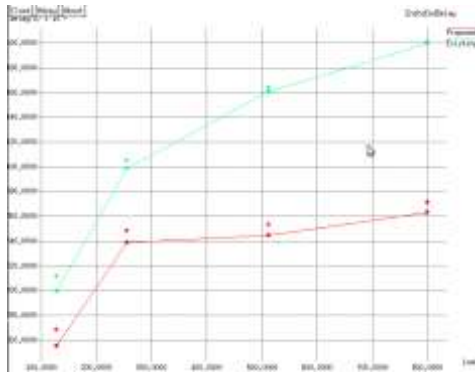


figure 2:end to end delay

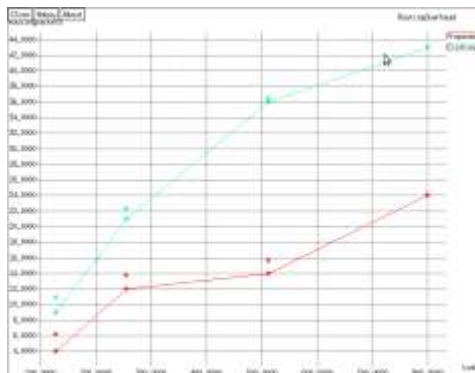


figure 3:Routing overhead

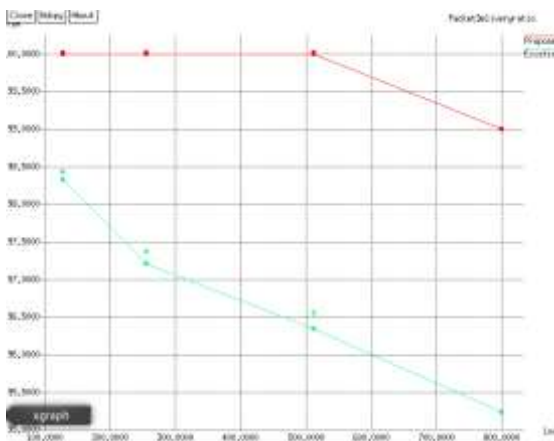


figure 4:Packet Delivery Ratio

7. Conclusion

The proposed a code-based discovery protocol for D2D communications in OFDMA cellular networks. The proposed protocol aims at finding out all nearby devices with common mobile applications in a radio resource and energy-efficient way and also to cover the discovery distance. The proposed protocol makes use of the hash function-based and bloom filter-based discovery codes which contain compressed information of mobile applications in a device. By simulation and analysis, it has been shown that the proposed discovery protocol greatly reduces the expected number of bits for discovery procedure by filtering out the information of applications not in common.

8. Reference

- [1] L. Lei, Z. Zhong, C. Lin, and X. Shen, "Operator controlled device-to-device communications in LTEAdvanced networks," *IEEE Wireless Commun. Mag.*, vol. 19, no.3, pp. 96–104, Jun. 2012.
- [2] G. Fodor, E. Dahlman, G. Mildh, S. Parkvall, N. Reider, G. Mikls, and Z. Turnyi, "Design aspects of network assisted device-to device communications," *IEEE Commun. Mag.*, vol. 50, no. 3, pp.170–177, Mar. 2012.
- [3] S. J. Yi, S. D. Chun, Y. D. Lee, S. J. Park, and S. H. Jung, "Radio Protocols for LTE and LTEAdvanced". Singapore: John Wiley & Sons Singapore Pte. Ltd., 2012.
- [4] 3GPP TR 22.803, "Technical specification group services and system aspects; feasibility study for proximity services (ProSe); release 12," 2012.
- [5] 3GPP TS 36.843, "Study on LTE device to device proximity services radio aspects; release 12," 2014.
- [6] N. Kayastha, D. Niyato, P. Wang, and E. Hossain, "Applications, architectures, and protocol design issues for mobile social networks: A survey," *Proceedings of the IEEE*, vol. 99, no. 12, pp.2130–2158, Dec. 2011.
- [7] M. Corson, R. Laroia, J. Li, V. Park, T. Richardson, and G. Tsirtsis, "Toward proximity-aware internetworking," *IEEE Wireless Commun. Mag.*, vol. 17, no. 6, pp. 26–33, Dec. 2010.
- [8] A. Broder and M. Mitzenmacher, "Network applications of bloom filters: A survey," *Internet Mathematics*, vol. 1, no. 4, pp. 485–509, Jan. 2004.
- [9] F. Baccelli, N. Khude, R. Laroia, J. Li, T. Richardson, S. Shakkottai, S. Tavildar, and X. Wu, "On the design of device-to-device autonomous discovery," in *Proc. COMSNETS 2012*, Bangalore, India, Jan. 2012, pp. 1–9.
- [10] A. Vigato, L. Vangelista, C. Measson, and X. Wu, "Joint discovery in synchronous wireless networks," *IEEE Trans. Commun.*, vol. 59, no. 8, pp. 2296–2305, Aug. 2011.
- [11] K. Doppler, C. B. Ribeiro, and J. Knecht, "Advances in D2D communications: Energy efficient service and device discovery radio," in *Proc. Wireless VITAE 2011*, Chennai, India, Feb. 2011.
- [12] F. Liu and G. Heijenk, "Context discovery using attenuated Bloom filters in ad-hoc networks," in *Proc. WWIC 2006*, Bern, Switzerland, May 2006.
- [13] P. Goering and G. Heijenk, "Service discovery using bloom filters," in *Proc. ASCI 2006*, Lommel, Belgium, Jun. 2006.

- [14] F. Sailhan and V. Issarny, "Service discovery using olsr and bloom filters," in Proc. OLSR Interop & Workshop 2008, Ottawa, Canada, Oct. 2008.
- [15] G. Koloniari and E. Georgia, "Filters for XMLbased service discovery in pervasive computing," The Computer Journal, vol. 47, no. 4, pp. 461–474, Apr. 2004.
- [16] S. Cheng, C. K. Chang, and L.-J. Zhang, "An efficient service discovery algorithm for counting Bloom filter-based service registry," in Proc. IEEE ICWS 2009.
- [17] O. R. Helgason, E. A. Yavuz, S. T. Kouyoumdjieva, L. Pajevic, and G. Karlsson, "A mobile peer-to-peer system for opportunistic content centric networking," in Proc. ACM Mobi Held 2010, New Delhi, India, Aug. 2010.