Analysis of a Hypercube-Based Social Feature Multipath Routing in Delay Tolerant Networks

Yunsheng Wang, *Student Member*, *IEEE*, Wei-Shih Yang, and Jie Wu, *Fellow*, *IEEE*

Abstract—Social behavior plays a more and more important role in delay tolerant networks (DTNs). In this paper, we present an analytical model for a hypercube-based social feature multipath routing protocol in DTNs. In this routing protocol, we use the internal social features of each node (individual) in the network for routing guidance. This approach is motivated from several real social contact networks, which show that people contact each other more when they have more social features in common. This routing scheme converts a routing problem in a highly mobile and unstructured contact space (M-space) to a static and structured feature space (F-space). The multipath routing process is a hypercube-based feature matching process where the social feature differences are resolved step-by-step. A feature matching shortcut algorithm for fast searching is presented where more than one feature difference is resolved at one time. The multiple paths for the routing process are node-disjoint. We formally analyze the delivery rate and latency by using hypercube-based routing. The solutions for the expected values of latency and delivery rate are given under different path conditions: single-/multipath and feature difference resolutions with/without shortcuts. Extensive simulations on both real and synthetic traces are conducted in comparison to several existing state-of-the-art DTN routing protocols.

Index Terms—Delay tolerant networks, delivery rate, hypercubes, latency, multipath routing, node-disjoint paths, social features

1 Introduction

Delay tolerant networks (DTNs), known for their intermittent connectivity and high node mobility, can allow for much-needed connectivity and other settings with limited or nonexisting infrastructures. There is no end-to-end path between some or all of the nodes in DTNs, which makes routing quite different from other types of wireless networks.

Mobile phones are popular with more than 5.6 billion phones in use worldwide [1]. Smartphones with programmable capability are a growing fraction of these phones. As stated in Nielson's report: "as of Q4 2011, 46 percent of US mobile consumers had smartphones, and that figure is growing quickly. In fact, 60 percent of those who said they got a new device within the last three months chose a smartphone over a feature phone" [2]. Mobile social network [3] is a new type of DTNs in which social features play an important role and where individuals move around and interact at each contact based on their common interests through smartphones.

Most of the social-behavior-based DTN routing schemes [4], [5], [6], [7], [8], [9], [10], [11], [12] that have been proposed recently do not consider the real *social features* of each node (individual). These social features are considered to be very important—namely, in that *people come into contact more frequently if they have more social features in common* [13]. We obtain the number of contacts between pairwise individuals in different feature distances (the number of different features) from Infocom 2006 trace and MIT reality mining data in Fig. 1. It shows the total contact times among the individuals. Fig. 1 also shows that the individuals with a smaller feature distance come into contact more often. In this paper, we use the feature extraction method from data mining [14], [15] to obtain the most informative features, including affiliation, country, language, and so on, for routing guidance.

In most of the state-of-art DTN routing schemes, the routing problem considers in a highly mobile and unstructured contact space (M-space). In hypercube-based social feature multipath routing [16], the routing scheme converts the routing problem from the M-space to a feature space (F-space), which is static and structured, as shown in Fig. 2. In Fig. 2b, nodes with the same feature (same color in Fig. 2a) are grouped together. Using internal social features for routing guidance can prevent a long-term information collection process. Social features are static and easy to obtain; hence, we consider social feature information to be more reliable. More specifically, each node (individual) is represented by a social profile, which is composed of a vector $(F_1, F_2, \ldots, F_m)$, where each feature F_i has n_i distinct values for $1 \le i \le m$. In this way, the F-space contains $n_1 \times n_2 \times \cdots \times n_m$ groups. The groups in the F-space can be mapped into an *m-dimensional hypercube*, in which two groups are connected if and only if they differ in only one feature.

We first review the efficient hypercube-based social feature matching process in [16]. Feature differences are resolved step-by-step until the destination is reached, which is a node-disjoint multipath routing scheme according to the property of hypercube. The shortcut is introduced for fast feature matching, where more than one feature difference

- Y. Wang and J. Wu are with the Department of Computer and Information Sciences, Temple University, 1805 N. Broad Street, 324 Wachman Hall, Philadelphia, PA 19122. E-mail: {yunsheng.wang, jiewu}@temple.edu.
- W.-S. Yang is with the Department of Mathematics, Temple University, Philadelphia, PA 19122. E-mail: yang@temple.edu.

Manuscript received 16 May 2012; revised 19 Aug. 2012; accepted 10 Sept. 2012; published online 21 Sept. 2012.
Recommended for acceptance by D. Wang.
For information on obtaining reprints of this article, please send e-mail to: tpds@computer.org, and reference IEEECS Log Number TPDS-2012-05-0469.
Digital Object Identifier no. 10.1109/TPDS.2012.281.

1045-9219/13/$31.00 © 2013 IEEE Published by the IEEE Computer Society

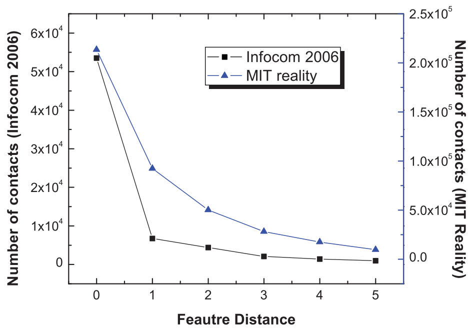

Fig. 1. Comparison of contacts in real traces: *Infocom 2006* and *MIT reality mining*.

can be resolved at one time. The shortcut still can ensure the node-disjointness of these multiple parallel paths.

In this paper, we formally analyze the performance of the proposed hypercube-based social feature routing by looking at the delivery rate and latency. First, we study the shortcut case. Recursive formulas are presented in both multipath and single-path cases. Then, we show the benefit of using multipath routing. If no shortcut is used, the exact solutions for the expected values of latency and delivery rate are given for different routing schemes, with both single- and multipath routing schemes. The analysis results show that using the feature matching shortcut can increase the delivery rate and reduce the latency.

In the simulation, we first verify our analysis results in synthetic and real traces. Then, we compare hypercube-based social feature multipath routing with state-of-the-art routing protocols: spray-and-wait (S-W) [8] and spray-and-focus (S-F) [9], both in synthetic and real traces.

The major contributions of our work are as follows:

- We formally analyze the delivery rate and latency by using hypercube-based routing in DTNs.
- We make comparisons between the multipath and single-path cases.
- We show the efficiency of the shortcut-based fast feature matching.
- We evaluate the proposed scheme in both synthetic and real traces.
- The simulation results demonstrate the competitive performance of multipath routing in DTNs and confirm the correctness of our analysis results.

The remainder of this paper is organized as follows: Section 2 reviews related work. Section 3 shows the preliminary work. Section 4 introduces the social feature extraction mechanism. Section 5 describes the details of hypercube-based social feature routing. Section 6 formally analyzes the proposed routing protocol. Section 7 focuses on the simulation and evaluation. We conclude our work in Section 8.

2 Related Work

Recently, DTN research received a lot of attention in multiple fields: routing [3], [4], [5], [6], [7], [8], [9], [10], [11], [12], [17], [18], [19], resource allocation [20], [21], [22], network analysis [23], [24], and so on. The simplest DTN routing scheme is epidemic routing [17]. To reduce the overhead of DTN routing, Lee et al. introduced two-hop routing [18], where the source gives a copy to relay nodes, each of which holds the packet until it contacts the destination. In [8] and [9], two multicopy routing schemes, S-W and S-F, are proposed. S-W always halves the number of copies at each spray phase; it allows for multihop unless the current node has one copy left. S-F goes further to allow multihop, even when there is one copy left. Delegation forwarding [19] uses a quality metric to guide the multihop process, which can further reduce the overhead.

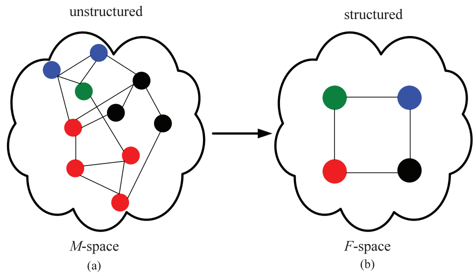

Fig. 2. Converting from M-space to F-space.

Recently a number of DTN routing protocols have attempted to enhance packet delivery by using social contact information [3], [4], [5], [6], [7], [8], [9], [10], [11], [12], [25]. Daly and Haahr analyzed the social network characterizing information for routing in DTNs [12]. Their designed SimBet routing scheme is based on a node's egocentric betweenness centrality and a node's social similarity. In [7], Hui et al. used human mobility in terms of social structures in the design of forwarding algorithms for pocket switched networks. The authors designed BUBBLE Rap: social-based forwarding routing in DTN by exploring heterogeneous of human interaction. In [10], Mtibaa et al. developed PeopleRank social opportunistic forwarding routing scheme, in which nodes are ranked using tunable weighted social information, which is similar to the PageRank idea. Gao et al. exploited node centrality and social community structures, and designed multicast protocol in DTNs [11]. However, the betweenness centrality, social similarity, and social community structures are expensive to obtain, especially in a dynamic network such as DTNs. In this paper, the internal social feature information is used for routing guidance, which can avoid a global information collection process.

Initially, the applications of the hypercube have been studied in parallel and distributed computing [26], [27]. There has been some recent work done on hypercube routing in wireless networks [28], [29], [30]. Our approach utilizes the advantage of hypercube properties—the node-disjointness of the multiple paths. Our proposed shortcut fast feature matching process, which improves the efficiency of routing, can still guarantee the node-disjointness. Note that such a property is absent in existing state-of-the-art DTN routing protocols.

3 Preliminaries

In this section, we first present the objectives of our proposed routing protocol. Then, we briefly introduce the

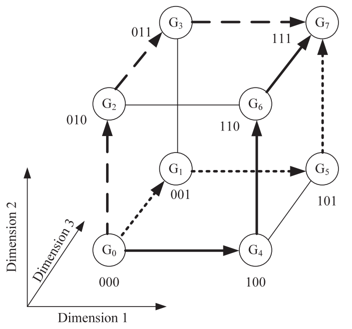

Fig. 3. A 3D hypercube.

system model. The concepts of social feature space and hypercube are presented in Sections 3.3 and 3.4.

3.1 Objectives

The objective of this paper is to develop an analytical model for the efficient multipath routing scheme based on hypercube social feature matching in DTNs. Two performance metrics are used to measure the routing performance: 1) *delivery rate*: the average delivery ratio of the routing packet; 2) *latency*: the average duration between a packet's generation and the arrival time.

3.2 System Model

We model a DTN as a set of mobile nodes (individuals). Two nodes transfer packets to each other when they are within each other's communication range. Here, we assume that all of the nodes have the same communication range. Assume that there are N individuals in the whole network. Each individual can be represented by his/her social features. The social features represent either physical properties, such as gender, or logical ones, such as a membership in a social group.

3.3 Social Feature Space

We convert the mobile and unstructured contact space (M-space) with N individuals into a static and structured feature space (F-space) with M nodes (or groups). Each individual belongs to one of the groups in the F-space. Fig. 3 illustrates a 3D F-space with eight groups. In this example, there are three different social features in the F-space, represented by two distinct values. Dimension 1 (the left most position) corresponds to *Language*: English (0) or Chinese (1); dimension 2 (the second left most position) shows *position*: professor (0) or student (1); and dimension 3 represents *gender*: male (0) or female (1). In Fig. 3, two groups have a connection if they differ in exactly one feature.

3.4 Hypercube

Given the above definition of the F-space, we can represent the fact that individuals with the same social features can form groups, and each group is a node in a hypercube. More specifically, the F-space is mapped into an m-dimensional hypercube, which consists of $\prod_{i=1}^{m} n_i$ nodes, where n_i denotes the number of distinct values in dimension i. Two nodes, $A : \{a_1, a_2, \dots, a_m\}$ and $B : \{b_1, b_2, \dots, b_m\}$, in an m-dimensional hypercube are connected if and only if their social features differ in exactly one dimension (say i, such that $a_i \neq b_i$) [26]. The feature distance (or feature difference) (H_{AB}) is the measure of closeness between two individuals and is used to express the virtual similarity between individuals in a hypercube. In this paper, we focus on the binary hypercube where $n_i = 2$ for all i, as shown in Fig. 3.

TABLE 1
Entropy of the Social Features in Real Traces

Infocom Feature	Entropy	MIT Feature	Entropy
$Affiliation$	4.64	*Neighborhood*	4.13
$City$	4.45	*Daily commute*	4.05
$Nationality$	4.11	*Hangouts*	3.86
$Language$	4.11	*Working hour*	3.54
$Country$	3.59	*Affiliation*	2.56
$Position$	1.37	*Research group*	1.53

4 Feature Extraction

The individuals are characterized by a high-dimensional feature profile. However, usually only a small subset of features is important. We use the feature extraction method from data mining [14], [15] to obtain key features.

There are N individuals with m' features, which are denoted as $F_1, F_2, \dots, F_{m'}$. The goal of our social feature extraction is to extract the *most informative subset* with $m (< m')$ key features. We use Shannon entropy [31], which quantifies the expected value of the information contained in the feature, to select the key features: $E(F_i) = -\sum_{k=1}^{n} p(x_k) log_2 p(x_k),\ (i = 1, 2, \dots, m')$, where $E(F_i)$ denotes the entropy of the feature F_i, and p denotes the probability mass function of F_i. $\{x_1, \dots, x_n\}$ are the possible values of feature F_i. The entropy of the feature considers not only the number of possible values, but also the distribution of their frequencies.

Table 1 shows the entropy of each social feature that we obtained from the Infocom 2006 trace [32]: $m = 6$ most informative features out of $m' = 10$ total features. It also lists $m = 6$ most informative features from the MIT reality mining data.

5 Hypercube-Based Social Feature Multipath Routing

We present a novel hypercube-based social feature routing scheme. It is a multipath routing scheme with the objective of reaching the destination quickly, while maximizing the delivery rate. The number of copies of the packet in the whole routing process can be controlled in multipath routing. In other words, the overhead is constant. The main objective is to distribute the copies of a packet in a cost-effective way.

The copies of the packet are distributed to multiple node-disjoint paths to resolve the feature distance between the source and destination. When a packet holder comes in contact with another individual with a smaller feature distance to the destination, the packet will be forwarded to the encountered individual. The new holder will continue seeking the relay node until the packet enters the destination group.

5.1 Basic Hypercube Routing

We assume that the source and destination differ in k dimensions $\{1, 2, \ldots, k\}$, denoted as a set, C. We define the *coordinate sequence* $C^0 : \langle 1, 2, \ldots, k\rangle$ from a given C as the basic routing path. C^0 can be any permutation of C. C^0 determines how the multiple node-disjoint paths are constructed, based on the resolution order of dimension differences. C^i is defined as i circular left shifts of C^0. $C^0, C^1, \ldots, C^{k-1}$ will create k node-disjoint shortest paths:

- *Path 1* — generated by C^0: $\langle 1, 2, 3, \ldots, k\rangle$;
- *Path 2* — generated by C^1: $\langle 2, 3, 4, \ldots, k, 1\rangle$;
- *Path 3* — generated by C^2: $\langle 3, 4, 5, \ldots, k, 1, 2\rangle$;

  ......
- *Path k*—generated by C^{k-1}: $\langle k, 1, 2, \ldots, k-2, k-1\rangle$.

Here, the path generated from source S by coordinate sequence C^0 follows a matching process along dimension 1 to dimension k. In Fig. 3, source G_0 and destination G_7 differ in three dimensions $\{1, 2, 3\}$. The shortest node-disjoint paths are as follows:

- *Path 1* with sequence $\langle 1, 2, 3\rangle$ is (G_0, G_4, G_6, G_7);
- *Path 2* with sequence $\langle 2, 3, 1\rangle$ is (G_0, G_2, G_3, G_7);
- *Path 3* with sequence $\langle 3, 1, 2\rangle$ is (G_0, G_1, G_5, G_7).

In hypercube routing, the coordinate sequence of a path is sent along with the packet. After a successful forwarding along dimension i, dimension i will be deleted from the sequence. Finally, the sequence becomes empty upon reaching the destination group. Algorithm 1 shows basic hypercube routing until the packet enters the destination group.

Algorithm 1. Hypercube-based Routing
1: $/*$ When A with a packet to D encounters B. $*/$
2: **if** B and D are in the same group **then**
3: Forward the packet to B.
4: **else**
5: **if** $H_{BD} \leq H_{AD}$ **then**
6: The forwarding is based on the sequence in the k multiple paths.

When the packet reaches the destination group, we propose two schemes to guide the packet to the destination node. One is *wait*, that the packet holder only forward the packet to the destination node. This scheme can reduce the number of forwardings at the cost of the increased latency. Another is $focus$, that the packet holder only forward the packet to an individual in the same group with a higher active level, and will not retain the packet itself. We introduce another metric: *active level*, which is used to measure the social activities of the individuals in their group. This is the number of times that the individual comes into contact with other individuals in the same group.

5.2 Hypercube Routing with Shortcuts

In basic hypercube routing, each step can only resolve in one dimension at one time. Some chances to shorten the distance to the destination will be lost when the packet holder meets another individual who is more than one feature distance away, but is closer to the destination. Therefore, we propose the *feature matching shortcut* to enable fast searching. To ensure a reduction of the distance to the destination, the packet can jump to another group that is more than one feature difference away. Such a controlled jump is called a shortcut, which is a *prefix*[1] of the coordinate sequence.

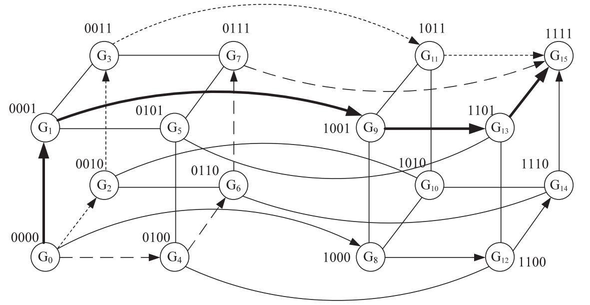

Fig. 4. An example of hypercube-based routing from G_0 to G_{15}, where the directed lines are the multiple node-disjoint paths.

According to the hypercube property [26], these multiple node-disjoint paths are composed of k shortest paths of length k when the source and the destination differ in k dimensions in an m-dimensional hypercube. All of these paths are generated from the coordinate sequence. Since each shortcut is a prefix of a coordinate sequence, all resultant paths are still node-disjoint. The benefit of the node-disjointness is that the paths will not cross each other, which increases the efficiency of the routing.

5.3 An Example

As shown in Fig. 4, one individual in G_0 has a packet for another individual in G_{15}. The source and the destination differ in four feature dimensions; hence, there are four shortest paths: $(G_0, G_8, G_{12}, G_{14}, G_{15})$, $(G_0, G_4, G_6, G_7, G_{15})$, $(G_0, G_2, G_3, G_{11}, G_{15})$, and $(G_0, G_1, G_9, G_{13}, G_{15})$. These paths are node-disjoint.

The shortcuts can happen at (G_0, G_{12}), (G_0, G_6), (G_0, G_3), (G_0, G_9), (G_8, G_{14}), (G_{12}, G_{15}), (G_4, G_7), (G_6, G_{15}), (G_2, G_{11}), (G_3, G_{15}), (G_1, G_{13}), and (G_9, G_{15}), which are two-hop shortcuts; (G_0, G_{14}), (G_0, G_7), (G_0, G_{11}), (G_0, G_{13}), (G_8, G_{15}), (G_4, G_{15}), (G_2, G_{15}), and (G_1, G_{15}), which are three-hop shortcuts; and (G_0, G_{15}), which is a four-hop shortcut. All of these shortcuts still guarantee the node-disjointness.

6 Analysis

In this section, we formally analyze the delivery rates and latencies of the hypercube-based feature matching process. Our formulas of latency are valid for any contact time distributions. We note that the contact time is typically assumed to be exponentially distributed in many studies, but recent empirical results show that, in some cases, it follows a power-law distribution; see, for example, [33]. Since our formulas are valid for any contact distributions, our results may be useful for further studies.

In Section 6.1, we first obtain a recursive formula for the latency in routing with shortcuts of all distances included. The recursive formula then allows for an efficient way of calculating the delivery rates and the latencies in the whole graph, with shortcuts of all distances included. Then, we

1. $\langle 1, 2, \ldots, k'\rangle$ is a *prefix* of $\langle 1, 2, \ldots, k\rangle$, where $k' \leq k$.

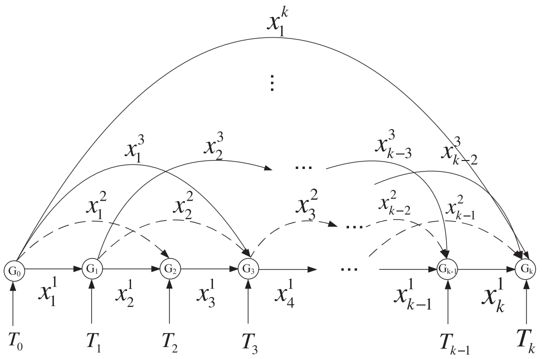

Fig. 5. An illustration of the shortcut fast feature matching process.

will compare the delivery rates and latencies of routing where *shortcuts of all distances*[2] are included, with the case of routing with only neighboring groups[3] (routing without shortcuts) under the single-path condition. In Section 6.2, we extend our recursive formula to the multipath case.

In Section 6.3, for the case of routing without any short cuts, we have an analytic expression for the delivery rate and the expected latency for single and multiple paths. These expressions are also valid for any type of contact time distribution. For the case of exponential contact distribution, we obtain an exact formula for the delivery rate and the expected latency. Our results also show that the multipath routing scheme is more efficient than the single-path routing.

6.1 Single-Path Routing with Shortcuts

In this section, we first obtain a recursive formula for the latency in routing with shortcuts of all distances included. The recursive formula then allows for an efficient way of calculating the delivery rates and the latencies in the whole graph, with short-cuts of all distances included. Then, we will compare the performances of the shortcut and nonshortcut in the single-path scenario.

From Fig. 5, we assume that the source group and destination group is in k feature distances. Let $G_0, G_1, G_2, \ldots, G_k$ be the nodes of a single path of total feature distance k, where the feature distance between two adjacent nodes G_{j-1} and G_j is equal to 1, for all $j = 1, \ldots, k$. For $0 \leq i \leq k$, let x_j^i be the minimum arrival time of the packet sent directly from G_{j-1} to G_{j-1+i}, $j = 1, 2, \ldots, k+1-i$. For the following recursive formula of latency, x_j^i can be any distribution.

Theorem 1. *In a single-path routing with shortcuts of all distances included, let $G_0, G_1, G_2, \ldots, G_k$ be the nodes of a single path of total feature distance k, where the feature distance between two adjacent nodes G_{j-1} and G_j is equal to 1, for all $j = 1, \ldots, k$. For $0 \leq i \leq k$, let x_j^i be the minimum arrival time of a packet sent directly from G_{j-1} to G_{j-1+i}, $j = 1, 2, \ldots, k+1-i$. Let T_i be the latency of the packet arriving at group G_i, whose feature distance to the source group G_0 is i. Then, the following recursive relations hold:*

2. *Shortcuts of all distances* are the shortcuts in any hops.
3. The individuals that are in the same node within the hypercube are considered as a group. The neighboring groups are groups with exactly 1 feature distance.

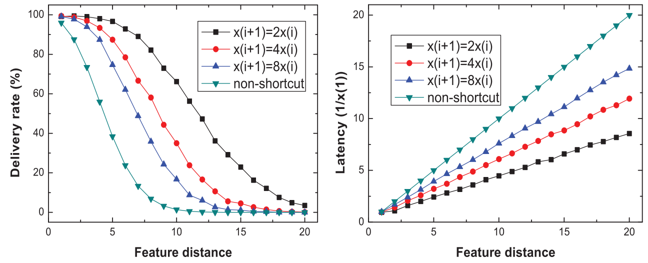

Fig. 6. Comparison between nonshortcut and two-hop shortcuts including in the single-path case. ($x(i)$ is x_j^i for $\forall j$.)

$$\begin{aligned} T_0 =\ & 0, \\ T_1 =\ & x_1^1, \\ T_2 =\ & min\big((T_1 + x_2^1), (T_0 + x_1^2)\big), \\ T_3 =\ & min\big((T_2 + x_3^1), (T_1 + x_2^2), (T_0 + x_1^3)\big), \\ & \ldots, \\ T_k =\ & min\big((T_{k-1} + x_k^1), (T_{k-2} + x_{k-1}^2), \ldots, (T_0 + x_1^k)\big). \end{aligned} \tag{1}$$

Proof. We will prove by induction. From Fig. 5, we have

$$T_0 = 0, T_1 = x_1^1.$$

Assuming that (1) holds for all $i \leq k$, we will prove that it holds for $k+1$. We note that, when routing for a packet to arrive in G_{k+1}, it can first arrive in node G_i, then take a shortcut of distance $k+1-i$ whose time is x_{i+1}^{k+1-i}. If G_i is the last node that the packet visited before arriving at G_{k+1}, then the minimum delivery time for taking this route from G_0 to G_{k+1} is $T_i + x_{i+1}^{k+1-i}$. Therefore, the overall minimum delivery time from G_0 to G_{k+1} is

$$T_{k+1} = min\big((T_k + x_{k+1}^1), (T_{k-1} + x_k^2), \ldots, (T_0 + x_1^{k+1})\big).$$

Therefore, (1) holds for $k+1$. □

The above recursive relations allow for an efficient way of calculating the expected latency of all shortcuts included. In what follows, we will give examples where x_j^i's are independent and exponentially distributed. We will compare the results between the cases where all shortcuts are included, and the case where only two-hop shortcuts are included.

First, if only two-hop shortcuts exist, then $x_j^i = \infty$ for all $i \geq 3$ and all j. By the recursive relations, we have

$$\begin{aligned} T_2 &= min\big((T_1 + x_2^1), (T_0 + x_1^2)\big), \\ T_3 &= min\big((T_2 + x_3^1), (T_0 + x_2^2)\big), \\ &\ldots, \\ T_k &= min\big((T_{k-1} + x_k^1), (T_{k-2} + x_{k-1}^2)\big). \end{aligned} \tag{2}$$

Since individuals with more features in common will contact each other more, we assume that $x_j^{i+1} = a * x_j^i$ and that a is equal to 2, 4, and 8. Based on the above recursive relations, we can calculate the delivery rate and latency, as shown in Fig. 6. If all shortcuts are included, T_k can also be recursively calculated. Fig. 7 shows the comparison results. From both Figs. 6 and 7, we find that using shortcuts can significantly improve the performance. We also compare two-hop shortcuts including, two- and three-hop shortcuts including, and all shortcuts including, which is shown in Fig. 8.

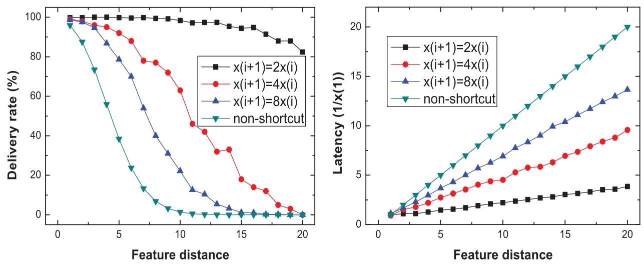

Fig. 7. Comparison between nonshortcut and all shortcuts including in the single-path case.

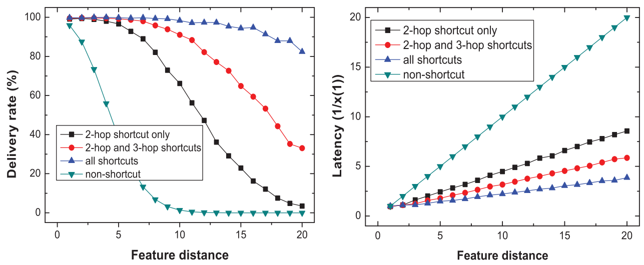

Fig. 8. Comparison of the performance after including different numbers of hop shortcuts.

6.2 Multipath Routing with Shortcuts

In what follows, we give a recursive formula for multipath routing with all shortcuts of all distances included. In this case, we consider the case where the source group G_0 and destination group G_k are in k feature distance. Since there are exactly k node-disjoint shortest paths, for $1 \leq l \leq k$, we let $G_{l0}, G_{l1}, G_{l2}, \dots, G_{lk}$ be the nodes of the lth path of total feature distance k that connects G_0 and G_k, where the feature distance between two adjacent nodes $G_{l(j-1)}$ and G_{lj} is equal to 1, for all $j = 1, \dots, k$. For $0 \leq i \leq k$, let x^i_{lj} be the minimum arrival time of the packet sent directly from $G_{l(j-1)}$ to $G_{l(j-1+i)}$, $j = 1, 2, \dots, k+1-i$. For our recursive formula of latency, x^i_{lj} can be any distribution. Using a similar argument as the one in the proof of Theorem 1, we have the following.

Theorem 2. *In a multipath routing with shortcuts of all distances included, let T_{li} be the latency of the packet arriving at group G_{li} whose feature distance to the source group G_{l0} is i in the path l, and let T'_{li} be the latency of the packet arriving at group G_{li} in the path l without using the longest shortcut x^k_{l1}. Let $S_{r,k}$ be the latency of the packet sent from the group G_0 to arrive at group G_k, using r shortest paths $l = 1, 2, \dots, r$. Then the following recursive relations hold:*

$$\begin{aligned} T'_{lk} &= min\big((T_{l(k-1)} + x^1_{lk}), (T_{l(k-2)} + x^2_{l(k-1)}), \dots, \\ &\quad (T_{l1} + x^{k-1}_{l1})\big), \\ S_{r,k} &= min\big(T'_{1k}, \dots, T'_{rk}, x^k_{11}\big). \end{aligned} \tag{3}$$

where the longest shortcuts in all paths are the same; i.e., $x^k_{l1} = x^k_{11}$.

A numerical study for $S_{r,k}$ is shown in Fig. 9. As shown in Table 2, using shortcut fast feature matching can improve the performance.

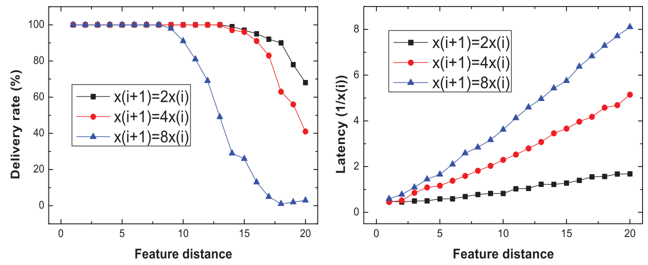

Fig. 9. Comparison between nonshortcut and all shortcuts including in the multipath case.

TABLE 2
Comparison of Shortcut and Nonshortcut in the Multipath Case

	Feature distance	2	3	4	5
Delivery rate	*Non-shortcut*	96%	93%	83%	64%
	Shortcut	100%	100%	100%	100%
Latency ($\frac{1}{\lambda_1}$)	*Non-shortcut*	1.25	1.68	2.18	2.72
	Shortcut	0.46	0.50	0.51	0.59

6.3 Routing without Shortcuts: Multipath versus Single Path

In this section, we consider the case of routing without using shortcuts, where each pair of neighboring groups meet according to an independently identical distribution f. We give a general formula for the delivery rate and latency of the multiple paths routing scheme. In case f is an exponential density function with mean $\frac{1}{\lambda}$, we have an analytical expression for the delivery rate and the latency. We note that if the feature distance between the source and the destination is k, then there are exactly k shortest paths between the source and the destination of length k. In the following theorem, we find the delivery rate if r shortest paths are used in the routing scheme.

Theorem 3. *Suppose the contact times for all pairs with feature distance 1 are independent with the same probability density function f. For the multipath routing scheme, if r shortest paths are used, then the following hold:*

(a) the delivery rate from the source group to the destination, with k feature distance in time t, is given by

$$1 - \left[\int_t^\infty \int_0^{x_k} \int_0^{x_{k-1}} \dots \int_0^{x_1} f(x_k - x_{k-1}) f(x_{k-1} - x_{k-2}) \dots f(x_2 - x_1) f(x_1) dx_1 \dots dx_k\right]^r; \tag{4}$$

(b) the expected latency from the source group to the destination, with k feature distance, is given by

$$\int_0^\infty \left[\int_t^\infty \int_0^{x_k} \int_0^{x_{k-1}} \dots \int_0^{x_1} f(x_k - x_{k-1}) f(x_{k-1} - x_{k-2}) \dots f(x_2 - x_1) f(x_1) dx_1 \dots dx_k\right]^r dt. \tag{5}$$

Proof. (a) Since the multiple paths from the source to the destination are node-disjoint, the multiple paths are independent. By assumption, they also have the same distribution. Therefore, if r shortest paths are used, then the probability that the destination group has not received the packet by time t is

$$P(\tau_r > t) = \prod_{j=1}^{r} P(\tau^j > t), \tag{6}$$

where τ_r is the delivery time when r paths are used and τ^j is the delivery time when only jth path is used.

Since the length of the shortest paths in k feature distance is k, and all neighboring groups meet according to the same distribution with density function f, the delivery time in k feature distance is a sum of k independent random variables with common density function f, and its probability density function is given by the convolution of k copies of f. Therefore, the probability that for each one of the multiple paths fail in time t is

$$P(\tau^j > t) = \int_t^{\infty} h_k(x)dx, \tag{7}$$

where

$$\begin{aligned} h_k(x) = \int_0^{x} \int_0^{x_{k-1}} \dots \int_0^{x_2} f(x - x_{k-1}) \dots \\ f(x_2 - x_1) f(x_1) dx_1 \dots dx_{k-1}. \end{aligned} \tag{8}$$

Therefore, (6) can be represented as follows:

$$P(\tau_r > t) = \prod_{j=1}^{r} P(\tau^j > t) = \left[\int_t^{\infty} h_k(x)dx\right]^r. \tag{9}$$

Therefore, the delivery rate in time t is

$$\begin{aligned} P(\tau_r \le t) &= 1 - P(\tau_r > t) \\ &= 1 - \left[\int_t^{\infty} \int_0^{x_k} \dots \int_0^{x_2} f(x_k - x_{k-1}) \dots \right. \\ &\quad \left. f(x_2 - x_1) f(x_1) dx_1 \dots dx_k \right]^r. \end{aligned} \tag{10}$$

(b) Since the expected value of a nonnegative random variable is equal to the integral of its failing rate at time t with respect to t, we have (b). □

In the case, where f is exponentially distributed with mean $\frac{1}{\lambda}$, we have the following analytical expressions for the delivery rate and latency.

Theorem 4. *For the multipath routing scheme, if r shortest paths are used, then the delivery rate from the source group to the destination, with k feature distance in time t, is*

$$1 - \left[\int_t^{\infty} \frac{\lambda^k x^{k-1} e^{-\lambda x}}{(k-1)!} dx\right]^r. \tag{11}$$

Proof. If f is an exponential density function with mean $\frac{1}{\lambda}$, then $h_k(x)$ is a gamma distribution

$$g_{k,\lambda}(x) = \frac{\lambda^k x^{k-1} e^{-\lambda x}}{(k-1)!}.$$

Therefore, the theorem follows from Theorem 3 (a). □

In the proposed hypercube-based routing scheme, there are k multiple node-disjoint shortest paths from the source to the destination, which differ in k feature dimensions. Hence, r is equal to k in the hypercube-based routing. It follows from Theorem 11 that for the case of exponential distribution, the delivery rate is $1 - [\int_t^{\infty} \frac{\lambda^k x^{k-1} e^{-\lambda x}}{(k-1)!} dx]^k$.

Theorem 5. *For the multipath routing scheme, if r shortest paths are used, then the expected latency from the source group to the destination, with k feature distance, is $\frac{c_{r,k}}{\lambda}$, where*

$$c_{r,k} = \frac{(rk)!}{[(k-1)!]^r} \int_1^{\infty} \dots \int_1^{\infty} \frac{u_1^{k-1} \dots u_r^{k-1}}{(u_1 + \dots + u_r)^{rk+1}} du_1 \dots du_r.$$

The proof of Theorem 5 is given in the supplementary section, which can be found on the Computer Society Digital Library at http://doi.ieeecomputersociety.org/10.1109/TPDS.2012.281.

7 Simulation

First, we verify the analysis results in Matlab using both real and synthetic traces in the following categories:

1. *Comparison between multipath and single-path schemes.* Both schemes do not use the shortcut.
2. *Comparison between nonshortcut and shortcut schemes.* Both schemes in this category do not consider the process after the packet arrives at the destination group.

Then, we compare the performance of the proposed hypercube-based social feature routing scheme with several state-of-the-art ones, including S-W [8] and S-F [9].

7.1 Simulation Data Sets

In the simulation, we make comparisons of the performance of the routing schemes, both in real and synthetic traces.

7.1.1 Real Trace

We use two real traces, the *Infocom 2006* trace [5], [32] and the *MIT reality mining* data [34], in our simulation.

The Infocom 2006 data set consists of two parts: *contacts* between the iMote devices carried by participants and *social features* of the participants, which are the statistics of participants' information from a questionnaire form. First, we discard some participants that do not have social features in their profiles. In this way, we reduce the number of participants to 61. There are 74,981 contacts between these participants over a period of 337,418 time slots in seconds. We extract five social features from the original data set: nationality, language, affiliation, position, and country.

The MIT reality mining data set also consists of two parts: *contacts* and *social features*. After discarding some participants with no complete input information, we reduce the number of participants to 57. There are 411,313 contacts between these participants over a period of 897,921 time slots in seconds. We extract five social features from the original data set: daily commute, research group, affiliation, neighborhood, and working hour.

7.1.2 Synthetic Trace

We assume that the two groups in i feature distances meet according to a uniform exponential distribution with mean time $\frac{1}{\lambda_i}$. Since people contact each other more frequently if

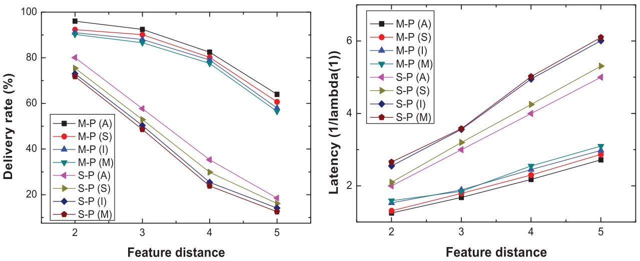

Fig. 10. Comparison of the performance between *multipath* and *single path*: (L): delivery rate and (R): latency. (*M-P*: multipath, *S-P*: single-path; A: analysis, S: synthetic, I: Infocom, and M: MIT.)

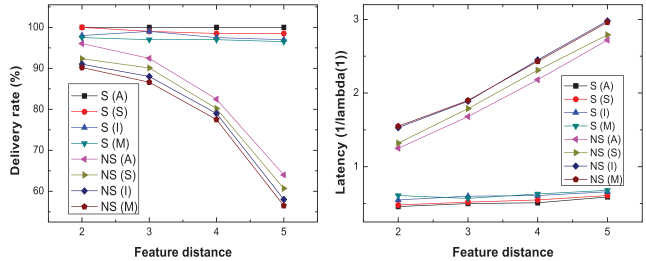

Fig. 11. Comparison of the performance between *shortcut* and *nonshortcut*: (L): delivery rate and (R): latency. (S: shortcut, *NS*: nonshortcut.)

they have more social features in common, we assume that $\lambda_i = 2 * \lambda_{i+1}$, according to Fig. 1. The contact rate between individuals in the same group (λ_0) is $2 * \lambda_1$. The contact rate of pairwise individuals corresponds to the properties of their belonging groups. We create 64 individuals and 50,000 time slots in seconds. Contacts are randomly selected from these time slots based on their contact rate.

The estimation of λ_1 is according to the real traces. For example, in the case of a 3D hypercube, as shown in Fig. 3, there are 12 edges. If the contact rate of each edge e_i is λ_{e_i}, the estimated λ_1 in the synthetic trace is

$$\frac{1}{\lambda_1} = \frac{1}{12}\sum_{i=1}^{12}\frac{1}{\lambda_{e_i}}. \qquad (12)$$

In the simulation, we will compare two performance metrics: delivery rate and latency.

7.2 Simulation Methods

We implement and compare several routing schemes in the simulation. In all schemes, we consider the following:

1. *Hypercube-based social feature routing with wait-at-destination (HSFR-W).* Waiting for the destination after the packet enters the destination group in hypercube-based social feature routing.
2. *Hypercube-based social feature routing with focus-at-destination (HSFR-F).* Forwarding the packet to higher active level nodes after the packet enters the destination group in hypercube-based social feature routing.
3. *Spray-and-wait (S-W) [8]. Spray* phase—any node with copies will forward half of the copies to the encountered node with no copy; *Wait* phase—if the destination is not found in the spray phase, the copy carriers wait for the destination.
4. *Spray-and-focus (S-F) [9]. Spray* phase—same as S-W; *Focus* phase—if the destination is not found in the spray phase, the copy carriers forward the copy to the encountered node with a smaller feature distance to the destination.
5. *SimBet [12].* The packet forwarding is based on the SimBet utility, which is a combination of locally determined social *similarity* to the destination node and pre-estimated *betweenness* centrality metrics.

The first two schemes use hypercube routing, in which the multiple paths are node-disjoint. In the other three schemes, the multiple paths may cross each other, which may reduce the efficiency of the routing process.

7.3 Simulation Results

7.3.1 Comparison of Multipath and Single-Path Schemes

In this section, we will show the benefits of the multipath scheme. From Fig. 10, we find that both analysis and simulation results show that multipath routing has a higher delivery rate and a smaller latency compared to the single-path scheme. Multipath routing increases delivery rate by about 115 percent, and decreases latency by about 45 percent. The simulation results are consistent with the analysis results. The difference between the analysis and synthetic simulation results is about 3 percent in the delivery rate and 5 percent in the latency. The results from the real traces reduce the delivery rate by about 6 percent, and increase the latency by about 15 percent compared to the analysis results. There are two reasons that the real trace simulation results are a little bit different from the analysis results. First, the estimated contact rates (λ) are not very accurate, because we assume that the pairwise contact rates are all the same in the analysis part; however, in the real trace, they are different. Second, in the simulation part, there are multiple nodes in each group, but we assume that there is only one node in one group in our analysis.

7.3.2 Comparison of Shortcut and Nonshortcut Schemes

Our simulation demonstrates the importance of the shortcut fast feature matching in hypercube-based routing. Fig. 11 shows the performance of the shortcut and nonshortcut schemes in the proposed hypercube routing. Both analysis and simulation results show that using the shortcut fast feature matching process can significantly increase the delivery rate and reduce the latency. In the synthetic trace, using the shortcut can increase the delivery rate by about 15 percent, and reduce the latency by about 60 percent. When the feature distance increases, the improvement also increases. The results in the real traces show the same phenomenon. The simulation results are consistent with the analysis results, especially in the synthetic trace. From the synthetic trace results, we see that it decreases the delivery rate by about 2 percent, and increases the latency by about 4 percent. The results from two real traces show that it decreases the delivery rate by about 5 percent, and increases the latency by about 15 percent compared to the analysis results.

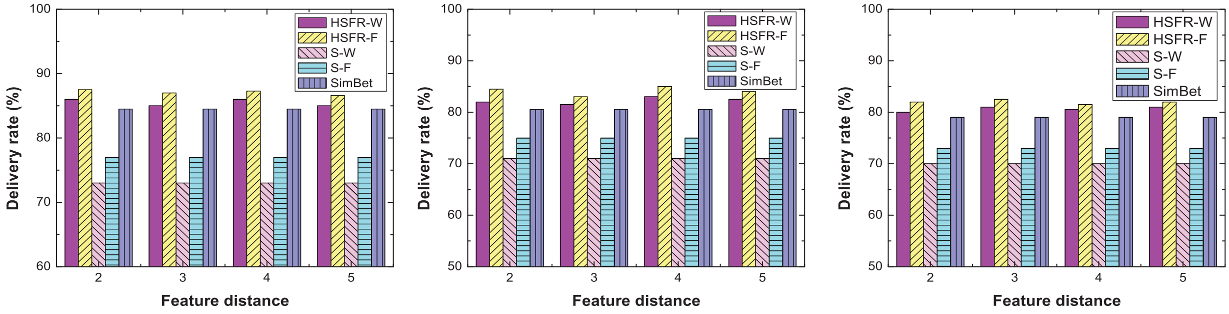

Fig. 12. Comparing the delivery rate: (L): synthetic, ($Center$): Infocom, and (R): MIT.

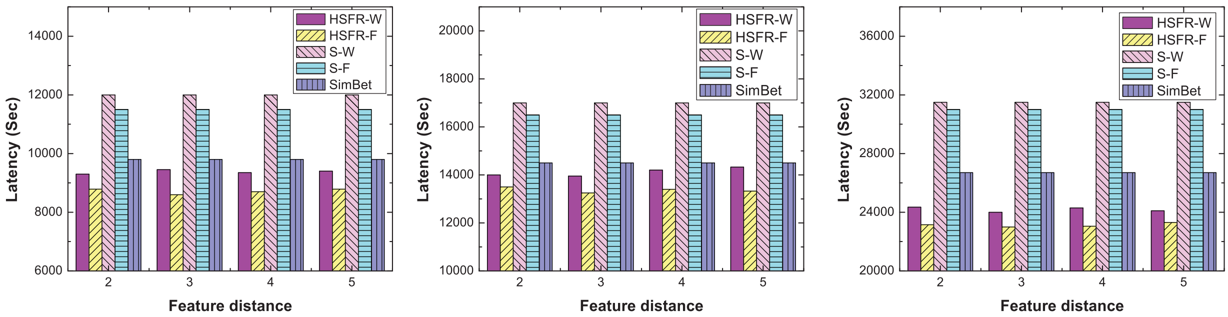

Fig. 13. Comparing the latency: (L): synthetic, ($Center$): Infocom, and (R): MIT.

7.3.3 *Comparison with Other State-of-the-Art Routing Protocols*

We compare the performance of hypercube-based routing with S-W, S-F, and seek-active in both synthetic and real traces. From Figs. 12 and 13, we find that hypercube-based routing performs better than other schemes. In the synthetic trace, HSFR-W increases the delivery rate by about 17 percent and reduces the latency by about 22 percent, compared to S-W. HSFR-F increases the delivery rate by about 14 percent and reduces the latency by about 24 percent, compared to S-F. In real traces, the proposed hypercube-based social feature increases the delivery rate by about 14 and 13 percent, respectively, in the Infocom and MIT reality mining traces, compared to S-W and S-F. The latency can be reduced by about 18 and 24 percent, respectively. In both synthetic and real traces, hypercube-based routing has a higher delivery rate and smaller latency compared to the SimBet scheme. After the packet enters the destination group, using $focus$ can improve the performance, as compared to using $wait$. From Figs. 12 and 13, we can see that HSFR-F increases the delivery rate by about 2 percent in the synthetic trace, 3 percent in the Infocom trace, and 2.5 percent in the MIT reality mining trace compared to HSFR-W. HSFR-F reduces the latency by about 7 percent in the synthetic trace, 5 percent in the Infocom trace, and 4 percent in the MIT reality mining trace compared to HSFR-W, in Figs. 12 and 13.

7.4 Summary of Simulation

We first verify the analysis results in the simulation. Then, comparisons with other state-of-the-art DTN routing protocols are presented, both in synthetic and real traces.

By comparing the simulation results to the analysis results, we find the consistency between these two. Because of the estimation deviation for the contact rate, the simulation results reduce the delivery rate, and increase the latency by a little. Another reason the latency is a little bit different in the simulation results is because, in simulation, there are multiple nodes in one group, which is different from the analysis assumption.

Our simulation concludes that, compared with S-W, S-F, and SimBet schemes, the hypercube-based social feature routing scheme has a significantly higher delivery rate and reduced latency. The hypercube-based routing is a multipath routing scheme, in which there are multiple node-disjoint paths seeking the destination that helps to improve search efficiency. The shortcut fast feature matching process can increase the delivery rate and reduce the latency. When the packet enters the destination group, forwarding the packet to the active relay nodes can improve the performance, as compared to waiting for the destination node.

8 Conclusion

In this paper, we provide an analytical model for a hypercube-based social feature routing scheme in DTNs, which converts the routing problem from a high mobility space (M-space) into a static feature space (F-space). Hypercube-based routing is a feature matching process where the feature difference between the source and the destination is resolved step-by-step. The shortcut algorithm is used for fast feature matching, where the feature distance can be reduced more than one at one time. The shortcut also can guarantee the node-disjointness of these multiple paths, which can improve the efficiency of the routing process. In Section 6, we formally analyze the delivery rate and latency of the hypercube-based routing. Exact solutions are presented in multipath and single-path cases when there is no shortcut. If shortcuts are included, recursive formulas are introduced for both multipath and single-path cases. We prove that multipath scheme has better performance than single-path scheme. Shortcut can increase the delivery rate and reduce the latency. Trace-driven simulation results show that our proposed hypercube-based routing scheme performs better than the S-W, S-F, and SimBet schemes. The simulation results also verify the accuracy of our analysis results. We believe that the social features will play an important role in DTN routing.

References

[1] H.M.A. There, "How Many Mobile Phone Users in the World," http://www.howmanyarethere.org/how-many-mobile-phone-users-in-the-world/, Apr. 2012.

[2] Nielsen, "More US Consumers Choosing Smartphones as Apple Closes the Gap on Android," http://blog.nielsen.com/nielsenwire/consumer/, Feb. 2012.

[3] J. Scott, J. Crowcroft, P. Hui, and C. Diot, "Haggle: A Networking Architecture Designed Around Mobile Users," *Proc. Third Ann. IFIP Conf. Wireless On-Demand Network Systems and Services (WONS),* 2006.

[4] J. Burgess, B. Gallagher, D. Jensen, and B.N. Levine, "Maxprop: Routing for Vehicle-Based Disruption-Tolerant Networks," *Proc. IEEE INFOCOM,* 2006.

[5] A. Chaintreau, P. Hui, J. Crowcroft, 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.

[6] J. Wu and Y. Wang, "A Non-Replication Multicasting Scheme in Delay Tolerant Networks," *Proc. IEEE Seventh Int'l Conf. Mobile Adhoc and Sensor Systems (MASS),* pp. 89-98, 2010.

[7] P. Hui, J. Crowcroft, and E. Yoneki, "Bubble Rap: Social-Based Forwarding in Delay Tolerant Networks," *Proc. ACM MobiHoc,* pp. 241-250, 2008.

[8] T. Spyropoulos, K. Psounis, and C.S. Raghavendra, "Spray and Wait: An Efficient Routing Scheme for Intermittently Connected Mobile Networks," *Proc. ACM SIGCOMM Workshop Delay-Tolerant Networking (WDTN),* pp. 252-259, 2005.

[9] T. Spyropoulos, K. Psounis, and C.S. Raghavendra, "Spray and Focus: Efficient Mobility-Assisted Routing for Heterogeneous and Correlated Mobility," *Proc. IEEE Fifth Int'l Conf. Pervasive Computing and Comm. Workshop (PERCOMW),* pp. 79-85, 2007.

[10] A. Mtibaa, M. May, C. Diot, and M. Ammar, "Peoplerank: Social Opportunistic Forwarding," *Proc. IEEE INFOCOM,* pp. 111-115, 2010.

[11] W. Gao, Q. Li, B. Zhao, and G. Cao, "Multicasting in Delay Tolerant Networks: A Social Network Perspective," *Proc. ACM MobiHoc,* pp. 299-308, 2009.

[12] E.M. Daly and M. Haahr, "Social Network Analysis for Routing in Disconnected Delay-Tolerant Manets," *Proc. ACM MobiHoc,* pp. 32-40, 2007.

[13] A. Mei, G. Morabito, P. Santi, and J. Stefa, "Social-Aware Stateless Forwarding in Pocket Switched Networks," *Proc. IEEE INFOCOM,* pp. 251-255, 2011.

[14] S. Jain, K. Fall, and R. Patra, "Routing in a Delay Tolerant Network," *Proc. ACM SIGCOMM,* pp. 145-158, 2004.

[15] I. Guyon and A. Elisseeff, "An Introduction to Variable and Feature Selection," *J. Machine Learning Research,* vol. 3, pp. 1157-1182, 2003.

[16] J. Wu and Y. Wang, "Social Feature-Based Multi-Path Routing in Delay Tolerant Networks," *Proc. IEEE INFOCOM,* 2012.

[17] A. Vahdat and D. Becker, "Epidemic Routing for Partially Connected Ad Hoc Networks," technical report, Dept. of Computer Science, Duke Univ., 2000.

[18] U. Lee, S.Y. Oh, K.-W. Lee, and M. Gerla, "Relaycast: Scalable Multicast Routing in Delay Tolerant Networks," *Proc. IEEE Int'l Conf. Network Protocols (ICNP),* pp. 218-227, 2008.

[19] V. Erramilli, M. Crovella, A. Chaintreau, and C. Diot, "Delegation Forwarding," *Proc. ACM MobiHoc,* pp. 251-260, 2008.

[20] A. Balasubramanian, B. Levine, and A. Venkataramani, "DTN Routing as a Resource Allocation Problem," *Proc. ACM SIGCOMM,* pp. 373-384, 2007.

[21] M. Seligman, K. Fall, and P. Mundur, "Alternative Custodians for Congestion Control in Delay Tolerant Networks," *Proc. ACM SIGCOMM Workshop Challenged Networks (CHANTS),* pp. 229-236, 2006.

[22] K. Lee, Y. Yi, J. Jeong, H. Won, I. Rhee, and S. Chong, "Max-Contribution: On Optimal Resource Allocation in Delay Tolerant Networks," *Proc. IEEE INFOCOM,* 2010.

[23] V. Vukadinovic and S. Mangold, "Opportunistic Wireless Communication in Theme Parks: A Study of Visitors Mobility," *Proc. ACM SIGCOMM Workshop Challenged Networks (CHANTS),* pp. 3-8, 2011.

[24] P. Nikolopoulos, T. Papadimitriou, P. Pantazopoulos, M. Karaliopoulos, and I. Stavrakakis, "How Much Off-Center are Centrality Metrics for Routing in Opportunistic Networks," *Proc. ACM SIGCOMM Workshop Challenged Networks (CHANTS),* pp. 9-14, 2011.

[25] G.S. Thakur, A. Helmy, and W.-J. Hsu, "Similarity Analysis and Modeling in Mobile Societies: The Missing Link," *Proc. ACM SIGCOMM Workshop Challenged Networks (CHANTS),* pp. 13-20, 2010.

[26] Y. Saad and M. Schultz, "Topological Properties of Hypercubes," *IEEE Trans. Computers,* vol. 37, no. 7, pp. 867-872, July 1988.

[27] P. Ramanathan and K. Shin, "Reliable Broadcast in Hypercube Multicomputers," *IEEE Trans. Computers,* vol. 37, no. 12, pp. 1654-1657, Dec. 1988.

[28] R. Manoharan and P. Thambidurai, "Hypercube Based Team Multicast Routing Protocol for Mobile Ad Hoc Networks," *Proc. Ninth Int'l Conf. Information Technology (ICIT),* pp. 60-63, 2006.

[29] C.-T. Chang, C.-Y. Chang, and J.-P. Sheu, "Bluecube: Constructing a Hypercube Parallel Computing and Communication Environment over Bluetooth Radio System," *Proc. IEEE Int'l Conf. Parallel Processing (ICPP),* pp. 447-454, 2003.

[30] H. Huo, W. Shen, Y. Xu, and H. Zhang, "Virtual Hypercube Routing in Wireless Sensor Networks for Health Care Systems," *Proc. IEEE First Int'l Conf. Future Information Networks (ICFIN),* pp. 178-183, 2009.

[31] C. Shannon, N. Petigara, and S. Seshasai, "A Mathematical Theory of Communication," *Bell System Technical J.,* vol. 27, pp. 379-423, 1948.

[32] J. Scott, R. Gass, J. Crowcroft, P. Hui, C. Diot, and A. Chaintreau, "CRAWDAD Trace Cambridge/Haggle/Imote/Infocom2006 (v. 2009-05-29)," 2009.

[33] H. Cai and D.Y. Eun, "Crossing over the Bounded Domain: From Exponential to Power-Law Intermeeting Time in Mobile Ad Hoc Networks," *IEEE/ACM Trans. Network,* vol. 17, pp. 1578-1591, Oct. 2009.

[34] A.P.N. Eagle and D. Lazer, "Inferring Social Network Structure Using Mobile Phone Data," *Proc. Nat'l Academy of Sciences,* vol. 106, no. 36, pp. 15274-15278, 2009.

Yunsheng Wang received the BEng degree in electronic and information engineering from Dalian University of Technology, China, in 2007, the MRes degree in telecommunication from the University College London, United Kingdom, in 2008. He is currently working toward the PhD degree in the Department of Computer and Information Sciences, Temple University, Philadelphia, Pennsylvania. His research interests include various topics in the application and protocols of wireless networks. His current research focuses on the efficient communication in delay tolerant networks and mobile social networks. He is a student member of the IEEE.

Wei-Shih Yang received the BS degree in mathematics from National Taiwan University, in 1976, and the MA and PhD degrees in mathematics from Cornell University, in 1981 and 1984, respectively. He is a professor in the Department of Mathematics at Temple University, Philadelphia, Pennsylvania. His research interests include probability theory, mathematical physics, and quantum computing. He has worked on phase transitions and critical phenomena in statistical mechanics, Gaussian and non-Gaussian random fields with applications to quantum field theory, percolation, Ising model, and self-avoiding random walks. His most recent works include the following topics: quantum random walks with applications to quantum computing and ruin probability of risk processes.

Jie Wu (F'09) is the chair and a Laura H. Carnell professor in the Department of Computer and Information Sciences at Temple University. Prior to joining Temple University, he was a program director at the National Science Foundation and distinguished professor at Florida Atlantic University. His research interests include wireless networks, mobile computing, routing protocols, fault-tolerant computing, and interconnection networks. He has published regularly in scholarly journals, conference proceedings, and books. He has served on several editorial boards, including *IEEE Transactions on Computers* and *Journal of Parallel and Distributed Computing*. He was general cochair for IEEE MASS 2006, and IEEE IPDPS2008, and was the program cochair for IEEE INFOCOMM 2011. He serves as general chair for IEEE ICDCS 2013 and program chair for CCF CNCC 2013. He was an IEEE Computer Society Distinguished Visitor and the chair for the IEEE Technical Committee on Distributed Processing. He is the recipient of 2011 China Computer Federation Overseas Outstanding Achievement Award. Currently, he is an ACM distinguished speaker and a fellow of the IEEE and the IEEE Computer Society.