Profile-Cast: Behavior-Aware Mobile Networking

Wei-jen Hsu[1], Debojyoti Dutta[2], and Ahmed Helmy[1]
[1]Department of Computer and Information Science and Engineering, University of Florida [2]Cisco Systems, Inc.
Email: [1] {wjhsu, helmy}@ufl.edu, [2]dedutta@cisco.com

***Abstract*—In this paper we advocate a service paradigm, *profile-cast*, within the communication framework of delay tolerant networks (DTN)[2]. This novel approach leverages the behavioral patterns of mobile network users for delivering messages to a sub-group of users as defined by their *profiles* (e.g., interest, social affiliation, etc.). We study large data sets of user *mobility profiles* and present a case-study of *mobility profile-cast* with a *similarity-based* forwarding protocol. We show that behavior-aware protocol design has a great potential – we reduce the total number of transmissions to 45% of flooding under 92% delivery success rate, or to only 3% transmissions of flooding under 61% delivery success rate. It also leads to shorter delay (at least 30% less) as compared to a random transmission protocol.**

I. INTRODUCTION

Recent years have witnessed significant growth in the adoption of portable wireless communication and computing devices (e.g., laptops, PDAs, smart phones) and large-scale deployment of wireless networks (e.g., cellular, WLANs). We envision that future usage of mobile devices and services will be highly personalized. Users will incorporate these new technologies into their daily lives, and the way they use new devices and services will reflect their personality and lifestyle. This opportunity opens up the door for novel paradigms such as *behavior-aware* protocols and services. Such services analyze user behavior and leverage the underlying patterns in user activity to adapt their operation and have the potential to work more efficiently and suit the real needs of the users better. One classical example is the online stores (e.g., Amazon.com) that provide personalized shopping offers based on browsing history. However, little attention has been directed towards leveraging behavioral patterns for services or protocol design in the mobile communication paradigm.

In this paper we focus on a new class of service named *profile-cast*. In this service, instead of targeting a particular end-point or host, the message is to be delivered to *all* hosts with a certain property (i.e., those who match with the specified *profile* are intended receivers). There exist a wide variety of ways by which a *profile* can be defined. The *profile* can be based on the user's interest (e.g., movie-goers or baseball lovers), social affiliation (e.g., students in the computer science department), or other behavioral patterns. Potential applications of such a service are notification or advertisement for a scoped group within the general population, or a matching service trying to find people with certain characteristics or interests. Note that the notion of *profiles* refers to the implicit, intrinsic properties one discovers from the behavioral patterns of users. This distinguishes *profile-cast* from traditional multi-cast where users join multicast groups explicitly: In *profile-cast*, a user does not join particular groups to receive messages. Instead, the groups are implicitly defined by the intrinsic properties of the users, and revealed by the way the users utilize the network.

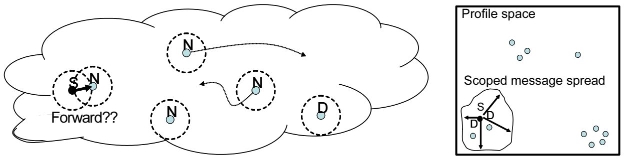
The contribution of the paper is two-folded. First, we advocate profile-cast as a viable service paradigm in mobile wireless networks, as it works under the properties of such networks (e.g., dynamic network structure and varying user population, hence difficult to maintain centralized registry) and suits the core needs for many applications (e.g., when the target nodes are not defined by their identities, but by their properties). We advocate the *profile-cast* within the *communication framework* of *delay tolerant networks (DTN)*. We believe that this is a promising new direction – as the small hand-held devices equipped with short-range radio (e.g., bluetooth) gain popularity, *profile-cast* services in DTNs might provide a new paradigm to navigate the messages through the mobile society without relying on established infrastructure or registry, reaching the targeted groups defined by their underlying properties (i.e., the chosen *profiles*).

Second, we consider user *mobility profile* as a case study to demonstrate the efficacy of the *profile-cast* paradigm. We leverage the long-run trends in user mobility to reveal the social context and categorize users into groups[1], and then target the message propagation to the identified groups. Our *similarity-based* profile-cast protocol makes the message forwarding decision based on the distance between users in the multi-dimensional *profile space*. By incorporating mobility profiles, we limit the scope of message delivery in DTNs to a specific behavioral group. Thus we avoid the high overhead of the epidemic routing [4] (eliminate more than half of the transmissions with a little reduction in delivery success rate) and out-perform random-walk based protocols in terms of delivery delay.

The paper is organized as follows: In section II we further elaborate the concept of profile-cast and its linkage to routing decisions in the DTN framework. After the formal introduction of the problem, we present our solution, the *similarity-based* profile-cast protocol, in section III. We further explain the setup of evaluations and present the results in section IV. Our future work is outlined in section V and related work is presented in section VI. We conclude in section VII.

II. PROFILE-CAST IN DELAY TOLERANT NETWORKS

Delay tolerant networks (DTNs) [2] are networks characterized by sparse, time-varying connectivity, in which end-to-end spatial paths from source to destination nodes are



(a) Physical view. (b) Conceptual view.

Fig. 1. Two different views of the profile-cast service in DTN. The conceptual view of scoped flooding in the profile space has to be implemented through message forwarding decisions at physical nodal encounter events.

often not available. Messages are stored in intermediate nodes and moved across the network with nodal mobility. Nodal encounter events (i.e., when nodes move into the transmission range of each other) provide the opportunities for nodes to communicate with each other in DTNs. One particular important decision to make for nodes in DTN is whether to forward a packet to other nodes they encounter with, as illustrated in Fig. 1(a). Such decisions have implications on many aspects of how efficiently the routing strategies work, such as delay, overhead, and message delivery success rate. There exists a tradeoff between these performance metrics, and a well-designed protocol should provide a mechanism for its users to strike a right balance for the given environment. The key research challenge is to make an intelligent decision with the *local* information available to the two encountering nodes, assuming no knowledge about *global network properties*, which is usually unavailable in decentralized networks such as DTNs.

For our *profile-cast* application, the goal is to reach a set of nodes with a certain similar property. The conceptual view of the problem is illustrated in Fig. 1(b). We imagine a virtual, high-dimensional *profile space* where each node is represented by a point in the space. The nodes that are similar with respect to the property we use to construct the profile space should be close to each other in this space, and dissimilar nodes should sit far apart. Our *profile-cast* application corresponds to a scoped-flooding in the *profile space*: The nodes should keep forwarding the message to those who are similar to them under the considered profile, but ignore those who are dissimilar. Linking the figures in Fig. 1, they point out a need for nodes to evaluate their mutual similarity in the considered profile space when they encounter, and use this piece of information to guide the routing decisions in the DTN. We propose a *similarity-based protocol* for this purpose in the next section.

We use *mobility profile* as an example to illustrate the efficacy of the *profile-cast service paradigm*. We choose the mobility profile for the study for the following reasons. First, it has been shown mobility is one of the distinguishing features to differentiate users from a large population[1]. Groups with distinct behavioral patterns can be identified with respect to the long-run mobility patterns, and we use these groups as our targets in the *profile-cast* protocol. Second, *mobility-profile-cast* ties with some new services in the ad hoc network. For example, a student loses a wallet and wishes to send a message to other fellow students who visit similar places often as he does to look for it. Or, the manager of the library may want to send an announcement about power shutdown only to its frequent patrons. These services are mobility pattern specific, and none of the existing service paradigms serves the need of identifying the intended message recipients from a diverse population well. Third, to evaluate the effectiveness of our proposed protocol realistically, we need detailed traces of user behavior with respect to the profile we choose. User mobility data is more available than other network traces (e.g., user interest or social affiliation), hence we choose to leverage these data sets [6], [7] first[1].

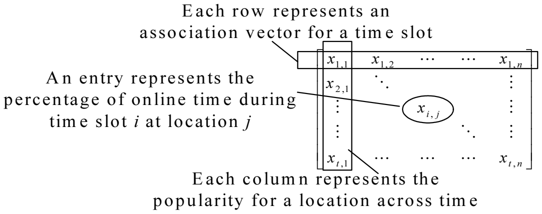


Fig. 2. Illustration of the association matrix representation.

III. A SIMILARITY-BASED PROFILE-CAST PROTOCOL

In this section we explain the details of the similarity-based profile-cast protocol, using *mobility profile* as the example. The protocol contains two phases. (1) *Profiling*: Each mobile node keeps track of its own *mobility profile* as it moves around the given environment. This is an individual effort made by each node independently – every node is responsible only for keeping its own *mobility profile*. (2) *Forwarding decision*: When nodes encounter with each other, they exchange the *mobility profiles* to determine whether a message forwarding should take place.

A. Profiling User Mobility

To enable *mobility-profile-cast* services, it is important to first have a descriptive representation for user mobility profiles. We choose to construct the *association matrix*, as illustrated in Fig. 2, to describe the long-run mobility trend of a mobile user. All users first agree upon a set of locations that are used as references to keep track of their mobility profile. For each time slot, each user generates an *association vector* that summarizes its association with the reference locations during this time slot. We choose to use a day as the time slot since it represents the most natural behavior cycle in our lives. Each entry in the vector represents the *fraction* of time the user spends at the location during the time slot. To represent a user's *mobility profile* for the long run, we concatenate the association vectors for each time slot (day) to construct the *association matrix* for the user. If there are n distinct locations and the trace period consists t time slots, the *association matrix* for a user is a t-by-n matrix.

The *association matrix* representation captures the relative importance of locations to each user (i.e., the *preference* in the user mobility process). Based on this representation, we classify the whole user population into distinct behavioral groups with clustering methods detailed in [1]. These groups

---

[1] However, if other data sets were available, similar protocols as the one proposed in section III could be used for other user *profile* as well.

correspond to users with unique *mobility profiles*. In this work, we take these behavioral groups as the targets for *mobility profile-cast*.

B. Evaluating User Similarity based on the Mobility Profiles

When nodes encounter with each other, they need to exchange the *mobility profile* for the evaluation of their similarity. However, the raw *association matrix* is too large in size to be exchanged efficiently. Hence we need a good method for summarizing the association matrix. We have established that singular value decomposition (SVD) provides an efficient way for this purpose[1]. SVD can be viewed as a systematic procedure to obtain representative vectors that capture the most remaining power in the *association matrix*, defined by

$$\begin{aligned} u_1 &= \arg\max_{\|u\|=1} \|X \cdot u\| \\ u_k &= \arg\max_{\|u\|=1} \|(X - \sum_{i=1}^{k-1} X u_i u_i')u\| \quad \forall k \geq 2, \end{aligned} \tag{1}$$

where X is the *association matrix* and u_i' denotes the transposed vector u_i. We can interpret the vector u_j's as the vectors that describe the user's mobility in decreasing order of importance, with its relative weights quantified by the ratio of the corresponding singular values, i.e., $w_{u_j} = \sigma_j^2 / \sum_{i=1}^{Rank(X)} \sigma_i^2$. We refer to these unit-length vectors as the *eigen-behavior* vectors for the user. The absolute values of entries in an *eigen-behavior* vector u_j quantify the relative importance of the locations in the user's j-th behavioral mode. Empirically, we observe from two large-scale mobility traces collected from WLANs [6], [7] that this technique is very effective for most users[1]. For more than 90% of the mobile users, we can summarize at least 90% of the power in their *association matrices* with at most *five* eigen-behavior vectors for semester-long mobility profile. Hence we have a concise yet highly accurate representation of user *mobility profile* for exchange when the users encounter with each other.

When two users meet with each other, they exchange the summarized *mobility profiles* (i.e., *eigen-behavior vectors* with their weights) of their previously collected mobility pattern and decide whether they are similar at the spot. The similarity index between users U and V, $Sim(U,V)$, is calculated as the weighted sum of inner products of the *eigen-behavior* vectors[1].

$$Sim(U,V) = \sum_{i=1}^{rank(U)} \sum_{j=1}^{rank(V)} w_{u_i} w_{v_j} |u_i \cdot v_j|, \tag{2}$$

where u_i and v_j are *eigen-behavior* vectors from user U and V, and w_{u_i} and w_{v_j} are their corresponding weights as defined previously. If the similarity index is larger than a threshold, they exchange the message. Note this decision is solely local, involving only the two encountered nodes. The philosophy behind the protocol is, if each node delivers the message only to others with high similarity in mobility profile, the propagation of the message copies will be scoped within a group of similar users. The threshold that triggers the message transmission provides a control knob for the protocol user to adjust the tradeoff between performance metrics. A high-valued threshold favors low transmission overhead, while a low-valued threshold leads to short delivery delay and high delivery success rate.

Finally, as a side comment, the approach of exchanging only the relevant behavioral summaries between the encountered nodes helps to preserve user privacy – such behavioral information is not kept in a public directory, and each user can manipulate them to hide their true behavior if they so desire.

TABLE I
FACTS ABOUT THE USC TRACE[6]

Time/duration of trace	2006 spring semester (94 days)
Start/End time	01/25/06 - 04/28/06
Location granularity	building
Unique locations	137 buildings
Unique MACs analyzed	5,000

IV. EVALUATION AND COMPARISON

A. Evaluation Setup

In this section we describe the experiment setup to evaluate our similarity-based profile-cast protocol presented in the previous section. We utilize the USC wireless LAN (WLAN) trace [6] to study the message transmission schemes *empirically*. Some logistic details of the data set can be found in Table I. We assume that two nodes are able to communicate (i.e., encounter with each other) when they are associated to the same location in the WLAN. Note that the WLAN infrastructure is merely used to collect user location information, and the messages can be transferred only between the users without using the infrastructure. We split the WLAN trace into two halves. The first half of the trace is used to determine the grouping of users based on their *mobility profile* and we identify 200 groups with distinct behavioral pattern in terms of *mobility*[1]. Then we evaluate the group-cast protocol performances using the second half of the same trace. For each group with more than five members, we randomly pick 20% of the members as the source nodes sending out a *one-shot message* to *all other members in the same group* at the beginning of the second half of the trace. We use the same set of senders for all evaluated protocols to ensure a fair comparison.

We compare the performance of our *similarity-based* protocol with several alternative protocols described below based on the following metrics: (1) *Delivery ratio*: The number of nodes received the message over the number of intended receivers. (2) *Delay*: The average time taken to deliver the messages to recipient nodes. (3) *Overhead*: The total number of transmissions involved in the process of message delivery.

Flooding (epidemic routing)

This is the simplest decision rule for message forwarding in DTN. All nodes in the network are oblivious to mobility profiles and blindly send out copies of the message to nodes who have not received it yet. This scheme is also known as the *epidemic routing*[4] in DTN, using the analogy that the message propagates in the network like an epidemic. This is

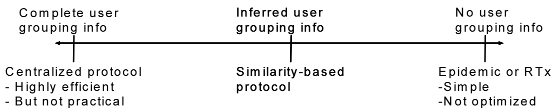


Fig. 3. The chosen protocols for evaluation span the spectrum of user grouping knowledge used in the forwarding decision process.

also the most aggressive forwarding strategy in DTN. Under idealistic environment (i.e., no packet drop due to wireless contention or insufficient buffer size), this is also the strategy that achieves the shortest possible delay and the best delivery success rate.

Centralized

In this ideal scenario, we assume that all nodes acquire perfect knowledge of the group membership through an oracle with no additional cost. In order to reduce the transmission overhead, nodes only propagate the message to others if they are in the same group. This ensures the message will never propagate to an unintended receiver, and only members of each group participate in message dissemination for their own group.

Random-transmission (RTx)

In the random transmission protocol, the current message holder sends the message to another node randomly with probability p when they encounter[2], and never transmits again (i.e., only the node who last received the message will transmit in the future). The message propagates across the network as a random walk among the nodes. Loops are avoided by not sending to the nodes who have seen the same message before. This process continues until a pre-set hop limit (i.e., TTL limit) is reached. We also vary the number of copies of active message (i.e., number of threads in the random walk) in the network. When m random walk threads are issued, the message originator is responsible for spreading the copies to m different nodes it encounters with, and each thread carries on independently as described above.

We have chosen the above protocols to span the spectrum of the degree of knowledge about the user grouping in the evaluation, as illustrated in Fig. 3. On one extreme of the spectrum we have the *centralized* protocol which has perfect knowledge about user grouping. This information provides an opportunity of highly efficient operation, but it is not realistic to assume its availability, hence the *centralized* protocol serves only as the *theoretical upper bound* of the performance. On the other extreme are the *flooding* and *RTx* protocols, both assuming no knowledge about user grouping at all. They are extremely simple but not optimized for the specific task of *profile-cast*. Our *similarity-based* protocol uses the similarity index defined in Eq. (2) to estimate the boundary where the scoped flooding should be stopped. It operates in the middle of the spectrum with *inferred* grouping information.

B. Evaluation Results

We simulate the protocols for *mobility-profile-cast* and show the results in Fig. 4. For all the performance metrics, we choose *flooding (i.e., epidemic routing)* as the baseline and show the performance metrics of the other protocols relative to that of epidemic routing in the figures.

In the figures we see that *flooding* has the lowest delay and the highest delivery ratio as it utilizes all the available encounters to propagate the message. However, it also incurs significant overhead. The average delay, which is the lowest possible under the given encounter patterns, is in the order of days (3.56 days in this particular case). *Profile-cast* based on *centralized* group membership information, the ideal scenario, shows great promise of behavior-aware protocols, as it significantly reduces the overhead (to only 3% of the *flooding*) while maintains almost perfect delivery ratio, with a little extra delay. There is such extra delay in the *centralized* protocol because the messages are carried by nodes in the targeted group only. It is possible to even reduce this delay by obtaining predictions of future encounter events through an oracle, as in [13]. We choose not to address this issue and instead show what can be achieved based on the perfect knowledge of user grouping alone, focusing our analysis on the spectrum of grouping information availability. The *centralized* protocol displays the ceiling performance one can achieve in terms of overhead reduction by incorporating knowledge of user grouping in the *profile-cast* service. However, note that it is not realistic to assume such centralized knowledge.

For our *similarity-based* protocol, its aggressiveness can be tuned with the forwarding threshold of the similarity index. We show the simulation results with various similarity thresholds in the figures. Label *Similarity* x indicates we use x as the threshold for message forwarding[3]. Experiment results show a significant reduction of overhead (only 2.5% of *flooding*) at the cost of delivery ratio (61% of *flooding*) if we set a high threshold such as 0.7 (i.e., sending almost exclusively within the same group). Note that the overhead is even less than that of the *centralized* protocol. This setting is perhaps more suitable for applications that one would want to operate with low overhead, and it is sufficient to reach a good part of the group but not essential to reach everyone. On the other hand, setting a low threshold (e.g., 0.5) leads to better delivery ratio (92% of *flooding*) but still cuts the overhead to 45% of *flooding*. This is suitable for messages that are intended to be received by most of the group, but one would not mind some misses in order to cut down unnecessary transmissions to irrelevant users. Tuning the transmission threshold provides a natural mechanism to strike a desired balance between overhead and delivery ratio. The delay incurred in *similarity-based* protocol is also not much different from the optimal case, the *flooding* (up to 14% more, in the case of similarity threshold 0.5).

For the *random transmission* protocol, its aggressiveness is tuned through the setting of number of active copies of the message (m) in the network and the TTL value for each thread. We use different variants of settings and show the results in the figure with labels *RTx*. We first show that *RTx* with infinite TTL does not perform well. Even if there is only

---

[2] In this paper we only show the results of $p = 1.0$. We have experimented with other values and discovered that they result in inferior performance.

[3] x can be in the range of $[0, 1]$. Setting the threshold to 1 would eliminate all transmissions, while setting it to 0 would degenerate the *similarity-based* protocol to *flooding*.

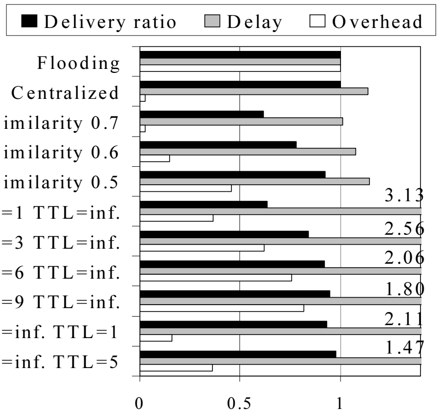


Fig. 4. Relative performance metrics of the group-cast schemes normalized to the performance of *flooding*.

one active copy (i.e., $m = 1$) in the network, the overhead is not low (0.37% of the *flooding* protocol). Comparing with the *similarity-based* protocol, when the delivery ratio is similar, the *RTx* protocol incurs much larger overhead (e.g., comparing *similarity* 0.7 with *RTx* $m = 1$ $TTL = inf.$, or *similarity* 0.5 with *RTx* $m = 6$ $TTL = inf.$. In both cases the *RTx* has 30% more overhead than the *similarity-based* protocol.). This is due to the group-membership oblivious behavior of the *RTx* protocol – in many cases the message is transmitted to some node out of the desired group, as membership information is not included to guide the forwarding decisions. Hence the *RTx* protocol, without a proper TTL control, makes a lot of unnecessary transmissions and results in high overhead. Using multiple threads with long TTL essentially degenerates the protocol to *flooding*.

On a different note, we try to exercise better control of the *RTx* protocol by using infinite number of threads with small TTL. The extreme example is to use $m = inf.$ $TTL = 1$. This degenerates the protocol to the scenario where the message sender sends directly to all the nodes it encounters with. We observe that the delivery ratio is quite high with this setup. This is mainly due to the choice of our application – when the goal is to send to a group with similar mobility patterns as the sender, intuitively the intended receivers will eventually meet with the sender directly. However, notice that the delay is still much higher than the *centralized* or *similarity-based* protocols, as in this case the *RTx* protocol does not take advantage of the intermediate nodes in the network. We further experiment with *RTx* $m = inf.$ $TTL = 5$, and discover it achieves good delivery ratio under moderate overhead, with improved delay. However, picking a suitable TTL is context-dependent, and it is only effective if the goal is to send messages to the nodes that are similar to the sender itself (i.e., close to the sender in the *profile space*).

We further illustrate the tradeoff between delivery ratio and overhead in Fig. 5, and mark the "operational region" of the compared protocols. The darkness of the ellipses in the figure represents the delivery delay. Ideally, one would want the protocol to work at the bottom-right corner, with high delivery ratio and low overhead, as close to the *centralized* protocol as possible. The *flooding* protocol also achieves good delivery ratio with low delay, but the overhead is too much. Our *similarity-based* protocol is shown by the white ellipse. Its operational region stretches from moderate delivery ratio with low overhead to high delivery ratio with moderate overhead. The *RTx* protocol with infinite TTL is represented by the dark gray ellipse, taking the space of moderate delivery ratio with moderate overhead to high delivery ratio with high overhead. As m increases, it degenerates to *flooding*. We also show with a properly chosen stopping threshold, the *RTx* protocol has the potential to operate in the high delivery ratio, low overhead area, as indicated by the light gray ellipse. However, its average delivery delay is still much higher than that of the *flooding* or *similarity-based* protocols (in the best case, at least 30% more than the similarity-based protocol), as *RTx* does not take full advantage of the available intermediate nodes in the DTN framework.

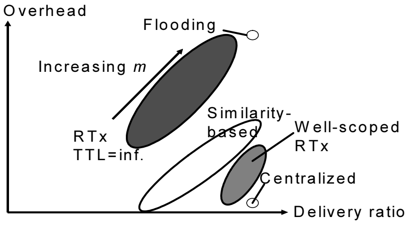


Fig. 5. The operation regions of the compared protocols in the delivery rate-overhead space. The darkness of the ellipses represents the delivery delay.

V. DISCUSSIONS AND FUTURE WORK

In this paper we designed and analyzed a similarity-based protocol for the *mobility-profile-cast* service with target nodes being the ones in the *same* behavioral group as the sender. In this section we briefly discuss about the relaxation of these assumptions, in particular, (1) How could a message be delivered to a group with a *specific mobility profile* given by the sender? (2) How could we use different type of *profiles* in the *profile-cast* service paradigm we advocate?

The similarity metric as defined in Eq. (2) can also be used to guide the message across the profile space to reach the targeted area defined by a *specific target profile* given by the sender – each message holder along the path compares the similarity between the *target profile* and the profiles of the encountered nodes, and forwards the message to another node that is more similar to the *target profile* than itself[3]. In this sense, the similarity metric constructs a gradient[15] for the message to follow and eventually reach the target area populated by nodes with similar behavior to the *target profile*. When the similarity is high enough, a scoped flooding should be performed to spread the message in the local neighborhood of the *targeted profile*. Under the scenario where the destined groups have different behaviors than the sender, the *similarity-based profile-cast* is necessary, as there is no information to guide behavior-oblivious protocols, such as the *RTx*, to the destination.

We briefly outline the extension of the *profile-cast* service to other types of *profiles* than mobility. This is our main direction of future work. The key challenge here is the separation of the virtual *profile space* and the physical encounters between nodes: It is not clear whether similar nodes in the *profile space* will encounter in the physical network, hence it is in doubt whether the similarity-based protocol will be able to guide the message as efficiently as in the *mobility space*. However, one promising finding in [8] points out that the encounter patterns of nodes in realistic mobility traces form SmallWorlds[12]. Each individual node meets occasionally with those who are not in its frequent encounter list, and it is such meetings that form the "short-cuts" within the network to make the network a SmallWorld. Leveraging this property, it is possible for a message sender to reach diverse groups (in terms of the *mobility profile*) within a small number of message exchanges. We envision a mechanism similar to that in [9] uses a small number of "contacts" to reduce the message dissemination overhead.

VI. RELATED WORK

In this paper we advocate the service paradigm of *profile-cast*. *Profile-cast* is related to multi-cast as both of them target at multiple receivers. However, in *profile-cast* the intended receivers are defined by their intrinsic properties, and there would be no explicit join to subscribe to a group as in multicast. Managing group membership in highly dynamic networks such as DTNs has attracted some attention recently[5] but it is still a hard problem to solve. The goal of the *profile-cast* service is to leverage underlying behavioral patterns (i.e., the *profiles*) to guide message delivery (i.e., incorporating *profile-awareness* into the protocol design and the routing procedure itself), which ties naturally to many context-centric services. Also note this goal is very different from many existing unicast routing protocols in DTN (e.g., PRoPHET[11]), as most of them focus on delivering messages to a node with *known, given node ID*. To apply such identity-centric routing protocols for *profile-cast*, additional directory services must be in place for property-identity lookup. However, such services is difficult to maintain in highly dynamic environments.

We leverage *mobility-based profile-cast* as an example in the paper to demonstrate the potential benefits when routing protocols incorporate user profiles into its design. *Mobility profile* has been used in [3] to guide unicast messages with a greedy gradient ascend approach. Our work differs from [3] in that we focus on reaching multiple users with similar *profile*, while their goal is to reach a particular user with a known *profile*. Finally, note that this application is different from *geo-cast*[14], which targets at the nodes *currently* within a geographical region as the receivers. Our target receivers are nodes with a certain mobility profile, *regardless* of their actual locations at the time the message is sent.

Understanding the underlying structure of network and the traffic patterns is important in designing routing protocols or services. Our work leverages the fact that *mobility profile* can be used as a distinguishing feature of the mobile users[1]. We expect to extend our work to leverage the SmallWorld property [8] for efficient *profile-cast* with any *profiles*. Routing protocols leveraging the SmallWorld property have been proposed mainly for query dissemination[9] or unicast services[10] in the literature, but it could be useful for other types of services.

VII. CONCLUSION

In this paper, we advocate *profile-cast* as a new service paradigm. We demonstrate that *mobility-based profile-cast* can be utilized for scoped message dissemination in DTNs and show improved performance over other candidates (i.e. epidemic routing or random transmission). The proposed *similarity-based* protocol shows significant overhead reduction (less than 45% of overhead compared to *flooding* with high delivery rate, or as low as 3% of overhead with a moderate 61% delivery rate). It is also better than the *random-transmission* protocol in terms of average delay (at least 30% less). We demonstrate that the insight from a detailed study of user behavior might provide new directions to improve services and protocols, especially as services become highly personalized.

REFERENCES

[1] W. Hsu, D. Dutta, and A. Helmy, "Extended Abstract: Mining Behavioral Groups in Large Wireless LANs" In Proceedings of ACM MOBICOM, Sep. 2007. Longer technical report available at http://arxiv.org/abs/cs/0606002

[2] K. Fall, "A Delay-Tolerant Network Architecture for Challenged Internets," In Proceedings of ACM SIGCOMM, August 2003.

[3] J. Leguay, T. Friedman, and V. Conan, "Evaluating Mobility Pattern Space Routing for DTNs," in Proceedings of IEEE INFOCOM, April, 2006.

[4] A. Vahdat and D. Becker, "Epidemic Routing for Partially Connected Ad Hoc Networks," Technical Report CS-200006, Duke University, April 2000.

[5] W. Zhao, M. Ammar, and E. Zegura, "Multicasting in Delay Tolerant Networks: Semantic Models and Routing Algorithms," In Proceeding of the 2005 ACM SIGCOMM workshop on Delay-tolerant networking, Aug. 2005.

[6] W. Hsu and A. Helmy, MobiLib USC WLAN trace data set. Downloaded from http://nile.cise.ufl.edu/MobiLib/USC_trace/

[7] D. Kotz, T. Henderson and I. Abyzov, CRAWDAD data set dartmouth/campus/ movement/01_04 (v. 2005-03-08). Downloaded from http://crawdad.cs.dartmouth.edu/dartmouth/campus/movement/01_04

[8] W. Hsu and A. Helmy, "On Nodal Encounter Patterns in Wireless LAN Traces," the Second International Workshop On Wireless Network Measurement (WiNMee 2006), April 2006.

[9] A. Helmy, S. Garg, P. Pamu, N. Nahata, "CARD: A Contact-based Architecture for Resource Discovery in Ad Hoc Networks", ACM Mobile Networks and Applications (MONET) Journal, Vol. 10, Issue 1, pp. 99-113, Feb. 2005.

[10] E. Daly and M. Haahr, "Social Network Analysis for Routing in Disconnected Delay-Tolerant MANETs," In Proceedings of ACM MOBIHOC, Sep. 2007.

[11] A. Lindgren, A. Doria, and O.Schelen, "Probabilistic Routing in Intermittently Connected Networks," Lecture Notes in Computer Science, vol. 3126, pp. 239-254, Sep. 2004.

[12] D. J. Watts and S. H. Strogatz. "Collective Dynamics of 'Small-World' Networks," Nature, vol. 393, pp. 440-442, 1998.

[13] S. Jain, K. Fall, and R. Patra, "Routing in a delay tolerant network," In Proceedings of ACM SIGCOMM, Aug. 2004.

[14] Y. Ko and N. Vaidya, "Flooding-Based Geocasting Protocols for Mobile Ad Hoc Networks," Mobile Networks and Applications, vol. 7, Issue 6, pp. 471-480, Dec. 2002.

[15] J. Faruque and A. Helmy, "RUGGED: RoUting on finGerprint Gradients in sEnsor Networks," In Proceedings of the The IEEE/ACS International Conference on Pervasive Services (ICPS'04), Jul. 2004.