Putting Contacts into Context: Mobility Modeling beyond Inter-Contact Times*

Theus Hossmann
Communication Systems Group
ETH Zurich, Switzerland
lastname@tik.ee.ethz.ch

Thrasyvoulos Spyropoulos
Mobile Communications
EURECOM, France
first.last@eurecom.fr

Franck Legendre
Communication Systems Group
ETH Zurich, Switzerland
lastname@tik.ee.ethz.ch

ABSTRACT

Realistic mobility models are crucial for the simulation of Delay Tolerant and Opportunistic Networks. The long standing benchmark of reproducing realistic pairwise statistics (e.g., contact and inter-contact time distributions) is today mastered by state-of-the-art models. However, mobility models should also reflect the macroscopic community structure of who meets whom. While some existing models reproduce realistic community structure - reflecting groups of nodes who work or live together - they fail in correctly capturing what happens *between* such communities: they are often connected by few bridging links between nodes who socialize outside of the context and location of their home communities. In a first step, we analyze the bridging behavior in mobility traces and show how it differs to that of mobility models. By analyzing the context and location of contacts, we then show that it is the social nature of bridges which makes them differ from intra-community links. Based on these insights, we propose a Hypergraph to model time-synchronized meetings of nodes from different communities as a social overlay. Applying this as an extension to two existing mobility models we show that it reproduces correct bridging behavior while keeping other features of the original models intact.

Categories and Subject Descriptors

C.2.1 [**Network Architecture and Descign**]: Wireless Communications; C.4 [**Performance of Systems**]: Modeling Techniques

General Terms

Performance, Theory

1. INTRODUCTION

Realistic mobility models are crucial for the simulation and performance analysis of wireless networking protocols. Simple random models, such as the once popular random walk and random waypoint [1], were proven unable to model important characteristics (e.g., periodicity and location preference) of real mobility. Consequently, they can lead to inconclusive or even wrong conclusions about the performance of networking protocols [2]. Hence, the properties of human mobility have recently attracted a growing amount of research interest [3, 4]. To this end, numerous sophisticated mobility models have emerged that try to better capture observed mobility characteristics [5, 6, 7, 8]. Each of these models tries to strike a different balance between accuracy, complexity, and analytical tractability.

In the context of Delay Tolerant [9] and Opportunistic Networks, where wireless contacts are exploited as opportunities to forward content, the long-standing benchmark for models is to reproduce realistic pairwise statistics (e.g. contact duration and inter-contact time distribution). These *microscopic* mobility properties and their characteristics have been extensively studied [2, 10, 11, 12] and are today correctly reflected by mobility models.

However, besides the microscopic properties, human mobility also has characteristic *macroscopic* structure [13, 14]: People meet *strangers* by chance, *colleagues*, *friends* and *family* by intention or *familiar strangers* because of similarity in their mobility patterns. This creates complex patterns of who meets whom, how often and for how long, which cannot easily be observed and understood by looking at individual node pairs (e.g., pair-wise inter-contact times) or time instances (e.g., distribution of number of neighbors).

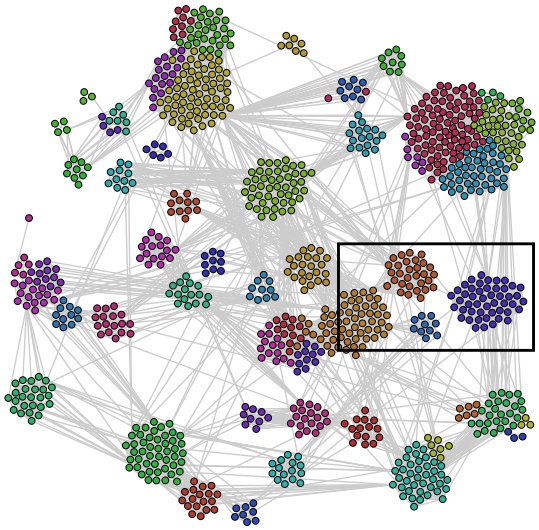
To this end, the concept of *contact graphs* has been recently proposed [13, 14, 15]: the sequence of *actual* contacts over time is mapped into a *conceptual* contact graph, where a link weight between two nodes captures the strength of the "relationship" (e.g. frequency of contacts or aggregated contact time between the two). Looking at mobility scenarios from the contact graph angle reveals the prevalence of *communities*, *bridges*, *hubs*, and other structures common in *social networks*. What is more, this social structure of mobility has been found crucial for the design of efficient DTN protocols [16, 17]. Fig. 1a shows an example of such a conceptual contact graph from a campus mobility trace.

In light of this, the bar is raised for mobility models. Good models, especially ones used for the analysis and simulation of Opportunistic Networking protocols, should also be able to accurately capture social properties. Preliminary analysis reveals that some of these models can reproduce realistic community structure, reflecting groups of nodes who work or live together. They can also reproduce different community sizes and modularity levels.

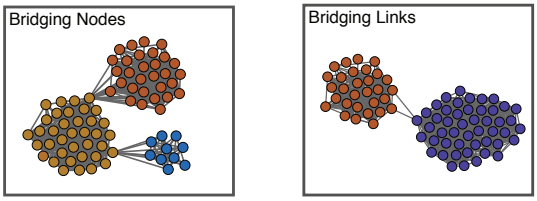
The *first contribution* of this paper is to show, however, that the vast majority of models fail to correctly capture what happens *between* communities. Detailed contact trace analysis of various traces (from different origins) reveals that communities are often connected by few *bridging nodes* and/or strong *bridging links*. Fig. 1b provides examples of these two types of interfaces. Comparing the findings from contact traces to contact graphs obtained from syn-

*This work was partially funded by the European Commission under the SCAMPI (258414) FIRE Project.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.
MobiHoc '11 , May 16-20 2011, Paris, France
Copyright 2011 ACM 978-1-4503-0722-2/11/05 ...$10.00.



(a) Communities (highly clustered nodes).



(b) Zoom-in: Bridges between communities.

Figure 1: Contact Graph of Dartmouth trace.

thetic mobility models, we find that the latter do not manifest bridging links.

Our *second contribution* is to analyze in detail the context and location of inter-community meetings in large datasets, where such context information is available or can be inferred. While this analysis is of more general interest, it also provides us insights in the origin of the discrepancy between bridges in traces and models: intra-community meetings (i.e., meetings between nodes belonging to the same community) predominantly occur inside a *small, often exclusive* set of locations (which we refer to as the "home location" for each community); however, inter-community meetings mostly occur outside the home location of either community.

Our *third contribution*, based on the above insights, is to propose a social overlay represented as a *Hypergraph* to model time-synchronized meetings between nodes of different communities. Applying this as an extension to two state-of-the-art mobility models we show that it is able to reproduce correct bridging behavior while keeping other features of the original model intact.

As a final note, we stress here the importance of accurately capturing the interface between communities. The cuts between tightly connected communities are bottlenecks for example for dissemination or diffusion processes, and reflecting them correctly is important for obtaining realistic results from simulation. While a detailed study of the impact of inter-community interfaces on performance protocols is beyond the scope of this paper, we believe that they provide a number of insights that should not be overlooked for the design of DTN protocols and their simulation.

The paper is structured as follows. We start by detailing our mobility datasets and the methodology used to map contacts to a conceptual social graph in Sec. 2. We then study the structure of this graph (e.g., communities and interface types) in Sec. 3. We look again at contacts and their context and location in Sec. 4 and their time synchronization in Sec. 5. Sec. 6 demonstrates how our main findings can enhance mobility models towards more realism.

2. METHODOLOGY AND DATASETS

We first describe the mobility datasets that we use for our analysis. These consist of traces from (i) real mobility measurements and (ii) mobility models. We then describe how we construct a contact graph out of each of these traces.

2.1 Empirical Datasets

To ensure generality of our results, we analyze four different mobility traces collected in several environments and with various methods: campus traces from Access Point associations and Bluetooth scans, and self-reported location data from a geo-social network application. Key features of the traces are listed in Table 1.

Dartmouth (DART) We use 17 weeks of the WLAN Access Point (AP) association trace [18], between Nov. 2, 2003 and Feb. 28, 2004. We chose the 1044 nodes which have associations at least 5 days a week on average i.e., at least $5 \times 17 = 85$ days. The trace is preprocessed to remove short disconnections ($< 60s$) and the ping-pong effect. Each AP is associated with a location context (e.g., library, residential, academic) that we will detail later on. We assume that two nodes are in contact when they are associated to the same AP at the same time.

ETH Campus (ETH) The trace collection at the ETH campus [19] is similar to the Dartmouth campus. We use 15 weeks of AP association trace between Oct. 25, 2005 and Feb. 3, 2006 and chose 285 nodes which associate to the network at least 5 days a week (i.e., 75 days). Short disconnections and ping-pong effect are filtered as well. Additionally, the location of each AP in the buildings is known.

Gowalla (GOW) Gowalla[1] is a geo-social network service where users check-in to close-by spots (e.g., restaurants, office buildings), logging their position and context. We use the location data of 473 heavy-users who, during the 6 months from Apr. to Sept. 2010, checked-in at least 5 days a week in the State of Texas. Since users only check-in and do not check-out, we cannot infer the stay duration at a spot. Therefore, we assume users are in contact when they have check-in less than 1 hour apart at the same spot. Note that by its nature, the GOW trace is more sparse than the other traces. However, it still contains a large number of check-ins ($\sim 400'000$) leading to a total of $19'000$ contacts[2].

MIT Reality Mining (MIT) The MIT trace [20] logs contacts between 92 campus students and staff, detected by Bluetooth scans with a scanning interval of 5 minutes. It is the only trace we use where contacts are measured directly (and not inferred from location), but has the drawback that we do not know in what context and location a contact happens.

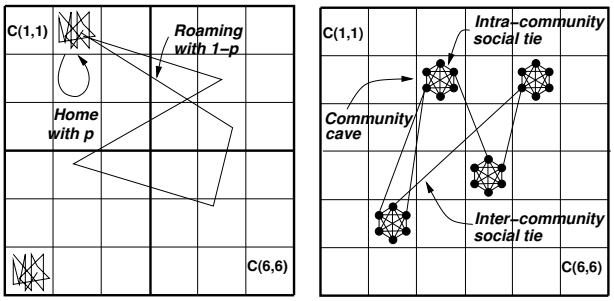
The Gowalla dataset is fundamentally different from the other ones. While the AP association datasets capture mainly campus work and study environment (plus residential for the DART trace), the Gowalla dataset, by the nature of the application, captures users while going out to eat or entertain. We hence assume that the AP datasets are biased a bit towards overestimating the number of contacts in home and work context, while the Gowalla dataset is biased towards overestimating the number of contacts in a social context. MIT captures work, home and leisure equally.

[1] `http://gowalla.com`

[2] Additional statistics about the GOW datasets (and the other ones), including distributions of pairwise tie strengths (see Sec. 2.3) can be found in [14].

	DART	ETH	GOW	MIT
# People and context	1044 campus	285 campus	473 Texas	92 campus
Duration	17 weeks	15 weeks	6 months	3 months
Type	AP associations	AP associations	Self-reported location	Bluetooth scanning
# Contacts total	$4'200'000$	$99'000$	$19'000$	$81'961$
# Contacts per dev.	$4'000$	350	40	890

Table 1: Mobility traces characteristics.



(a) TVCM 6×6 grid. (b) HCMM grid and social net.

Figure 2: TCVM and HCMM models.

2.2 Synthetic Mobility Models

We consider three synthetic mobility models [7, 5, 6] known to reproduce various empirical mobility properties. These models cover a range of different design philosophies (location-driven vs. social network driven) and complexity (tractable and simple vs. more realistic, but not tractable). Our goal is to better understand to what extent these models can capture the complex (social) structure observed in the traces.

TVCM: In the Time-variant Community Model [5], each node is randomly assigned to one or more home location areas on the plane. Transitions in, out, and between home locations are governed by a simple 2-state Markov Chain as illustrated in Fig. 2a. Nodes perform random waypoint trips inside and outside home locations, with a probability $1 - p$ of roaming (in the next trip) and a probability p of staying or getting back in the home location. By choosing different transition probabilities p for each node, a large range of heterogeneous node behaviors can be reproduced. We use a simple TVCM scenario throughout our analysis, with only one home location per node (see Fig. 2a)[3]. To reproduce community structure observed in traces, we group the nodes in 10 communities and assign each of them to one of the 6×6 location areas (each of 100×100 meters). The number of nodes for each of the 10 communities is sampled from the community sizes found in the DART trace (see Sec. 3), which gives a total of 505 nodes.

HCMM: The Community-based Mobility Model (CMM) [21] was the first mobility model directly driven by a social network. The Caveman model [22] is used to define a network with (social) communities (in our scenario 10 caves with 10 nodes, reproducing roughly the average community size of the ETH trace) and each community is assigned to a home location as shown in Fig. 2b. In contrast to TVCM, transition probabilities are directly linked to the weights on the Caveman graph. Specifically, the probability that a node performs a mobility trip towards a community C depends on her social ties towards nodes currently in community C. The Home-cell Community-based Mobility Model (HCMM) [6] is an extension to CMM, adding location-driven mobility. The transition probability to a location L no longer depends on nodes currently at that location but on the total weight of nodes assigned to L as their home location (i.e., irrespective of their current position). With this extension, HCMM fixes a problem of CMM, i.e., that all nodes of a community tend to follow the first node roaming to a different community. As the authors of [6] show, this can cause all nodes to meet in one location and suppresses community structure. Since we are interested in realistic community structure, we thus use HCMM instead of the original CMM in our simulations.

[3]TVCM supports much additional complexity (see [5]). We choose here to use the minimum amount of complexity needed to create some non-trivial community structure.

	TVCM	HCMM	SLAW
Nodes	505	100	100
Speed	1-3 m/s	1-3 m/s	1 m/s
Structure	10 comm.	10 nodes per community	Hurst param. $h = 0.85$
Transm. Range	30m	30m	30m
# Contacts	$9'000'000$	$2'300'000$	$520'000$
Duration	3 days	3 days	3 Days

Table 2: Mobility model parameters.

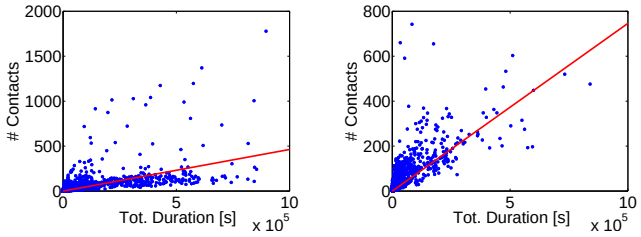
SLAW: SLAW [23] captures the (measured) Levy flight property of human displacements, where the travelled distance distribution between two points of interest (or waypoints) follows a Pareto law. In [7] the authors explain that this behavior can be seen as a consequence of people minimizing their daily displacements between clustered but remote points of interests.

We chose the parameters of the three models to be similar, such that they are comparable and produce community structures comparable to those found in the traces, especially with respect to modularity (see Sec. 3). The most imporant model parameters and statistics are reported in Tab. 2. For each synthetic models, we run a simulation to create a contact trace.

2.3 Contact Graph

Contacts happen due to the mobility of the users carrying the devices and reflect the complex structure in people's movements: meeting *strangers* by chance, *colleagues, friends and family* by intention or *familiar strangers* because of similarity in their mobility patterns. Our goal is to represent the complex resulting pattern in a compact and tractable way. This allows us to quantify structural properties beyond pair-wise statistics.

To represent the structure of a mobility scenario, we aggregate the entire sequence of contacts of a trace to a static, *weighted contact graph* $G(N, \mathbf{W})$ with weight matrix $\mathbf{W} = \{w_{ij}\}$. Each device (or rather person carrying a device) is a node of this graph and a link weight w_{ij} represents the strength of the relationship between nodes i and j. A key question is how to derive the tie strength between two nodes, i.e., what metric to use for w_{ij}, based on the observed contacts. This weight should represent the amount of mobility correlation (in space and time) between two nodes. Various metrics, such as the age of last contact [24], contact frequency [13] or aggregate contact duration [13] have been used as tie strength indicators in DTN routing.



(a) ETH. (b) MIT.

Figure 3: Dispersion of contact frequency and duration.

In our study, we consider *both*, contact frequency and aggregate contact duration. They capture different aspects, both of which are important for Opportunistic Networking (e.g., for data dissemination). Frequent contacts imply many meetings and hence many forwarding opportunities (short delays) and long contacts imply meetings where a large amount of data can be transferred (high throughput)[4]. The two features are correlated to different degrees as shown in Tab. 3.

DART	ETH	MIT	TVCM	HCMM	SLAW
0.52	0.63	0.81	0.96	0.98	0.92

Table 3: Correlation coefficients of duration and frequecy.

Since most network analysis metric require one-dimensional tie strengths, we map these two features to a scalar weight. We first assign each pair of nodes $\{i, j\}$ a two-dimensional feature vector, $\mathbf{z_{ij}} = \left(\frac{f_{ij}-\bar{f}}{\sigma_f}, \frac{t_{ij}-\bar{t}}{\sigma_t}\right)$, where f_{ij} is the number of contacts in the trace between nodes i and j, and t_{ij} is the sum of the durations of all contacts between the two nodes. $\bar{f}$ and $\bar{t}$ are the respective empirical means, and σ_f and σ_t, the empirical standard deviations. We normalize the values by their standard deviations to make the scales of the two metrics comparable.

We then transform the two-dimensional feature vector to a scalar feature value, using the *principal component*, i.e., the direction in which the feature vectors of all node pairs $\mathbf{Z} = \{\mathbf{z_{ij}}\}$, $i, j \in N$ have the largest variance. This is the direction of the eigenvector $\mathbf{v_1}$ (with the largest corresponding eigenvalue) of the 2×2 covariance matrix of frequency and duration. Fig. 3 shows two examples for frequency and duration values, along with the direction of the principal component. We then define the tie strength between i and j as the projection of $\mathbf{z_{ij}}$ on the principal component

$$w_{ij} = \mathbf{v_1}^T \mathbf{z_{ij}} + w_0,$$

where we add $w_0 = \mathbf{v_1}^T \left(-\frac{\bar{f}}{\sigma_f}, -\frac{\bar{t}}{\sigma_t}\right)$ (the projection of the feature value for a pair without contacts) in order to have positive tie strengths. The obtained weight is a generic metric that combines the frequency and duration in a scalar value and captures the heterogeneity of node pairs with respect to frequency and duration of contacts[5]. An analysis of weight distributions, degree distributions and other complex network metrics of the contact traces can be found in [14].

[4]Note that the age of last contact is not suitable for our purpose, since we need aggregate properties over the trace duration.

[5]This framework implicitly assumes stationarity of the underlying process, something not always true in some traces. In practice (e.g., for protocol design), one would implement some sliding window mechanism (see e.g.[17]). A thorough time-dependent analysis of these traces can be found in [25].

Trace/Model	# Communities	Q
DART	24	0.84
ETH	30	0.81
GOW	29	0.7
MIT	6	0.52
TVCM	10	0.73
HCMM	10	0.77
SLAW	2	0.063

Table 4: Number of communities and modularity.

3. SOCIAL STRUCTURE OF CONTACTS

We will now use the respective contact graphs to study the social ("macroscopic") properties of the mobility scenarios considered. As an example, Fig. 1a shows the contact graph of the DART trace. Clearly, we see that the edges are not randomly distributed over the nodes, but there is strong community structure (the nodes colored according to the community they belong to). *Communities* are (informally) defined as subsets of nodes with stronger connections between them than towards other nodes. They generally imply social groups (e.g., friends, co-workers) [26], and are thus of particular interests from both a sociological as well as a protocol design perspective (e.g.. for the design of DTN routing [15, 13] and multicast [16], collaboration for content distribution [27], security and trust systems, etc.). Looking at the community structure of the measured traces and from traces obtained from the synthetic models we will find that *synthetic mobility models, while able to generate high level community structure, they all fail to accurately capture inter-community linkage.*

3.1 Generic Community Structure

We use the Louvain algorithm [28] to identify communities. Finding the optimal allocation of nodes to communities is a computationally hard problem, and therefore, state-of-the-art algorithms use heuristics. The Louvain [28] algorithm starts with assigning each node its own community. It then iteratively – until no further improvement is possible – goes through all nodes and moves them to one of the existing communities, such that the gain in modularity ([29], see below) is maximal. In a second step, the communities are merged, if merging increases modularity. These two phases (moving nodes and merging communities) are iteratively repeated until no further improvement is possible. We choose this algorithm because it was reported to be fast and to find communities that are as good or better than other algorithms for a number of different graphs [28][6].

The number of identified communities is shown in Table 4 (left column). In addition to the number of communities, we are interested in the *modularity* of the resulting partition of nodes to communities. High modularity implies strong community structure, and has implications for node cooperation [32], community-based trust mechanisms [33], and routing. We compute the widely used *Newman modularity* [29]:

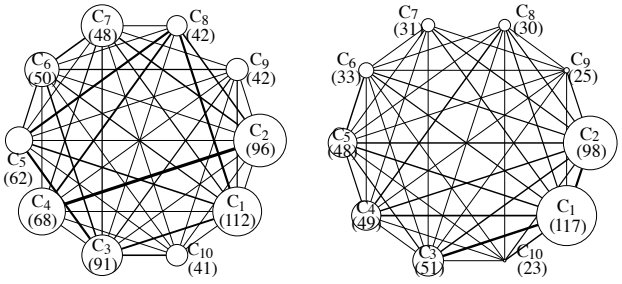
$$Q = \frac{1}{2m} \sum_{kl} \left(w_{k,l} - \frac{d_k d_l}{2m} \right) \delta(c_k, c_l),$$

where $d_k = \sum_l w_{k,l}$ is the degree of node k and $m = \frac{1}{2} \sum_l d_l$ is the total weight in the network. c_k denotes the community of node k thus, the Kronecker delta function $\delta(c_k, c_l)$ is 1 if nodes k and l share the same community and 0 otherwise. Q is always smaller

[6]We have used a second algorithm, based on *spectral clustering* [30], to detect communities. The community assignments are in agreement in most of the cases, with only small differences [31].

	$\mathbf{C_1}(24)$	$\mathbf{C_2}(23)$	$\mathbf{C_3}(16)$	$\mathbf{C_4}(16)$	$\mathbf{C_5}(7)$	$\mathbf{C_6}(6)$
$\mathbf{C_1}(24)$	**25%**	2.3%	0.8%	1.1%	0.03%	0.1%
$\mathbf{C_2}(23)$	2.3%	**27%**	3.6%	7.6%	0.45%	1.4%
$\mathbf{C_3}(16)$	0.8%	3.6%	**9.2%**	3.9%	0.19%	0.96%
$\mathbf{C_4}(16)$	1.1%	7.6%	3.9%	**9.7%**	0.29%	0.94%
$\mathbf{C_5}(7)$	0.03%	0.45%	0.19%	0.29%	**3.1%**	0.17%
$\mathbf{C_6}(6)$	0.1%	1.4%	0.96%	0.94%	0.17%	**2.2%**

Table 5: Percentages of total weight within and between communities (MIT trace). The number of nodes in the respective community shown in brackets.



(a) DART (10 largest comm.) (b) TVCM

Figure 4: Inter-community connections. The numbers below the community label indicate how many nodes are in the respective community.

than 1, can be negative, and $Q = 0$ is the expected quality of a randomly connected network with the same degree sequence (i.e., a network where the nodes have the same number of neighbors but are randomly connected). [29] reports modularities of above $Q = 0.3$ for different networks (social, biological, etc.).

The modularity for all contact traces and mobility models are listed in Table 4. From the list, we observe the following: (i) Modularity values vary in traces but are overall quite high[7]. (ii) SLAW has low modularity and no community structure (in the scenario we used, even with varying Hurst parameter values), hence we exclude it from the rest of our analysis. (iii) TVCM and HCMM have similar modularities as the traces. Hence, we conclude that existing models *can* emulate highly modular community structure even with simple scenarios.

3.2 Analysis of Community Cuts

While modularity captures *general* connectivity characteristics of a scenario, it does not say anything about how *individual* communities are connected. The interface (or "cut") between two communities on a static graph is known to be strongly linked to the speed of random walk based search (e.g. time to cross between two communities), sustainable throughput between the communities (through the max-flow min-cut theorem), etc. There is evidence that similar conclusions could be drawn for the contact graph.

3.2.1 Cut Capacity

Table 5 shows how the total weight in the network is distributed within and between the communities, for the example of the MIT dataset. The weight *within* a community is the sum of all weights (i.e., the *volume*) of edges with endpoints in the respective community. The weight *between* two communities C_i and C_j is the sum of weights of all edges across the *cut* between them:

$$\partial(C_i, C_j): \quad \textit{cut} \text{ between } C_i \text{ and } C_j$$
$$vol(\partial(C_i, C_j)) = \sum_{k \in C_i, l \in C_j} w_{k,l}.$$

A first observation is that inter-connections of communities are weak in many cases. For instance, in the MIT trace, communities 1 and 2 together contain more than 50% of the weights and 50% of the nodes. However, between them there is only 2.3% of the weight. For DART and TVCM, the same results are depicted in Figure 4 as a *community graph*. Each vertex represents a community, with vertex size showing the percentage of total weight accumulated within the community, and the edge width showing the weight in the cut between the communities. The community graphs of the other traces and models look similar.

Another observation is that heterogeneity in terms of *both* intra-community weight as well as inter-community (cut) weight *can* indeed be reproduced by the synthetic models. However, we want to stress that careful tuning of synthetic models is needed, when used to evaluate protocols based on social network structure [13, 15, 16, 34], to ensure that realistic community and inter-community structure is reproduced. While such tuning is possible, it may come at the expense of complexity and possibly loss of tractability.

3.2.2 Capacity Concentration

We finally zoom into the individual cuts and look at how the capacity is distributed over the node pairs. From Fig. 1b, we can already assume that for many community pairs, the capacity is not spread uniformly among the nodes. Instead, there are strong nodes and/or links, responsible for a large share of the weight between two communities. Analysis of the distributions of inter-community link weights shows that they are indeed strongly skewed [14]. This has important implications e.g. for *energy consumption*, *connectivity*, and *routing* in Opportunistic Networks. Clearly, the "narrower" the interface, the more challenging the problems: For the example of routing, if two communities are connected by a single strong link, this link may be hard to be identified and utilized by a routing protocol. Also, the narrower, the more stressed the bridges will be because they have to relay a large portion of the traffic.

This raises the following important question: Are nodes with (strong) cross-community links in general well connected to many nodes of the peer community, or is their weight towards the other community concentrated on a few links?

To answer this, we distinguish two different types of bridging behavior between communities: *bridging nodes* and *bridging links*. Informally, a bridging node is a node of community C_i which is also strongly connected to many nodes of community C_j (Fig. 1b, left)[8]. This is in contrast with the typical node of C_i which has strong links mainly within its community and much weaker links outside. A *bridging link* $\{k, l\}$ between nodes $k \in C_i$ and $l \in C_j$ is a stronger than average link and neither k nor l is a bridging node. In other words, k and l are only connected to *few* nodes of the respective peer community. In this case, the weight between the two communities is concentrated among much fewer pairs. As an example, imagine k has a total weight of 10 towards community C_j, and the average node of C_i has only a total weight of 1. If this total weight is distributed equally among 10 links (i.e., 10 links of weight 1), we call i a bridging node. However, if the total weight is concentrated on a single link to node l (of weight 10), $\{k, l\}$ is a bridging link (in this case k is *not* a bridging node).

[7]Similar values were reported by [13] for other traces and other community detection algorithms.

[8]Note that a bridging node has still more weight towards its own community; otherwise, it would have been assigned to C_j.

To differentiate between the two more formally, we introduce two metrics which we call, *node spread* and *edge spread*. We define the *node spread* of $k \in C_i$ to a community C_j as

$$S(k \to C_j) = \frac{|C_j^k|}{|C_j|},$$

where C_j^k is the smallest subset of C_j such that $\sum_{l \in C_j^k} w_{k,l} > 0.9 \times \sum_{l \in C_j} w_{k,l}$, i.e., the smallest subset that "contains" 90% of the weight of k towards C_j. We empirically choose a factor of 0.9 of the total weight to avoid counting the potentially many weights which happen just due to random contacts and remark that experiments with different factors do not change our results qualitatively.

Based on this, we define the *edge spread* of link $\{k, l\}$ as

$$S(k, l) = \max\{S(k \to C_i), S(l \to C_j)\},$$

i.e., the higher one of the two associated node spreads. Using the higher node spread allows us to implement the criterion that for a bridging link neither of the nodes may be a bridging node. With these metrics, we can now write down the criteria for bridging nodes and bridging links.

Bridging Node A bridging node k must fulfill the following two conditions:

1. It must be exceptionally strongly connected to C_j. We define $w_{k,C_j} = \sum_{l \in C_j} w_{k,l}$ and write the condition as

$$w_{k,C_j} \gg \underset{l \in C_i}{median}\{w_{l,C_j}\}$$

2. It has a high node spread $S(k \to C_j)$.

Bridging Link Similarly, a bridging link $\{k, l\}$ fulfills the following two conditions:

1. It has exceptionally strong weight compared to other links between C_i and C_j

$$w_{k,l} \gg \underset{m \in C_i, n \in C_j}{median}\{w_{m,n}\}$$

2. It has a low edge spread $S(k, l)$.

To see if we observe bridging nodes or rather bridging links in the traces, Fig. 5 shows histograms of the edge spread of strong inter-community links[9] for the traces and models. We observe a fundamental difference between the traces and the mobility models: Traces contain *both* bridging nodes and links, with a tendency towards bridging links, i.e., more concentration of the weight to few links. On the other hand, the models show exclusively bridging nodes, no bridging links. Unlike the case of heterogeneous inter-community volume (studied in Section 3.2.1), which can be emulated with careful fine-tuning of the models, this consistent lack of bridging links seems to be due to deeper, design-related reasons.

We believe that the main reason behind this is the following: *while models successfully incorporate geographic concepts and social relations to drive mobility decisions, they fail to consider one important element of human mobility, namely,* context. Nodes of different communities, meeting *outside* the location of their home communities in a social context, are typically not modeled. In the following section, we will show that it is indeed such meetings which create bridging links with a small spread.

[9]For this measurement, we classify a link as strong if it is at least 3 standard deviations stronger than the median. Experiments with other thresholds show qualitatively similar results.

Context	APs	Contacts	Community	Bridge
Academic	28%	7.7%	4.9%	32%
Admistration	12%	1.4%	1.4%	1.2%
Library	8.8%	1.2%	0.12%	11%
Residential	39%	86%	90%	45%
Social	8.4%	0.8%	0.5%	3.5%
Athletic	3.4%	3.1%	2.7%	6.5%
Total	533	4'184'804	3'756'503	428'301

Table 6: Percentage of contacts per context (Dartmouth): total (col. 3), intra-comm (col. 4), inter-comm (col. 5).

Context	Spots	Contacts	Community	Bridge
Office	4%	12%	18%	4.9%
Events	11%	18%	13%	24%
Food	54%	43%	46%	40%
Shopping	20%	17%	13%	20%
Home	2.5%	2.5%	2%	3.1%
Other	8.2%	8.2%	7.8%	8.6%
Total	3813	14875	7716	7159

Table 7: . Percentage of contacts per context (Gowalla): total (col. 3), intra-comm (col. 4), and inter-comm (col. 5)

4. CONTEXT OF CONTACTS

To show that social meetings happening outside the location and context of a community cause bridging links with small spread, we analyze the traces for the *context*. In most traces, context is not available and we only know who meets *whom* and *when*, but not *where* (location) or *why* (context). Exceptions, which we analyze here, are the Dartmouth and Gowalla trace. As these have a large number of nodes and are of different origins, we believe that they suffice to provide some important first evidence.

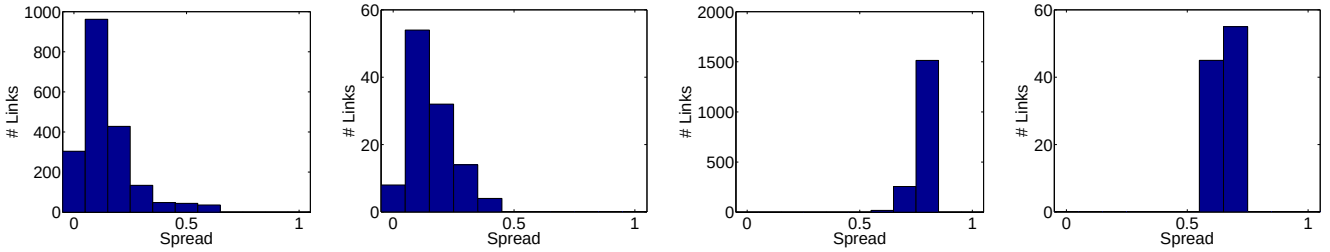
4.1 The Context of Contacts

In the *Dartmouth* trace, we derive the context of a contact from the building in which the respective AP is located. The trace distinguishes 6 different contexts (i.e., types of buildings): Academic, Administration, Library, Residential, Social and Athletic[10]. The percentage of APs and contacts per context is listed in Table 6. Note that the percentage of contacts does not perfectly correlate with the percentage of APs of a given context. For instance 86% of all contacts happen at *residential* APs, which account only for a 39% of all APs.

The last two columns show the percentage of intra-community ("*Community*") and inter-community ("*Bridge*") contacts happening in various contexts. Observe that for all contexts, the percentages of both community (column 3) and bridge contacts (column 4) differ largely from the aggregate statistics for all contacts (column 2). This suggests that meetings between nodes belonging to the same community happen at different contexts that meeting between nodes in different communities (which are responsible for the community bridging behavior). Put differently, *knowing whether a contact is of type Community or Bridge substantially changes the probability of the context in which it occurs*: For example, without further knowledge any given contact happens with a probability of 1.2% in a library. However, given that it is a *Bridge* contact, will increase this probability by a factor of 10.

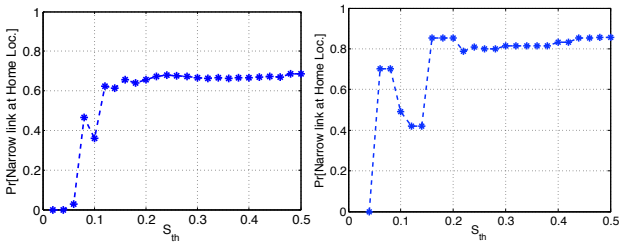
We can also infer the context of contacts in the *Gowalla* trace. Each Spot at which users can check-in is assigned to one of more than 300 categories by the creator of the spot. We group the cate-

[10]There is a seventh context "Other", which we dismiss here.



(a) DART mean=0.2 (b) GOW m=0.22 (c) TVCM m=0.74 (d) HCMM m=0.66

Figure 5: Histogram of the spread of bridging links. Not shown: ETH (m=0.32), MIT (m=0.46).



(a) DART (b) ETH

Figure 6: Probability of narrow bridges in home locations.

gories to 6 contexts[11]: Office, Events, Food, Shopping, Home and Other. Results are summarized in Table 7. Similarly, the probability of bridging contacts occurring, for example, in the office context is 4 times smaller than the probability of intra-community meetings occurring in the same context. Note that, due to its nature, the Gowalla dataset is biased towards social meetings, at events, restaurants, etc. (to which users check in more often than, say, at home). This causes identified communities to have a different meaning than in the DART trace, where communities happen mainly at home: In Gowalla, a community is rather a group of nodes often meeting in social contexts (e.g., eating, going out).

This analysis clearly shows that community and bridge contacts differ in terms of context in which they happen. In the next Section, we will refine the analysis and show that they also differ in terms of geographical location where they happen.

4.2 The Location of Contacts

As mentioned earlier, human mobility is strongly driven by location. Hence, a lot of synthetic mobility models introduce the concept of "home locations": nodes tend to move inside their home location(s) most of the time (e.g., home or office environment), and occasionally visit other locations. In addition to the general context of contacts studied before, we are therefore interested in the actual locations where contacts occur, both intra-community contacts (to confirm the basic intuition of synthetic models) and inter-community contacts.

To this end, we create for each community a *location profile* of where its intra-community contacts happen. For each community, we extract the list of intra-community contacts. For each location (AP) we count the intra-community contacts and define the *home location set* of the community as the smallest set of locations such that at least 90% of the intra-community contacts is covered. We empirically chose a threshold of 90% to account for the big majority of the contacts without taking into account random contacts happening at other locations (other thresholds give similar results). We denote the home location set of community C_i as $L(C_i)$.

[11]We only assign popular contexts to categories, that is, categories which account for at least 0.5% of all contacts

Our first finding is that, indeed, the number of locations in which most contacts between nodes in the same community happen is rather limited. On average, the home location profiles for the Dartmouth communities contain 4.5 locations. A second observation is that home locations sets of different communities do not overlap, that is, $L(C_i) \cap L(C_j) = \emptyset$ for all i and j. In other words, each community observed in the contact graph seems to have its own, mostly exclusive home location (implying perhaps that this *contact-based* community is an actual *social* community).

We next verify that bridging links not only happen at a different context, but very often outside the home location sets of both nodes' communities. We go through all inter-community contacts and check where they happen. In aggregate, we see a mixed picture: In the Dartmouth trace, 44% of all bridging contacts happen *outside* the respective home location sets, 56% *within* the home locations. While these numbers seem roughly split, they should be considered in light of the fact that 81% of all contacts happen inside home communities. This implies that inter-community contacts happen with 4.9 times higher probability outside home locations than the average contact.

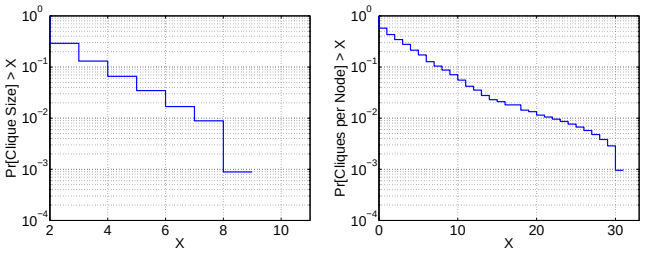
Our last finding is that the probability of a contact happening outside the home location sets of its respective communities, depends strongly on the spread of the link. To show this, we split the links into two classes *narrow*, if the spread is smaller than a threshold S_{th}, and *broad* if it is larger. Fig. 6 shows how the probability of a contact of a *narrow* link happening in a home location depends on the threshold S_{th}. We see that for both DART and ETH traces, for very narrow links the probability is very small, thus, almost all links happen *outside* home locations. As we include more and more links with higher spread (moving to the right in Fig. 6), the probability increases. Thus, links with higher spread tend to happen at home locations.

This observation has important implications for mobility model design. *While most mobility models cannot directly manipulate how the aggregated contact graph will look like, they can more easily manipulate the locations where contacts occur. The strong correlation of context and bridging behavior found in this section, directly implies that one could manipulate the location of contacts, in order to improve the social behavior of the model*[12]. We will show how this could be achieved in the next two sections.

5. SYNCHRONIZATION OF CONTACTS

We saw that links with low spread happen with high probability outside the respective communities' home locations, and how this affects the observed inter-community interface. In order to transform this into a model for inter-community meetings, the following

[12]It is important to note that, while mobility models using a social graph to drive mobility decisions (such as CMM and HCMM) can create bridging links on this graph, this does not mean that these links will appear in the contact graph as well.



(a) Clique Size CCDF (b) Cliques Per Node CCDF

Figure 7: Synchronization of nodes locations (DART trace).

questions arise naturally: Do the nodes of bridging links visit the same location independently and meet there randomly or do they visit such social locations synchronously? Is it just pairs of nodes visiting a certain location, or do we see groups (*cliques*) of nodes?

In order to see whether two nodes visit a location synchronously, we measure the *overlap* of the time they spend there. We define the overlap $O_i(u,v)$ of nodes u and v at location i using the *Jaccard similarity index* as follows

$$O_i(u,v) = \frac{|T_i^u \cap T_i^v|}{|T_i^u \cup T_i^v|}$$

where $|T_i^u \cap T_i^v|$ is the total time both u and v are at location i, and $|T_i^u \cup T_i^v|$ is the time either u or v (or both) spend at i.

To assess whether a certain overlap implies synchronous behavior, we relate the *measured overlap* to the *expected random overlap*. We estimate the expected random overlap $\widehat{O}_i(u,v)$ (i.e., the expected overlap if u and v visit i independently) using Monte Carlo simulation as follows. We extract the set of durations of visits to location i. Picking a random starting time for each visit of both nodes (such that visits of the same node do not overlap in time), we shuffle the order and time of the visits, while maintaining their number and durations. Given such a random arrangement, we measure the overlap as defined above. Repeating this 1000 times for each node pair, we estimate the expected value of the random overlap.

To distinguish synchronous from asynchronous behavior, we compare $O_i(u,v)$ and $\widehat{O}_i(u,v)$ using a simple test: If the measured overlap exceeds the expected random overlap by a factor of 10, we call the pair synchronized at this specific location[13].

People typically socialize not only in pairs but also in larger cliques. Knowing the size of such cliques is important for modeling inter-community meetings. Hence, we measure the clique size distribution. Using the DART trace, we create a *Synchronicity Graph* for each location i (which is not a home location to a community). The vertices of the graph are the set of users which visit AP i. We put an edge between two nodes if we find synchronous behavior for them, according to the test above. In this Synchronicity Graph, we detect all k-Cliques (fully connected sub-sets of k nodes, that are not contained in an even bigger clique) to detect groups of synchronized nodes[14].

In the DART trace, we find more than $1'000$ Cliques of size 2 or larger, with a maximum size of $k = 9$. Figure 7a shows the empirical CCDF of the clique size, over all APs. From the almost straight line in the log-linear axis, we conclude visually that clique size follows a geometric distribution. Using a maximum likelihood estimator, we fit a geometric distribution and find a parameter of $p_1 = 0.64$. Note that the geometric distribution is *shifted*, since the smallest clique size is 2 (i.e., if X denotes the random variable of clique size, $X - 2$ is geometrically distributed). A Kolmogorov-Smirnov test does not reject the null hypothesis of a geometric distribution at a significance level of $\alpha = 0.05$.

Finally, we would like to know how many locations a node typically visits together with other nodes. Hence, we analyze, how many cliques (of size 2 or bigger) a node is part of. The CCDF is shown in Figure 7b. Again, the almost straight line in the log-linear scale suggests a geometric distribution, and a maximum likelihood estimator gives a parameter of $p_2 = 0.28$[15].

Using these insights, we are now ready to create a model for social meetings across communities.

6. MODELING THE SOCIAL CONTEXT

Section 3 sheds light on an important problem with current synthetic mobility models. Sections 4 and 5 already suggest how this problem could be fixed. Based on these insights, we now proceed to propose a model for the social meetings outside community locations. Note that our goal is *not* to propose an entirely new mobility model, as there are already many good existing ones. Instead, we present an addition, which we call *social overlay (SO)*, that is applicable to various existing models. The social overlay is based on a Hypergraph [35] model, calibrated with the insights from trace analysis. It helps existing models to correctly reflect bridging links between communities by synchronizing meetings at social locations. We first explain the details of this Social Overlay (Section 6.1) and then apply it to TVCM (creating TVCM:SO in Sec. 6.2) and HCMM (creating HCMM:SO in Sec. 6.3)[16]. Evaluating contact traces from these two extended models, we will see that the social overlay indeed creates links with a small spread, while qualitatively maintaining other properties (e.g., inter-contact time distributions) of the original models.

6.1 Social Overlay

We start by discussing the Hypergraph model and how to calibrate it from contact traces. This step is not specific to the model to which the SO is applied. Then, we explain a generic set of steps to be implemented in order to apply the SO to a model.

A Hypergraph [35] (also called *Levi graph*) is a generalization of a Graph $G(N,E)$, with a set of nodes N and a set of edges E. In a Hypergraph $H(N,E)$, a Hyperedge (or just edge) $e_i \in E$ does not correspond to only a *pair* of nodes, as in a graph. Instead, an edge is a *list* of an arbitrary number of nodes, i.e., an arbitrary subset of N (not the empty set $\emptyset$). The *edge cardinality* $|e_i|$ is the number of nodes e_i connects (thus, a "simple" graph, is a Hypergraph where all edge cardinalities are 2). As in a simple graph, in a Hypergraph the *degree* of a node n is the number of edges that it is a member of. With the concept of a Hypergraph we can connect more than two nodes, which allows us to represent synchronized behavior of more than two nodes in a simple yet meaningful way.

In our social overlay Hypergraph, an edge represents a group of nodes, synchronously meeting outside their community locations. In order to be realistic, we need a Hypergraph model which repro-

[13]We have tried other threshold values, as well as statistical tests with the 95% and 99% quantile of the simulated random overlap and have found very similar results.

[14]Note that deducing group synchronization based on pairwise synchronization is only an approximation. For example it may happen that in a triangle of nodes which are all pairwise synchronized, all three of them never actually visit a location together.

[15]Note that up to a value of about 15 cliques per node, the geometric model is very good. Beyond 15, the tail seems a bit "heavier". However, we still believe that a geometric distribution is a good approximation for our purposes.

[16]Since the social overlay models bridges *between* communities, it only makes sense to apply it if the original model already creates modular contacts. Hence, we do not apply it to SLAW, as it does not manifest community structure (see Sec. 3).

duces the edge cardinality and node degree distribution we measure in a contact trace. As we have seen in Sec. 5, for the DART trace, both, edge cardinality and node degree are well approximated by geometric distributions. However, the following general procedure can also be applied using other distributions.

Given a edge cardinality and a degree distribution, our model works similar to the *configuration model* [36] for graphs. We connect the nodes (the number of nodes is given by the scenario we want to simulate) with Hyperedges in the following steps: (1) For each node, draw a degree from the degree distribution (geometric with $p_1 = 0.68$ in our example), and attach the according number of "stubs" to it. (2) Create a new Hyperedge of cardinality drawn from the respective distribution (geometric with parameter $p_2 = 0.28$ in our example). (3) Choose from the stubs at random to select which nodes are members of the edge (a node can not be assigned to the same Hyperedge twice). Once a stub is selected, it is connected and not eligible any longer. Steps 2 and 3 are repeated until no more stubs are left. This creates a random Hypergraph with the specified degree distribution and edge cardinality distribution[17].

The creation of the Hypergraph is the same for any mobility model to which the social overlay is applied. Given the overlay, a mobility model then must implement the following three general steps in order to create synchronized meetings of the nodes of an edge. Note that we here only explain the general idea, the concrete mechanisms can be chosen specifically for a model (Sec. 6.2 and 6.3 discuss two examples).

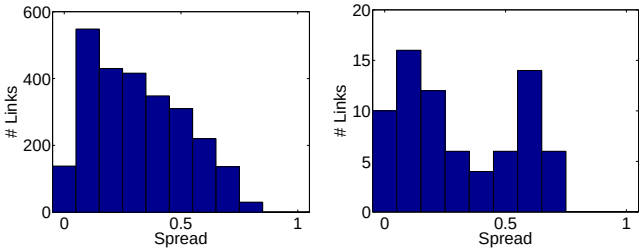
Step 1) Location Assign each Hyperedge to a random location on the simulated area (as the location where the members of the edge meet). Depending on the model, this may be e.g. an area or cell.

Step 2) Time For each Hyperedge, define an social activity period during which the edge counts as active. This is the time interval during which the members of the edge meet at the respective location. The social activity periods of adjacent edges (edges sharing one or more nodes) should not overlap. Assigning a given number of non-overlapping activity periods to edges, is an *edge coloring problem*. Note that the the smallest number of colors needed for a valid coloring (i.e., the edge chromatic number) must be smaller than the number of activity periods available.

Step 3) Synchronization When no edge of a node is active, it moves according to the normal rules of the mobility model. As one of its edges is active (during the social activity period), the behavior of the node changes such that it visits the location of the respective edge, thereby synchronously meeting the other members of the edge. The rules according to which a node moves during the social activity period of one of its edges can again be chosen specifically for a certain model (two clarifying examples will be given in Sec. 6.2 and 6.3).

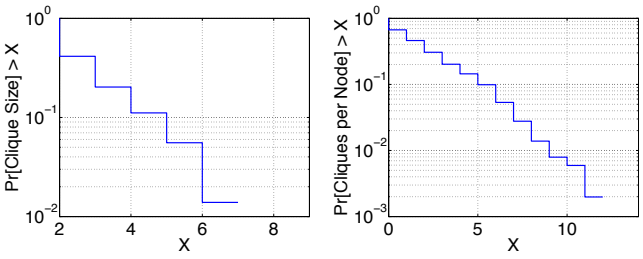
We deliberately keep these points very generic in order to make them applicable to many different mobility models. However, there are certain prerequisites for a model in order to make the SO applicable. First, the model must have distinguishable nodes, so that we can assign them to nodes of the Hypergraph. While this is the case for the majority of mobility models, there may be models for which nodes are not distinguishable (e.g., [8] which models the size distribution of clusters without accounting for *which* nodes are part of a cluster). Second, the mobility model must have a notion of time and location in order to assign the activity period and location to the Hyperedges. This is also fulfilled by the majority of models,

[17]At the end, it may happen that there are not enough stubs of different nodes left. In that case, we simply decrease the cardinality of the edge to fit the number of remaining stubs. This should only have insignificant influence on the sample of edge cardinalities.



(a) TVCM:SO (m=0.31). (b) HCMM:SO (m=0.35)

Figure 8: Edge spread of bridging links.



(a) Clique Size CCDF. (b) Cliques per node CCDF.

Figure 9: Synchronization of node locations (TVCM:SO).

exceptions being for instance pure contact models which do not care where and/or when nodes meet, but merely *that* they meet.

6.2 TVCM:SO

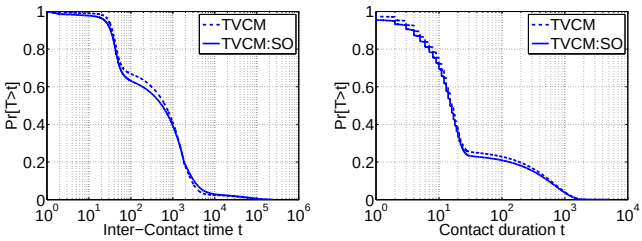
In the following, we describe the application of the social overlay to TVCM. As a basis, we use the scenario described in Section 2. For the social overlay, we create a Hypergraph of 505 nodes according to the parameters and configuration model described above (geometric distributions of edge cardinality and node degree with parameters p_1 and p_2). Having the Hypergraph, what remains is to implement the three steps described in Sec. 6.1. We use TVCM specific mechanisms to do so (i.e., the grid of the simulation scenario and Markovian node behavior) in order to modify the simulator and the original characteristics of TVCM as little as possible.

Step 1) Location Since in the scenario described in Sec. 2 the simulation area is already split into a grid of 6×6 areas, we reuse these areas and assign one of them to each edge of the Hypergraph. In order to ensure to create meetings *outside* home locations, communities' home locations are not eligible for assigning edges[18].

Step 2) Time To assign the edges to social activity periods, we split every 24 Hours of simulation time in non-overlapping periods of $3'600$ sec. (we empirically choose 1h as a typical duration of social activities). To avoid two edges of a node being active at the same time, we color edges such that no two edges of a node have the same color. Each color is then assigned such a 1h period every 24h, during which all edges of the respective color are active.

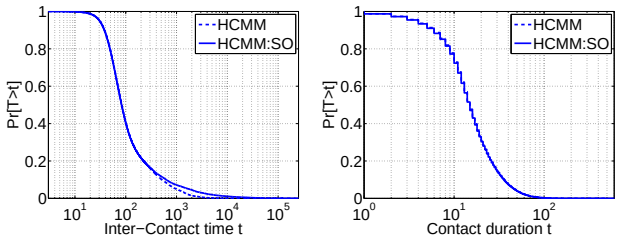
Step 3) Synchronization To make the nodes of an edge meet we replace the 2-state Markov Chain (described in Sec. 2) of the nodes during edge social activity periods of their edges: Instead of planning the next trip at its home location with probability p (and roaming with probability $1 - p$), the node plans a trip to the location of the active edge. Once arrived there, it moves within the edge location area according to normal TVCM rules (i.e., random

[18]Note that several edges can be assigned to the same location. Since edges of different colors have different activity periods (see point ii), chances that different edges overlap in time at the same location are small.



(a) Inter-contact time CCDF. (b) Contact duration CCDF.

Figure 10: Pairwise statistics for TVCM:SO.



(a) Inter-contact time CCDF. (b) Contact duration CCDF.

Figure 11: Pairwise statistics for HCMM:SO.

waypoint). After the social activity period, the next trip is again planned according to the "regular" 2-state Markov Chain.

To validate the social overlay for TVCM, we run the TVCM:SO model and analyze the resulting contact trace. We want to ensure that the described mechanisms create realistic social meetings outside the communities' home locations. Additionally, we verify that the original typical characteristics of TVCM (e.g., inter-contact time and contact duration distribution) do not change qualitatively.

As a first step, we measure the spread of the links between communities, to see whether TVCM:SO creates the desired bridging links. Fig. 8a (compare to TVCM in Fig. 5c) shows that indeed we now observe links with small spread. The mean value of the spread is with $m = 0.31$ comparable to the values we find in traces (see Fig. 5c). This is a first evidence that we now obtain more realistic bridges between communities.

In order to check whether the synchronization mechanism described above indeed synchronizes the social meetings as desired, we repeat the measurement of Sec. 5, using the trace obtained from TVCM:SO. We compute the overlap of time spent in areas (on the 6×6 grid of our scenario) which are not home locations to a community. For each of these locations, we determine the synchronicity graph and determine the cliques. This gives us the clique sizes and the number of cliques of which a node is part. Fig. 9 shows the respective distributions. From the straight lines we conclude that they follow geometric distributions and the Kolgomorov-Smirnov tests does not reject this at a significance level of 0.05. The parameters we get from a maximum likelihood estimator are $p_1 = 0.62$ and $p_2 = 0.28$, thus, they match the configured values for the social overlay very well. Hence, we conclude that the TVCM:SO indeed creates realistic meeting patterns outside community locations.

The question remains how much we have changed the general characteristics of the model by our modification. To check this, we compare the pairwise inter-contact time and contact duration distributions of the original TVCM to TVCM:SO. Fig. 10 shows that the CCDFs of both features follow each very closely. Note that comparing the pairwise statistics to the traces is out of scope of this paper. We assume that the models to which we apply the social overlay are designed to realistically reproduce such features (for TVCM a detailed evaluation can be found in [5]).

Further, we verify that the social overlay does not distort the community structure of the original TVCM. Detecting the communities in the TVCM:SO trace, we find that nodes are assigned to the same communities as before. However, we notice that the modularity was reduced a bit, from $Q = 0.73$ to $Q = 0.70$. This reduction is explicable, since the social overlay takes the nodes out of their "daily routing" (which creates communities) and creates bridges. However, a modularity of 0.7 is still high and comparable to the modularity of the measured traces.

Thus, we conclude that TVCM:SO creates more realistic community interfaces than TVCM, while maintaining the original TVCM characteristics.

6.3 HCMM:SO

As a second proof of concept, we apply the social overlay to HCMM, using HCMM specific mechanisms to create the synchronized social meetings. Again, we base the our scenario on the one described in Sec. 2, and create a Hypergraph of 100 nodes with the parameters determined in Sec. 5. We use the following mechanisms to integrate the social overlay into HCMM.

Step 1) Location In HCMM, the simulated area is divided into *home-cells*, in our case a grid of 6×6 cells. We reuse this structure by assigning Hyperedges to home-cells.

Step 2) Time As in TVCM, we divide every 24h of simulation time into 1h periods and assign edges to them.

Step 3) Synchronization In HCMM, the nodes plan trips to cells other than their home-cells based on node specific attraction values. During normal operation, these attraction values are based on the number of "friends" (in the social caveman graph) a node has in a given cell. During the social activity period of an edge, we increase the attraction of the respective cell for the edges' nodes to 1 and set the attraction for all other cells to 0. Consequently, the nodes will plan their next trip to the respective cell and meet there. Once the social activity period is over, the attractions are again set according to normal HCMM rules.

Fig. 8b (compare to Fig. 5d) shows the edge spread of the strong inter-community links of HCMM:SO. We observe that the social overlay creates links with small spread as desired. The mean of the spread is 0.35, similar to the one for TVCM:SO, and comparable to the traces (e.g., ETH with 0.31). Again, we also compare the inter-contact time and contact duration distributions of HCMM and HCMM:SO, to verify that the social overlay does not interfere to much with the original HCMM characteristics. Fig. 11 shows that the contact duration distribution is maintained (the two curves are not distinguishable because they overlap) and also the the inter-contact time distributions are very close to each other. Similarly to TVCM:SO, we also observe a reduction of modularity from 0.76 to 0.67 which is, however, not severe as the value is still in the range of what we observe in the traces.

In conclusion, we have seen for two models, how the social overlay can be implemented as an extension, creating realistic inter-community interfaces while largely maintaining the properties of the original model. We believe that the social overlay can and should be integrated also in other mobility models.

Finally, we want to mention that the shortcoming of the models under scrutiny may not be very surprising – after all they were most likely not designed with inter-community interfaces in mind. However, we argue that it is time to pay more attention to this issue when running simulations or creating mobility models. In the past, mobility models had to master the benchmark of creating realistic inter-contact time and contact duration distributions, we believe that in the future, other statistics such as edge spread between communities and cluster sizes for social meetings should not be neglected. The proposed social overlay is a first step in this direction.

7. CONCLUSIONS

Comparing the contact graphs of various measured contact traces and several mobility models, we have identified that mobility models do not correctly reflect the way communities are connected to each other: While traces tend to manifest *bridging links* between communities (i.e., *narrow* interfaces of only few strong edges connecting two communities), models tend to connect communities by *bridging nodes* (i.e., broader interfaces where community members are linked to many other nodes in the peer community).

To explain this difference, we have performed a detailed analysis of contact data, focusing on context (e.g., home, work), location and timing of contacts, discriminating between intra-community and inter-community contacts. This analysis provides notable insights into the nature of mobility and contacts: (i) Inter-community contacts and intra-community contacts happen in different contexts. (ii) Bridging links tend to happen outside the home location of the respective communities, whereas bridging nodes tend to visit the location of the peer community. (iii) Inter-community contacts happening outside community locations are often not random meetings but synchronized visits of two or more nodes. From experience with mobility models, we know that such synchronized meetings of pairs or small groups across community boundaries (but happening outside the realm of the respective communities) are usually not modeled, hence the different community interfaces.

To model such meetings of pairs or groups, we have proposed a Hypergraph model, where edge degree and node degree distributions can be calibrated from a contact trace. Assuming that a given mobility model correctly reflects communities (but fails to correctly model the community interfaces) this Hypergraph can be used as an extension which we call social overlay. With the examples of TVCM:SO and HCMM:SO we have demonstrated the application of the social overlay to two models and shown that it introduces bridging links in the contact graphs of both models, while largely maintaining other features of the original model.

In the future, we intend to automate the process of creating the Social Overlay for a specific scenario (e.g., trace) that needs to be emulated by a synthetic model. Further, we plan to investigate the impact of different inter-community interfaces (bridging nodes or bridging links) on the performance of dissemination processes.

Acknowledgments

We thank Cristian Tuduce for providing us with the ETH access point trace. Further, we thank our shepherd Randall Berry and the anonymous reviewers for their advice in improving the paper.

8. REFERENCES

[1] T. Camp, J. Boleng, and V. Davies, "A survey of mobility models for ad hoc network research," *WCMC*, 2002.

[2] A. Chaintreau, P. Hui, J. Crowcroft, C. Diot, R. Gass, and J. Scott, "Impact of human mobility on the design of opportunistic forwarding algorithms," in *IEEE Infocom*, 2006.

[3] C. Song, Z. Qu, N. Blumm, and A.-L. Barabási, "Limits of predictability in human mobility," *Science*, 2010.

[4] A. J. Nicholson and B. D. Noble, "Breadcrumbs: forecasting mobile connectivity," in *ACM MobiCom*, 2008.

[5] W. J. Hsu, T. Spyropoulos, K. Psounis, and A. Helmy, "Modeling time-variant user mobility in wireless mobile networks," in *IEEE Infocom*, 2007.

[6] C. Boldrini and A. Passarella, "Hcmm: Modelling spatial and temporal properties of human mobility driven by users' social relationships," *Elsevier Computer Commununication*, 2010.

[7] K. Lee, S. Hong, S. J. Kim, I. Rhee, and S. Chong, "Slaw: A mobility model for human walks," in *IEEE Infocom*, 2009.

[8] S. Heimlicher and K. Salamatian, "Globs in the primordial soup: The emergence of connected crowds in mobile wireless networks," in *ACM MobiHoc*, 2010.

[9] Delay tolerant networking research group. [Online]. Available: http://www.dtnrg.org

[10] V. Conan, J. Leguay, and T. Friedman, "Characterizing pairwise inter-contact patterns in delay tolerant networks," in *ACM Autonomics*, 2007.

[11] H. Cai and D. Y. Eun, "Crossing over the bounded domain: from exponential to power-law inter-meeting time in manet," in *ACM MobiCom*, 2007.

[12] T. Karagiannis, J.-Y. Le Boudec, and M. Vojnovic, "Power law and exponential decay of inter contact times between mobile devices," in *ACM MobiCom*, 2007.

[13] P. Hui, J. Crowcroft, and E. Yoneki, "Bubble Rap: Social-based forwarding in delay tolerant networks," in *ACM MobiHoc*, 2008.

[14] T. Hossmann, T. Spyropoulos, and F. Legendre, "A complex network analysis of human mobility," in *IEEE NetSciCom*, 2011.

[15] E. M. Daly and M. Haahr, "Social network analysis for routing in disconnected delay-tolerant MANETs," in *ACM MobiHoc*, 2007.

[16] W. Gao, Q. Li, B. Zhao, and G. Cao, "Multicasting in delay tolerant networks: a social network perspective," in *ACM MobiHoc*, 2009.

[17] T. Hossmann, T. Spyropoulos, and F. Legendre, "Know thy neighbor: Towards optimal mapping of contacts to social graphs for DTN routing," in *IEEE Infocom*, 2010.

[18] T. Henderson, D. Kotz, and I. Abyzov, "The changing usage of a mature campus-wide wireless network," in *ACM MobiCom*, 2004.

[19] C. Tuduce and T. Gross, "A mobility model based on WLAN traces and its validation," in *IEEE Infocom*, 2005.

[20] N. Eagle, A. Pentland, and D. Lazer, "Inferring Social Network Structure using Mobile Phone Data," *PNAS*, 2009.

[21] M. Musolesi and C. Mascolo, "A community based mobility model for ad hoc network research," in *ACM REALMAN*, 2006.

[22] M. E. J. Newman, "The structure and function of complex networks," March 2003.

[23] I. Rhee, M. Shin, S. Hong, K. Lee, and S. Chong, "On the levy-walk nature of human mobility." in *IEEE Infocom*, 2008.

[24] H. Dubois-Ferriere, M. Grossglauser, and M. Vetterli, "Age matters: efficient route discovery in mobile ad hoc networks using encounter ages," in *ACM MobiHoc*, 2003.

[25] A. Scherrer, P. Borgnat, E. Fleury, J. L. Guillaume, and C. Robardet, "Description and simulation of dynamic mobility networks," *Elsevier Computer Networks*, 2008.

[26] S. Fortunato, "Community detection in graphs," *Phys. Reports*, 2010.

[27] E. Jaho, M. Karaliopoulos, and I. Stavrakakis, "Social similarity as a driver for selfish, cooperative and altruistic behavior," in *AOC*, 2010.

[28] V. D. Blondel, J.-L. Guillaume, R. Lambiotte, and E. Lefebvre, "Fast unfolding of communities in large networks," *J.STAT.MECH.*, 2008.

[29] M. E. J. Newman, "Modularity and community structure in networks," *PNAS*, 2006.

[30] A. Y. Ng, M. I. Jordan, and Y. Weiss, "On spectral clustering: Analysis and an algorithm," in *Advances in Neural Information Processing Systems 14*. MIT Press, 2001.

[31] T. Hossmann, T. Spyropoulos, and F. Legendre, "Social network analysis of human mobility and implications for dtn performance analysis and mobility modeling," ETH Zurich, Tech. Rep., 2010.

[32] C. Boldrini, M. Conti, and A. Passarella, "Contentplace: social-aware data dissemination in opportunistic networks," in *ACM MSWiM*, 2008.

[33] L. A. Cutillo, R. Molva, and T. Strufe, "Safebook : a privacy preserving online social network leveraging on real-life trust," *IEEE Communications Magazine*, 2009.

[34] A. Picu and T. Spyropoulos, "Minimum expected *-cast time in DTNs," in *Bionetics*, 2009.

[35] C. Berge, *Graphs and Hypergraphs*. Elsevier, 1985.

[36] W. Aiello, F. Chung, and L. Lu, "A random graph model for massive graphs," in *STOC*, 2000.