An Information Sharing System for HUNETS without the use of Internet

Sheetal. S. R
M. Tech 4th sem Dept. of CSE
T. John Institute Technology
Bangalore, India

Mrs. Salonee Mishra
Asst. Professor, Dept. of CSE
T. John Institute of Technology
Bangalore, India

***Abstract*— The mobile devices that exist today make use of the wireless infrastructure to access Internet services provided by application providers. This architecture is inefficient in many situations. So here in this paper we discuss about human network (HUNET), a network architecture allows information sharing between mobile devices through direct inter device communication. B-SUB is an interest-driven information sharing system for HUNETs. In B-SUB, the user describes the content and user interests in the form of tags. These tags are human-readable strings. The tag-based content description method is found to be very effective. To facilitate efficient data dissemination, Temporal Counting Bloom filter (TCBF) is used that encodes tags, which also reduces the overhead of content routing. Theoretical analyses on B-SUB are presented and verified B-SUB works efficiently under various network topological conditions.**

***Key Words* —Content-based publish/subscribe, interest-driven information sharing, human network, bloom filter.**

I. INTRODUCTION

The rapid development of wireless technology has made the mobile devices an indispensable part of our lives. Existing wireless networking technologies allow mobile devices to communicate with each other through wireless infrastructures, for example, GSM, 3G, LTE, and so on. This architecture, however, is not ubiquitously applicable everywhere. First, it fails in many situations due to limited network resources. For example, in a conference room, the WiFi and cellular connection can be crippled because too many users are competing for the channel simultaneously. Second, this architecture [12] does not take advantage of the abundant interdevice communication opportunities available. Again, take the conference room scenario as an example because of the high density of wireless devices, there can be excellent wireless connections between nearby mobile devices. Existing wireless networks are unable to utilize such communication opportunities. As a result, this architecture fails to address novel application requirements. Nowadays, most mobile applications [13] are for information sharing; mobile devices are increasingly becoming the end points of information consuming. Evidence is that almost all existing smart phones and tablets are integrated with vendor-supplied music/video streaming services, and social-network-based information sharing services are extremely popular on mobile devices. Given the existing architecture, however, they have to connect with central service providers, which would fail in many situations as described above. Besides, this architecture can be inefficient in many scenarios. For instance, location-based chatting is more natural to implement in a peer-to-peer manner, so that nearby users can talk to each other directly.

Recently, a new architecture of networking portable wireless devices has emerged, which is called the delay tolerant networks (DTNs) [1]. DTNs adopt a “store-carry and-forward” model, which significantly expands the communication capability of mobile device. Driven by the new application demands and the limitations of the existing architecture, we envision a new type of dynamic networking service called human networks (HUNETs). Physically, a HUNET is composed of human-carried mobile devices, which have the same structure as DTNs. These devices use short-range wireless communication technologies, such as WiFi or BlueTooth, to communicate with each other. Functionally, HUNETs enable information sharing between users in a completely decentralized manner without the aid of an wireless communication infrastructure. A high-level illustration of this architecture is presented in Fig. 1. The figure shows a HUNET composed of four users, each of which carries a mobile device. Users share information they are interested in with nearby peers through direct interdevice wireless communication.

B-SUB is an interest-driven information sharing system for HUNETs, which stands for the bloom-filter-based publish/SUBscribe. B-SUB is designed for small to medium sized networks composed of dozens of devices restricted in a limited physical area where interdevice communication opportunities are abundant. Typical application scenarios are researchers inside a conference room, students inside a department building, visitors in a recreation center, and so on. The distinctive features of B-SUB are as follows: First, B-SUB employs content-based networking [2], [3] to achieve infrastructureless communication. B-SUB routes and forward messages based on their content instead of addresses, which enables autonomous access to interested information for users without an end-to-end addressing mechanism. Second, B-SUB is much more efficient than traditional content-based publish/subscribe. Mobile devices have weak processors and are powered by batteries. Their computational capability is rather limited. Additionally, the memory capacity and bandwidth of the nodes in a HUNET are also scarce. Traditional content-based networking systems [4], however, are complex and consume excessive memory and bandwidth.

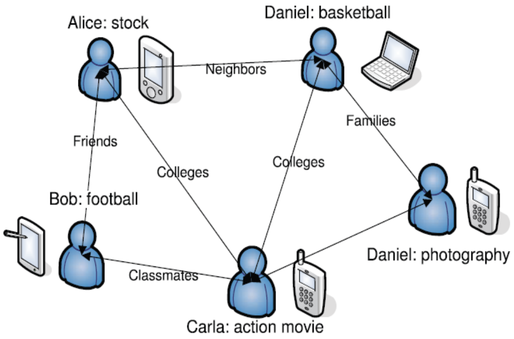


Fig. 1. A high-level illustration of the HUNET. Every user is equipped with a mobile device. Every user has his/her own interests, which is represented as a *name:interest* pair. Messages are forwarded between users via (multihop) store-carry-forward guided by the user interests.

B-SUB uses a tag-based content description model and uses Bloom filters [5] to compress content and user interests. We invent the Temporal Counting Bloom filter (TCBF), an extension of the Bloom filter, to encode tags, which achieves efficient content routing. However, the TCBF has false positives in their queries, which causes useless messages to be forwarded to nodes that are not really interested in their content. We analyze, in theory, several parameters that are related to the false positive probability of the TCBF and their impacts on B-SUB's performance. The analysis is verified through extensive simulation studies. To summarize, our contributions in this paper are as follows:
. We propose HUNET, a novel network architecture that facilitates efficient information sharing between portable mobile devices.
. We design B-SUB, an interest-driven information sharing system for HUNETs, a content-based publish/subscribe that achieves infrastructure-less communication between mobile devices.
. We invent the TCBF, an extension to the counting Bloom filter.
. We conduct extensive theoretical analyses and real world trace-driven simulations to evaluate the performance of B-SUB.

RELATED WORK

Trade-offs among certain computational factors in hash coding are analyzed. The paradigm problem considered is that of testing a series of messages one-by-one for membership in a given set of messages. Two new hashcoding methods are examined and compared with a particular conventional hash-coding method. The computational factors considered are the size of the hash area (space), the time required to identify a message as a nonmember of the given set (reject time), and an allowable error frequency.

In order to cope with the explosive demands and limited capacity provided by the current cellular networks, Delay Tolerant Networking (DTN) is used to migrate traffic from the cellular networks to the free and high capacity device-to-device networks. The current DTN-based mobile data offloading models do not address the heterogeneity of mobile traffic and are based on simple network assumptions. A mathematical framework to study the problem of multiple mobile data offloading under realistic network assumptions is established, where

1) Mobile data is heterogeneous in terms of size and lifetime,

2) Mobile users have different data subscribing interests, and

3) The storage of offloading helpers is limited.

The components of a loosely coupled system are typically designed to operate by generating and responding to asynchronous events. An event notification service is an application-independent infrastructure that supports the construction of event-based systems, whereby generators of events publish event notifications to the infrastructure and consumers of events subscribe with the infrastructure to receive relevant notifications. The two primary services that should be provided to components by the infrastructure are notification selection (i.e., determining which notifications match which subscriptions) and notification delivery (i.e., routing matching notifications from publishers to subscribers). Numerous event notification services have been developed for local-area networks, generally based on a centralized server to select and deliver event notifications.

The designers of communication networks are being challenged by the emergence of a new class of addressing and routing scheme referred to as content-based addressing and routing. This new approach differs from traditional unicast and multicast schemes in that it performs routing based on the data being transported in a message rather than on any specialized addressing and routing information attached to, or otherwise associated with, the message.

The aim of the paper is threefold. First we point out the common denominators of publish/subscribe schemes: time, space, and synchronization decoupling of subscribers and publishers. These decoupling dimensions are illustrated by comparing the publish/subscribe paradigm with "traditional" interaction schemes. Second, we compare the many variants of publish/subscribe schemes: namely, topicbased, content-based, and type-based.

This paper presents an algorithm for content-based forwarding, an essential function in content-based networking. Unlike in traditional address-based unicast or multicast networks, where messages are given explicit destination addresses,the movement of messages through a content-based network is driven by predicates applied to the content of the messages. Forwarding in such a network amounts to evaluating the predicates stored in a router's forwarding table in order to decide to which neighbor routers the message should be sent. We are interested in finding a forwarding algorithm that can make this decision as quickly as possible in situations where there are numerous, complex predicates and high volumes of messages.

Studying transfer opportunities between wireless devices carried by humans, we observe that the distribution of the inter-contact time, that is the time gap separating two contacts of the same pair of devices, exhibits a heavy tail such as one of a power law, over a large range of value. This observation is confirmed on six distinct experimental data sets. It is at odds with the exponential decay implied by most mobility models.

The highly successful architecture and supporting protocols of today's Internet operate poorly when faced with operating environments characterized by very long delay paths and frequent network partitions. These problems are exacerbated by end nodes that have severe power or memory constraints. Often deployed in mobile and extreme environments lacking "always-on" infrastructure, many such networks have their own specialized protocols, and do not utilize IP.

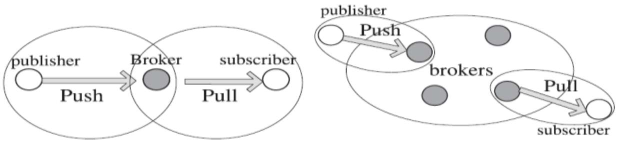
Due to uncertainty in nodal mobility, DTN routing usually employs multi-copy forwarding schemes. To avoid the cost associated with flooding, much effort has been focused on probabilistic forwarding, which aims to reduce the cost of forwarding while retaining a high performance rate by forwarding messages only to nodes that have high delivery probabilities. This paper aims to provide an optimal forwarding protocol which maximizes the expected delivery rate while satisfying a certain constant on the number of forwarding per message. In our proposed optimal probabilistic forwarding (OPF) protocol, we use an optimal probabilistic forwarding metric derived by modeling each forwarding as an optimal stopping rule problem.

Greedy Distance Vector (GDV) is the first geographic routing protocol designed to optimize end-to-end path costs using any additive routing metric, such as: hop count, latency, ETX, ETT, etc. GDV requires no node location information. Instead, GDV uses estimated routing costs to destinations which are locally computed from node positions in a virtual space. GDV makes use of VPoD, a new virtual positioning protocol for wireless networks.

II. THE ARCHITECTURE OF HUNET

The main aim of HUNET, is to facilitate efficient information sharing between humans using mobile devices. A HUNET is composed of portable devices that are equipped with wireless communication interfaces, like WiFi or BlueTooth [9]. Constrained by the relatively weak capability, these devices can only do short-range communication. Advanced wireless communication technologies, like directional antenna [6] could be used to expand the communication range. These devices are always carried and operated by human users, which gives the name of Human Network. They are collectively referred to as nodes in this paper. It is desirable to have non-human-operated devices to serve as "hot spots" or "offloading stations" [7] to boost the performance, but is not mandatory. In this paper, we assume that there are no such devices in HUNETs. The processing power of HUNET nodes is weak. Additionally, because battery is the sole energy source, efficiency becomes even more crucial in HUNET, which is an important driving factor of B-SUB's design. Users join a HUNET for sharing information that they receive from others or gather from elsewhere. The most important characteristic of HUNET compared to DTNs is that HUNET exclusively relies on peer-to-peer communication to do forwarding. DTNs, on the contrary, still focus on delivering messages from a source to a destination, although there are generally no end-to-end paths connecting them.

Although HUNETs embody the same network structure as DTNs, DTN routing protocols cannot be directly applied because: 1) DTNs do not support interest-driven communication; 2) DTN routing is based on the end-to-end model, which is not applicable in HUNETs because the information source is unaware of the users who are interested in the information; 3) many existing DTN routing protocols [8] require complex offline processing to achieve optimal performance, which is prohibitive in HUNETs because they consume excessive resources and the needed data are usually impossible to get. B-SUB overcomes these problems by employing content-based publish/subscribe to facilitate infrastructure-less communication in HUNETs, and relies on users' interests to guide content routing.



(a) Traditional networks (b) HUNETs/DTNs

Fig. 2. High-level illustrations of the pub/sub system in traditional networks and HUNETs/DTNs. In traditional networks, A central broker connects publishers with subscribers. However, the "central broker" abstraction is no longer valid in HUNETs, or is too expensive to maintain.

Fig. 2 depicts the concept of a publish/subscribe system. Fig. 2a shows the architecture of the traditional publish/ subscribe (pub/sub) systems, where a central broker connects message producers (publishers) with consumers (subscribers). Fig. 2b depicts the pub/sub system in HUNETs. A swarm of nodes form a mobile broker network. Multiple nodes serve as brokers to carry messages for users. They are distributed and are constantly changing connections due to mobility. These properties make it difficult to implement an efficient pub/sub system in HUNETs. We can still logically treat a swarm of brokers as a central broker, but this abstraction does not provide insights about the structure of HUNETs, nor does it improve the efficiency of information sharing. We take a radically different approach in designing B-SUB [14] to address the unique requirements of HUNETs. B-SUB exploits the peer-to-peer communication pattern in HUNETs, and lets all users exchange their interests during random contacts. Messages are then forwarded to interested users by following the trails of interest propagation.

III. TEMPORAL COUNTING BLOOM FILTER

A. *Design of the Temporal Counting Bloom Filter*

The TCBF , an extension of the counting Bloom filter works similar to a counting Bloom filter. A TCBF also uses a vector of counters. Insertion of the TCBF increments the associated counters of the inserted key by a fixed value II, called the initial counter value (ICV), instead of 1 in the counting Bloom filter. Each time a key is inserted into a TCBF [3], the counters associated with the key's hashed bits will be set to the ICV. If the counter has already been set by some other keys, we do not change its value. In other words, the results of insertions are always a TCBF with multiple counters of the same value of as that of II.

There are two ways of merging multiple TCBFs: the additive merging or A-merge and the maximum merging [21] or M-merge. In the A-merge, as shown in Fig. 3, the counters are set to the sum of the counters of the original filters. In the M-merge, as shown in Fig. 4, the values of the new filter's counters are set as the maximum value of the counters of the original filters. The intentions of these two merging operations will be clear after we present the design of B-SUB.

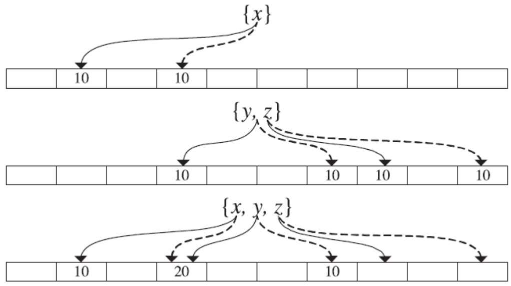


Fig. 3. Illustration of the A-merge of the TCBF. The fourth counter is the sum of the two original TCBFs.

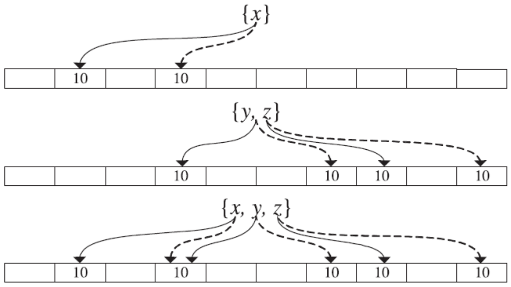


Fig. 4. Illustration of the M-merge of the TCBF. The fourth counter is the maximum of the two original TCBFs.

We can only insert a key into a filter that has never been merged before. To insert multiple keys into a merged filter, we first insert the keys into an empty TCBF; then, we merge the two TCBFs using A-/M-merge. Note that merging is different from union, the purpose of this design will be clear after introducing the concept of decay. The reasons behind forbidding the insertion from increasing counters' values, and to distinguish between A and M-merge, are related to the semantics of content processing, which will be elaborated later.

A unique operation of the TCBF is decay [13]. The TCBF does not use deletion directly. It supports temporal deletion through decay. The decay of a TCBF is to constantly decrement its counters' values with a rate given by the decay factor (DF) [20]. Equivalently, the decay cycle is the time for decrementing the counters' value by 1. The following equation holds for these two parameters: decay cycle = 1/decayfactor. Tuning DFs and ICVs produces a wide range of decay granularity that are used to represent different frequencies of insertion, A-merge, and M-Merge, which is the key to controlling the performance of B-SUB [15]. The operation is illustrated in Fig. 5, where several keys are put into the filter at different times. The final values of the filter's counters represent the frequency of the keys being inserted into the filter. After 19 delay units of time, the only keys left in the filter are (y,w). If a key is not inserted into the filter frequently enough (by direct insertion or merging with another filter), then it will be removed from the filter eventually [17]. The DF directly determines the scope of the interest propagation and, in turn, the performance of the message forwarding of B-SUB.

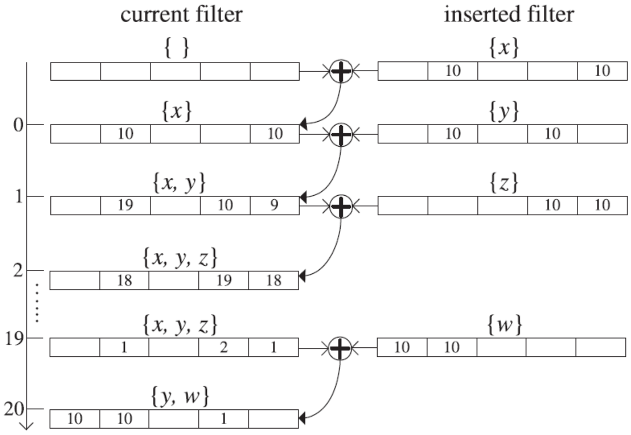


Fig. 5. An illustration of the decay operation of the TCBF. This TCBF uses a vector of five counters and two hash functions. The decay factor is set to 1/unit time.

The query that checks if a key is in a TCBF is called the existential query (E-query) [19] [19]. TCBFs have the same FPR as the Bloom filters for the E-queries. A difference is that a TCBF's FPR tends to decrease with the time because keys are gradually removed by decay. Another type of query of the TCBF is called the preferential query (P-query) [16]. P-query is done as follows: For a key k and two TCBFs, Fi and Fj, we get the values of the counters associated with k in Fi and Fj, which are two sets, Ci and Cj. Then, we obtain the minimum values of Ci and Cj, which is denoted as ci and cj. We define the preference of Fj to Fi against k, PREF (I,j,k), as follows:

$$PREF_{i,j}\{k\} = \begin{cases} \frac{c_j - c_i}{c_i} & \text{if } c_i \neq 0 \\ c_j & \text{if } c_i = 0 \end{cases}$$

B. *Benefits of using the TCBF*

TCBFs are used to store interests. Basically, TCBFs map [2], through k hash functions, each interest into k bits of the bit-vector. It reduces the space consumption of storing users' interests. The simulation results demonstrates that the TCBF uses half of the space used by the raw strings with an acceptable FPR. It also reduces the bandwidth consumption of the interest propagation. When two interests, I0 and I1, of a user cannot be forwarded at the same time due to the bandwidth restriction in the contact, without the TCBF, only one of them gets the opportunity to be served. If we can compress both I0 and I1 to fit the bandwidth, both interests might eventually be served. The TCBF does just that: in any situation, a bit-vector of a fixed length is enough to store multiple interests [16]. Besides, content matching using TCBF is also more efficient than string matching.

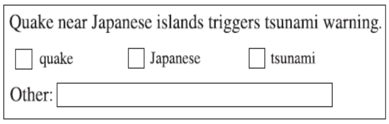
IV. BSUB DESIGN

A. Overview

B-SUB has two components: content representation and pub/ sub routing. B-SUB employs the tag-based content description model. The contents of messages and the interests of users are identified by tags, which are strings that summarize the topics of the message. They are stored in TCBFs, which are then used as probabilistic hints for forwarding messages. The pub/sub routing provisions two functions: interest propagation and message forwarding. Both rely on the TCBF to achieve low storage and computational complexities. B-SUB limits the size of messages to a few more than 100 bytes. Large volume content distribution is surly desirable, but is difficult to provision given the existing infrastructure. It is also true in existing social networking applications that users tend to publish many small-sized messages. For example, Twitter [9], a popular microblogging application, limits the maximum size of each post to 140 bytes.

B. Tag-Based Content Description

The tag-based content description model is used in B-SUB. To justify its effectiveness and applicability, we conducted an experiment: users are asked to choose appropriate tags to summarize the content of the given news titles. As depicted in Fig. 6, given a news title, different users may choose distinct tags. In order for the tag-based approach to work, different users shall have a common view, i.e., same tags, for the same message, i.e., a news title in this experiment. We define consensus to measure a user's ability to find the common tags [13]. It is calculated as follows: for the tag that is chosen by the most users, denote N as the number of users who chose that tag. The ratio of N to the total number of users is the value of the consensus of this news title. The higher the consensus of a message, the more likely its tags match those of other users who are interested in the message. A consensus of 1 indicates that all users have at least one common tag that describes the content of the news title. For example, if a message has a very low consensus, this means that every user has a different tag for it [20]. If it is labeled as "blue" by user A, and "red" by user B, and user B is interested in "red," then user A should send the message to user B because its label applied by B is "red," which is the same as his/her own interest. However, user A will not forward the message because user A applies a different tag to the message and determines that user B is not interested in it.



Quake near Japanese islands triggers tsunami warning.

□ quake □ Japanese □ tsunami

Other:

Fig. 6. A sample question used in the experiment of assessing the applicability of the tag-based content description model.

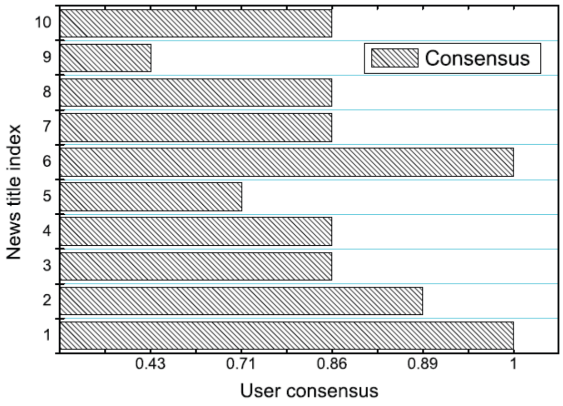


Fig. 7. The user consensus of 10 different news titles.

Fig. 7 shows the consensus of 10 news titles, all of which were collected from 20 people randomly selected from the authors' email contact lists. In the figure, most titles have high consensus values that are larger than 0.86. Only one has a relatively low value of 0.43. Judging from the fact that the targeted application scenario of B-SUB [17] is to have casual chats about popular topics, the consensus should be better than this experiment; alas, we deem that it suffices to describe contents and interests with the tag-based content description model.

C. Interest Propogation

In B-SUB, TCBFs are used to compress users' interests. A user stores its own interests in a TCBF, which is called the genuine filter. A broker [21] stores the interests collected from other users in another TCBF called the relay filter. TCBFs serve as a "compressed" matching hint for delivery, but they are not precise, due to the TCBF's false positives. The false positives of the TCBF causes interests being falsely matched by a message, so that the message will be forwarded to nodes that are not really interested in it. The message is sent according to the guidance of the TCBFs stored in all of the nodes in the network

D. Routing and Forwarding

The decay of the TCBF removes, from the nodes' relay filters, interests from the consumers that they meet infrequently. The preferential query is used by nodes to select forwarders for the buffer messages. The only data that is needed in the forwarding of messages is the TCBFs that encode the interests. The only operations performed are hashing [5] and table lookup [9].

V. CONCLUSION AND FUTURE ENHANCEMENT

Finally we presented B-SUB, an information sharing system for HUNETs. B-SUB employs content-based networking to help infrastructure-less communication between mobile devices in the locality. Specifically, all the BSUB employs a tag-based content description model. A novel data structure, called as the TCBF, is also invented to compress user interests and guide content routing. The use of TCBF reduces the memory and bandwidth consumption of B-SUB significantly. We systematically analyze the impact of several parameters of B-SUB on its behaviors and for performance comparison. An extension of B-SUB called B SUB-P is also proposed to better protect user privacy efficiently.

This paper has its limitation in the mode of transfer. That is it allows only single point message transfer mode. Only one person in a group can send message at one point of time. The other has to wait until the first person finishes sending message. This could be overcome by use of ad-hoc networks.

REFERENCES

[1] Space/Time Trade-offs in Hash Coding with Allowable Errors, BURTON H. BLOOM,1970
[2] Multiple Mobile Data Offloading Through Delay Tolerant Networks, Yong Liy, Guolong Suy, Pan Huiz, Depeng Jiny, Li Suy, Lieguang Zeng, 2011.
[3] Design and Evaluation of a Wide-Area Event Notification Service, ANTONIO CARZANIGA,DAVID S. ROSENBLUM,ALEXANDER L. WOLF, 2001.
[4] Content-Based Addressing and Routing: A General Model and its Application, ANTONIO CARZANIGA,DAVID S. ROSENBLUM,ALEXANDER L. WOLF, 2000.
[5] The Many Faces of Publish/Subscribe, PATRICK TH. EUGSTER,PASCAL A. FELBER,RACHID GUERRAOUI,ANNE-MARIE KERMARREC, 2003.
[6] Forwarding in a Content-Based Network, Antonio Carzaniga,Alexander L. Wolf, 2003.
[7] Impact of Human Mobility on the Design of Opportunistic Forwarding Algorithms, Augustin Chaintreau, Pan Hui, Jon Crowcroft, Christophe Diot, Richard Gass, and James Scott, 2007.
[8] A Delay-Tolerant Network Architecture for Challenged Internets, Kevin Fall, 2003.
[9] An Optimal Probabilistic Forwarding Protocol in Delay Tolerant Networks, Cong Liu and Jie Wu, 2009.
[10] Greedy Distance Vector Routing, Chen Qian and Simon S. Lam, 2011.
[11] J. Scott, R. Gass, J. Crowcroft, P. Hui, C. Diot, and A. Chaintreau, "CRAWDAD Data Set Cambridge/Haggle (v. 2009-05-29)," http://crawdad.cs.dartmouth.edu/cambridge/haggle, May 2009.
[12] N. Eagle and A.S. Pentland, "CRAWDAD Data Set Mit/Reality , Jul. 2005.
[13] A. Chaintreau P. Hui, J. Crowcroft, C. Diot, R. Gass, and J. Scott, "Pocket Switched Networks: Real-World Mobility and Its Consequences for Opportunistic Forwarding," IEEE J. Selected Areas Comm., vol. 26, no. 5, pp. 748-760, Feb. 2005.
[14] A. Chaintreau, P. Hui, C. Diot, R. Gass, and J. Scott, "Impact of Human Mobility on Opportunistic Forwarding Algorithms," IEEE Trans. Mobile Computing, vol. 6, no. 6, pp. 606-620, June 2007.
[15] C. Qian and S. Lam, "Greedy Distance Vector Routing," Proc. 31st Int'l Conf. Distributed Computing Systems (ICDCS '11), pp. 857-868, June 2011.
[16] S.S. Lam and C. Qian, "Geographic Routing in D-Dimensional Spaces with Guaranteed Delivery and Low Stretch," Proc. ACM SIGMETRICS Joint Int'l Conf. Measurement and Modeling of Computer Systems (SIGMETRICS '11), pp. 257-268, 2011.
[17] Y. Zhao and J. Wu, "B-Sub: A Practical Bloom-Filter-Based Publish-Subscribe System for Human Networks," Proc. Int'l Conf. Distributed Computing Systems (ICDCS '10), pp. 634-643, 2010.
[18] R. Zhang, Y. Zhang, and Y. Fang, "AOS: an Anonymous Overlay System for Mobile Ad Hoc Networks," Wireless Networks, vol. 17, no. 4, pp. 843-859, May 2011.
[19] R. Zhang, J. Shi, Y. Zhang, and J. Sun, "Secure Cooperative Data Storage and Query Processing in Unattended Tiered Sensor Networks," Selected Areas in Comm., IEEE J., vol. 30, no. 2, pp. 433-441, Feb. 2012.
[20] R. Zhang, J. Shi, and Y. Zhang, "Secure Multidimensional Range Queries in Sensor Networks," Proc. ACM MobiHoc '09, pp. 197-206, 2009.