Plausible Mobility Inference from Wireless Contacts Using Optimization

Alireza K. Monfared
Georgia Tech
alireza@gatech.edu

Mostafa H. Ammar
Georgia Tech
ammar@cc.gatech.edu

Ellen W. Zegura
Georgia Tech
ewz@cc.gatech.edu

ABSTRACT

Studies of mobile wireless protocols benefit from real-world traces that measure and record node locations over time. Such traces form an essential part of evaluation studies for many types of mobile and wireless networks, including but not limited to opportunistic networks.Mobility information provides the possibility of richer simulations in such studies, while contact information, irretrievably, loses considerable details like the exact locations of the nod es and their velocities. Despite this, contact information can be measured more easily compared to detailed locations of the nodes. We bridge the gap between relatively more available contact traces and potentially more useful mobility traces by presenting a solution for inferring *plausible mobility* from contact information. Our technique uses a simple non-linear optimization approach to formulate this as a feasibility problem with fundamental constraints on the mobility of the nodes. We introduce our technique and its underlying assumptions, and evaluate its performance in recovering mobility from synthetic and real traces. We also show that our approach brings orders of magnitude improvements in inference performance compared to state of the art.

Categories and Subject Descriptors

C.2 Computer-communication Networks [**Network Architecture and Design**]: [Network communications; Network topology; Wireless communication]

Keywords

Plausbile mobility; mobility traces; contact traces; mobility inference

1. INTRODUCTION

Studies of mobile wireless protocols benefit from real-world traces that measure and record node locations over time. These *mobility traces* can be combined with radio models to produce network connectivity for simulation studies. Measuring node locations with fine-grained time and space granularity is challenging and hence the number of real world datasets on mobility that are publicly available are not significant. Examples of mobility traces are reported by Rhee et al. [15] with scenarios in a university campus and in public places and by Banerejee et al. [6] for public transit buses in a campus. Outdoor positioning systems are sensitive to weather and obstacles, while indoor positioning systems generally rely on availability of WiFi or other signal infrastructure that may not be available.

As an alternative, *contact traces* measure and record pairwise node connectivity over time and do not require accurate positioning systems. Since they come from radio transmissions, contact traces are able to capture environmental effects on communication capability such as obstacles. These effects are mostly missing in mobility traces. Contact traces record on-off timing information about links, hence they can be more compact than mobility traces that record location data at every time instant.

Contacts are typically recorded by using Bluetooth or WiFi in an infrastructure-less mode such that all devices in range are seen as a contact. In such experiments, mobile devices equipped with a wireless interface are attached to moving entities and sample their environments periodically and log the presence of neighboring nodes. At the conclusion of the experiment these logs are collected and a contact trace is constructed from them. Contact trace collection and analysis is reported in the literature and is used extensively to study mobile networks in various settings ranging from a campus scenario [12][19] to a conference [18] to urban scenarios [14] or a disaster area [4]. Aschenbruck et al. [5] summarize these datasets. While contact trace collection is easier than mobility trace collection, it is still considered a challenging task. It requires instrumenting nodes with collection capability and conducting a well-planned and coordinated collection "run".

Despite collection difficulties, contact traces remain more available to researchers than mobility traces. However, mobility traces offer an opportunity for more detailed simulations and can be used to explore new scenarios, for example by combining mobility with alternative radio models to generate new contact traces. For these reasons, we are interested in the problem of inferring a *plausible mobility trace* from a contact trace. A *plausible mobility trace* is one that can produce the given contract trace. In general, there are many mobility traces that are consistent with a given contact trace, and it is impossible to reconstruct exact locations.

CHANTS'13, September 30, 2013, Miami, Florida, USA

http://dx.doi.org/10.1145/2505494.2505501.

However, we believe that exact locations are not required in many experiments that deal with studying mobility. Our research explores the effect of this loss of accuracy in relative locations that results from our inference algorithm.

Few solutions are proposed for this specific problem. Wang et al.[20] and Ristanovic et al. [16] describe the inference problem and suggest some approaches. These works are evidence that the problem is of interest to the community but neither explores the solution in detail.

Whitbeck et al. [21] propose a complete algorithm for the mobility inversion problem. This algorithm is inspired by the work in dynamic graph drawing and is based on the concepts of physical forces. The goal of the algorithm is to satisfy three fundamental constraints on the mobility of the nodes. These constraints limit the speed of nodes and their distances as they setup and lose contact. To do this, the authors define three forces on each node based on their contact history and future; it is the equilibrium of these forces that gives a relative set of locations satisfying the given contacts.

A major disadvantage of the force-based algorithm is its dependence on many parameters that need to be tuned properly in order for the algorithm to provide accurate contact to mobility transformation. These include the rigidity constant, the damping factor, the intensity constant, cutoff distance, etc that are used in the calculation of the forces. There is no general rule for tuning these parameters for the algorithm. Our work is motivated by the need to find a contact to mobility transformation algorithm that will work without the need for such parameter tuning.

Our proposed algorithm uses an approach based on formulating plausible mobility inference as a feasibility problem for a small set of fundamental constraints on mobility of the nodes. Our algorithm provides more accurate results and is more robust against choices of parameters

We describe the details of our algorithm in this paper.

The main contributions of our work are as follows:

- We propose a formulation of the mobility inference problem as an optimization problem that satisfies a set of proximity and mobility constraints derived from the contact trace. To our knowledge, this is the first mobility inference algorithm based on optimization. An advantage of our approach is that it is parsimonious; in comparison, the best alternative [21] has many parameters that must be tuned to achieve good results.

- We evaluate the accuracy of our solution using metrics that range from low-level location comparisons to high-level packet delivery comparisons.

- We make the code of our algorithm publicly available [3].

The rest of this paper is organized as follows. Section 2 describes our algorithm with technical details. Section 3 presents the methodology and results of the evaluation of the algorithm on a range of synthetic and realistic datasets. Finally, Section 4 concludes our work, and mentions the plans to expand it in the future.

2. MOBILITY INVERSION ALGORITHM

Terminology: Let N be the number of nodes that reside in a $d = 2$ dimensional $L \times L$ square field that consists of T time instances. A contact trace in this setting corresponds to an $N \times N \times T$ matrix. This representation implies that we have access to snapshots of the connectivity structure of nodes at time instants $\mathcal{T} = \{t_o, ..., t_{T-1}\}$ where $|\mathcal{T}| = T$, i.e. we know the connectivity structure for T discrete snapshots[1]. A mobility trace is a $d \times N \times T$ matrix, X, where its element $x_i^k = (:, i, k)$ is the location of node i at time t_k.[2]

Our algorithm does not use any information about initial locations or the underlying mobility behavior of the nodes. The only assumptions made are the following:

- At any moment $t_k \in \mathcal{T}$ in time, two nodes i and j are in contact once their distance, $||x_i^k - x_j^k||_2$ is less than a given radio range minus a margin, $R(1 - \epsilon)$, and are out of contact if this distance is larger than $R(1 + \epsilon)$ (modified circular transmission range radio model).

- Nodes move slower than a given maximum speed v. This means that the distance between the location of a node i at time t_k and t_{k-1} ($t_k \in \mathcal{T} \setminus \{t_0\}$), which is represented as $||x_i^k - x_i^{k-1}||_2$ is smaller than the maximum possible distance that can be traveled, namely $v \times \Delta t$, where Δt is the time difference between snapshots.

- Nodes are always confined inside the $L \times L$ mobility field.

Note that in the above algorithm, we have used $R(1 \pm \epsilon)$ as tighter bounds for connection/disconnection constraints. This assures that the algorithm does not make border mistakes, i.e., it does not place two connected nodes at a distance slightly larger than R as a sacrifice to satisfy other constraints. Our complete results in the technical report version this work [13] shows that a small value above 0 makes sure such mistakes are avoided and noticeably improves the performance. As a result, we set ϵ to 0.1 as a fixed parameter.

Algorithm Description: We use the notation $G = (V, E)$ for the evolving graph corresponding to the input contact trace matrix, C. In this notation $V = \{1, 2, \ldots, N\}$ is the set of all nodes. This evolving graph is a time-series of ordinary graphs $G^k = (V, E^k)$. An edge $e_{ij}^k \in E^k$ represents a connection between nodes i and j at time instant t_k. We associate graph G, with its adjacency matrix C, i.e., each of the graphs G^k has an adjacency matrix $C^k = C(:, :, k)$.

A High-level description of our algorithm is illustrated in Figure 1. Our algorithm starts by finding an initial set of locations for the nodes that is consistent with the initial state of the contact trace at time t_0, namely G^0 with adjacency matrix $C(:, :, 0)$. After this initial step, to find the locations

[1]Note that most real traces are in the form of a "continuous" contact trace in which start and stop times of contact events are recorded with some precision. Such contact traces can be transformed into our desired discrete notation by capturing connectivity structure of nodes at discrete time instants with a uniform step of $\Delta t = t_k - t_{k-1}$. The value of Δt needs to be chosen carefully to keep the contact trace reasonably small in size while capturing all contact events that are long enough.

[2]Throughout this paper, we use the notation of MATLAB-like colons,$(..., :, ...)$, to represent specific dimensions in a multidimensional matrix. For example $A(:, 2)$ corresponds to the second column in the matrix A.

of the nodes at time instants $t_1, \ldots, t_{T-1}$, we solve a feasibility problem[3] to find a set of locations for each of the snapshots in the contact trace corresponding to these time instants. This feasibility problem ensures consistency with the three fundamental assumptions mentioned earlier. We use the location of nodes at the previous step as the initial point for the next step, and solve the problem iteratively to find the location of the nodes at each of the snapshots. This process is described in Algorithm 1.

Algorithm 1 Mobility inversion algorithm.

1: **Input:** Contact Trace $C_{N\times N\times T}$, Transmission range R, Maximum speed v
2: **Output:** Mobility trace X_{final}

3: **Initialization:**
4: Find a set of initial locations X_0.
5: $X_{final}(:,:,0) \leftarrow X_0$

6: **Optimization:**
7: **for** $k := 1$ **to** $T-1$ **do**
8: $\quad X_{init} \leftarrow X_{final}(:,:,k-1)$
9: $\quad$ Solve the nonlinear optimization problem in Figure
10: $\quad$ 1 with $C^k = C(:,:,k), X_{init}, R, v$ to find X
11: $\quad X_{final}(:,:,k) \leftarrow X$
12: **end for**

13: **Return** X_{final}

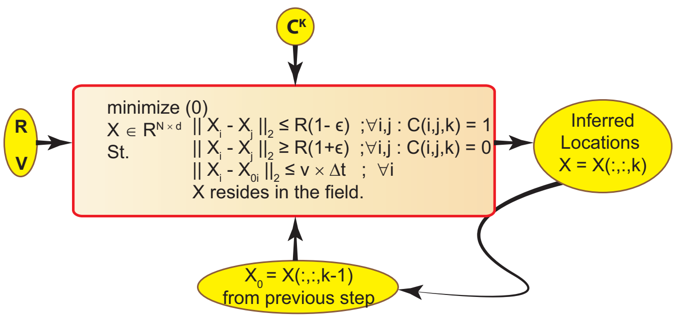


Figure 1: Optimization-based algorithm takes an initial point ($X_{init} = X(:,:,k-1)$), a contact trace ($C^k = C(:,:,k)$) along with its corresponding radio range (R) and a value of maximum speed (v) to infer a mobility ($X = X(:,:,k)$) in an iterative manner. Inferred locations for each time step are given as an initial point to the solver for the next step. This process is repeated for $t = t_0, \ldots, t_{T-1}$ (lines 7-11 in Algorithm 1).

Choice of the Initial Value: To choose a suitable value for X_0, we use the same method as above, except that the maximum speed constraint is ignored. Since the solver still needs an initial point to seek the stationary point of the subproblem solved at the first iteration of the algorithm (line 5 in Algortihm 1), we construct a distance matrix by finding the shortest paths among all node pairs in terms of the number of hops using the initial connectivity structure, G^0. Next, we multiply this hop-count matrix by the transmission range R, i.e., we assume two nodes m hops from each other are at a distance $m \times R$, and replace unreachable node pairs with a reasonably large distance. Finally, we apply the classical Multidimensional Scaling (MDS) [8] on this matrix to yield a decent initial point.

[3]We have tried various objectives to solve the problem as a complete optimization problem. Examples include minimizing nodes movement and minimizing node deviation from the center of the field. Despite this, our simulations show that the best results are achieved when the problem is stated as a feasibility problems. Besides this, there is no logical justification for an objective to be minimized in this problem.

3. PERFORMANCE EVALUATION

3.1 Methodology

Similar to the work in [21] our evaluation is based on using the *"original" contact traces* derived from known *original mobility traces*. Our algorithm is applied to produce an *inferred mobility trace* from this contact trace. This in turn is used to derive an *inferred contact trace* using the known radio range. This process is illustrated in figure 2. We can compare these traces in different ways. We use three levels of comparison.

Mobility-level Comparison: Assuming that we have access to the mobility trace from which the input contact trace originated. We can compare the original mobility trace and the inferred mobility trace for each location. As a direct comparison of absolute locations will not bear much information, we compare the distance between each node-pair at each time step for both traces in a two-dimensional histogram.

Contact-level Comparison: This comparison evaluates the accuracy of our algorithm in replicating the original contact trace. We compare the evolving graphs corresponding to these two contact traces over time based on the number of *missing links*, i.e., links present in the input contact trace and missing in the inferred contact trace, *extra links*, i.e., links only seen in the inferred contact trace, and the sum of these two numbers.

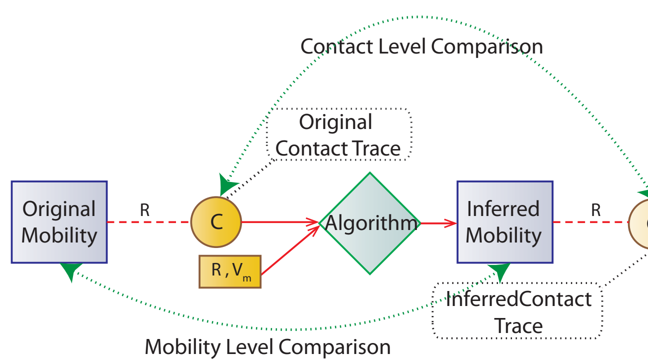


Figure 2: An illustration of concepts of original trace, inferred trace, original contact trace, and inferred contact trace. This figure explains which two traces are compared in mobility level and contact level comparisons.

Packet Delivery Ratio: This comparison involves running a simulation on the original and the inferred traces which is performed using the Opportunistic Network Environment simulator (ONE) [9]. In these experiments, we send random packets with a specific time to live (TTL) from a randomly chosen node destined for another randomly chosen node at every time step. Then, we compare the cumulative number of messages that reach their destinations before the TTL to total number of messages sent (packet delivery ratio) up until the current time step.

3.2 Results

Traces: We use three traces, two synthetic and one real, to evaluate our algorithm.

- *Random Waypoint (RWP)*[7], which is often used for simulation purposes in mobility literature despite its known limitations[22]. In this model speed, direction, and destination of each node is chosen randomly and independently of other nodes. Our RWP instance consists of 50 nodes with a duration of 1000 seconds with snapshots taken every second, spanning over a $1000 \times 1000\, m^2$ field. Maximum speed is $10m/s$ and transmission range is set to $100m$.

- *Self-similar Least Action Walk (SLAW)*[10], which tries to capture statistical patterns of human mobility including truncated power-law distribution of flights, pause-times and inter-contact times, and heterogeneously defined areas of individual mobility. One of the main features of this model is that it captures social contexts. The model is heavily based on GPS traces of human walks, including 226 daily traces collected from 101 volunteers in five different outdoor sites. Our instance of SLAW consists of 50 nodes simulated over 10 hours with snapshots taken every minute. The simulation field is $2000 \times 2000\, m^2$; maximum speed is $10m/s$ and transmission range is $60m$. Other important parameters of the SLAW are 0.75 for the self-similarity of waypoints (on a scale of 0 to 1), a minimum (maximum) pause of 30 seconds (1 hour) for the individuals chosen to match human mobility in a social setting.

- *Haggle INFOCOM 2005 (Haggle)*[17] ,which is collected in an experiment conducted during the IEEE INFOCOM 2005 conference in Miami with 41 iMotes carried by attendees for 3 to 4 days. Based on this trace, we find a contact trace with snapshots updated every 500 seconds, resulting in a trace of 1000 time steps. Maximum speed is set to $1.7m/s$ (typical human walking speed) and transmission range to $100m$, following simulations in Whitbeck et al[21].

Mobility-level Comparison Results: Figure 3 shows the histogram for pairwise distances normalized to the value of transmission range. We do not expect to see a perfect correlation between the original and inferred distances since the algorithm does not have any auxiliary information about the locations of the nodes. Instead, we observe a trend of large distances versus large distances and small distances versus small distances in the original and inferred trace respectively. Note that in the ideal case the histogram should have high values along its diagonal.

As the figures suggest, for RWP (Figure 3a), most of the distances are mapped accordingly in the inferred trace with a distribution of errors around the diagonal. For SLAW (Figure 3b), most of distances are small and mapped correctly (notice the dark areas in the bottom left)and for a few larger distances(the lighter areas in the upper half of the figure) performance deteriorates. We do not have any results for Haggle in this section, since Haggle is a real contact trace and does not include the original locations of individuals.

Contact-level Comparison Results: Figure 4 shows the contact-level errors. For a more meaningful comparison, we use the force-based heuristic of Whitbeck et al., [21] to infer the same trace. We use the default values in the force-based heuristic for inference. To preserve the readability of the figure, we plot the decomposed missing, extra, and total link errors for our optimization-based and only the total link errors for the force-based heuristic. For RWP, the figure suggests that most of the error is in the missing links which are present in the original contact trace but absent in the inferred contact trace. We observe that this proportion changes slightly with varying values of ϵ as it can put more emphasis on reducing extra link errors at the cost of increasing missing link errors and vice versa. The total link errors are less than 2%, which corresponds to $0.02 \times 1225 \simeq 24$ errors out of the 1225 possible links.

For SLAW, although the trace is based on a much more complicated model and is hours long in duration with update intervals of minutes, Figure 4b suggests errors lower than 1%, i.e., at most 12 links in error, almost all of which are missed connections. This is 3 times lower than the error of the force-based heuristic.

For Haggle, Figure 4c suggests that except for the transitory period in the beginning, error is less than 2% or 16 links. Without any knowledge about the patterns behind the mobility and even with rough estimates for the values of transmission range and maximum speed, this error is quite small and more than 8 times better than the average error of 17% of the force-based heuristic.

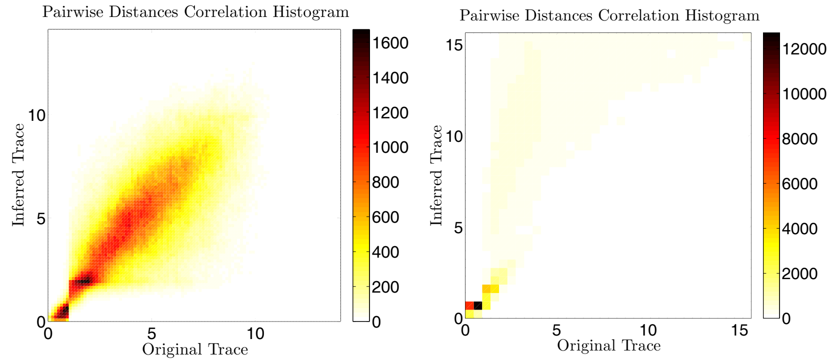
Packet Delivery Ratio Results: Figure 5 shows the packet delivery ratio, as defined above. In these experiments, we send a $1KB$ packet from a randomly chosen node to another node every second. The value of the TTL for messages is set to one-fifth of the trace length, e.g., 3 minutes for RWP. We use the Probabilistic ROuting Protocol using History of Encounters and Transitivity (PROPHET) [11] as the routing protocol for the simulation over simpler routing protocols like flooding in order to have a more realistic situation. We include the simulation results for the original trace, inferred trace using the optimization-based algorithm, and inferred trace using the force-based heuristic.

All figures suggest that the original and inferred traces show very similar behavior in packet delivery. They also suggest that the errors in Figure 4 do not significantly affect the packet delivery behavior. In all cases, force-based heuristic shows a higher packet delivery ratio, i.e., it has unnecessary links in the inferred trace (extra link errors). The inferred trace using optimization-based algorithm always performs more closely to the original trace than the force-based heuristic with the difference most obvious in Figure 5a and 5c.

Other Results

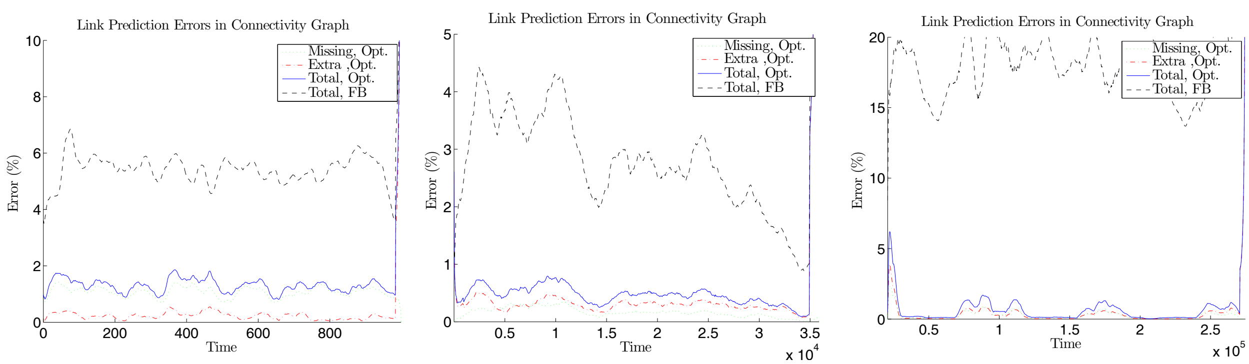
We have performed similar tests on other synthetic traces, as well as real traces including rollenet [1], sassy [2], and pmtr [12]. While we exclude the results for the sake of brevity, they further indicate that the optimization-based algorithm results in errors less than the force-based heuristic in all cases.

We have also performed tests on the robustness of our algorithm to the choice of transmission range, maximum



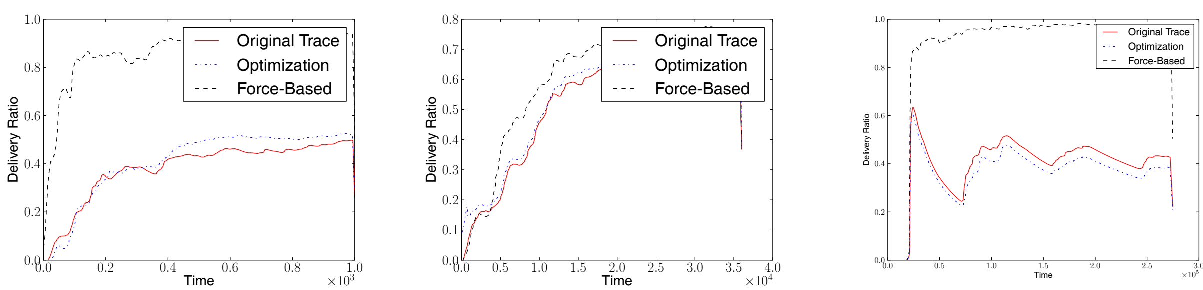
(a) RWP (b) SLAW

Figure 3: Pairwise distance histograms. Horizontal (Vertical) axis shows distance among node-pairs in the original (inferred) trace normalized by the value of the transmission range. Color intensities are proportional to the number of node-pairs having distances in the corresponding bin.



(a) RWP (b) SLAW (c) Haggle

Figure 4: Contact-level errors over time. Extra link errors (Extra) are missing in the original, but present in the inferred contact trace. Missing link errors (Missing) are present in the original, but missing in the inferred contact trace. Total link errors (Total) is the sum of these. Horizontal axis represents time and vertical axis shows the errors as percentage of total number of possible links. Figure includes all three types of errors for the optimization-based algorithm (Opt.) and the total link errors for the force-based heuristic(FB)



(a) RWP (b) SLAW (c) Haggle

Figure 5: Message Delivery Ration versus time. Vertical axis shows the total number of messages delivered until the current time divided by the total number of messages sent.

speed, and epsilon. The results are skipped due to space constraints and are available in the technical report version of the paper [13]. They further confirm that if the transmission range is not known exactly, which is the case for many real traces, a rough estimate given to the algorithm will still suffice. Also a choice of maximum speed that is non-zero and not too small, even if not accurate, suffices for the algorithm to predict the next locations of the nodes iteratively. Finally a non-zero value of epsilon that is not too big can result in a noticeable improvement in results. So the value can be set to 0.1 as fixed parameter rather than a user parameter.

4. CONCLUSION AND FUTURE WORK

We present an optimization-based algorithm to solve the problem of inferring mobility from contacts. Our algorithm accepts a contact trace that contains information about connectivity of nodes over time. Without any extra information or assumptions about the locations of these nodes, we infer a set of locations that could generate these contacts. Through extensive experiments with synthetic and real traces, we show that this inferred mobility has connectivity characteristics that are comparable to the original trace and can be used in simulations instead of the original contact trace.

As stated earlier, our algorithm, unlike the work in [21] has the important advantage that it works without needing cumbersome and error-prone parameter tuning. The optimization framework on which our algorithm is built has also additional promise for future extensions:

- In this work, we have assumed a simple circular radio range model, i.e., two nodes are in contact whenever their distance is less than a given transmission range and disconnected otherwise. In reality, a contact trace should be interpreted along with its corresponding radio model which specifies the circumstances under which two nodes are actually in contact and factors such as the existence of obstacles, fading and shadowing and inequality of signal strength inside the transmission range need to be taken into account. We believe our approach provides a suitable framework for incorporating a radio model since the feasibility constraints explicitly incorporate the radio range (see Figure 1). One idea would be to replace R in the constraints with a random number drawn from a distribution of radio ranges derived from a specific radio propagation model. We plan to explore this in our future research.

- An input contact trace induces a set of constraints in our algorithm. Because of this, our algorithm can take as input multiple contact traces which simply translates into multiple sets of constraints for which an approximate feasible solution can be obtained. We believe this will allow us to produce more accurate contact to mobility transformation. If this is confirmed it may indicate that trace collection exercises can enhance their usability if they are augmented to obtain multiple traces simultaneously.

5. ACKNOWLEDGMENTS

This material is based upon work supported by the National Science Foundation under Grant No. 1117190.

Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation.

6. REFERENCES

[1] CRAWDAD data set upmc/rollernet (v. 2009-02-02).

[2] CRAWDAD data set st_andrews/sassy (v. 2011-06-03). Downloaded from `http://crawdad.cs.dartmouth.edu/st_andrews/sassy`, June 2011.

[3] E. W. Z. Alireza K. Monfared, Mostafa Ammar. Optimization Algorithm for Plausible Mobility Inference. Accessible at `https://github.com/alirezamonfared/MobilityReport`, Apr. 2013.

[4] N. Aschenbruck, E. Gerhards-Padilla, and P. Martini. Modeling mobility in disaster area scenarios. *Performance Evaluation*, 66(12):773–790, 2009.

[5] N. Aschenbruck, A. Munjal, and T. Camp. Trace-based mobility modeling for multi-hop wireless networks. *Computer Communications*, 34(6):704–714, 2011.

[6] N. Banerjee, M. Corner, and B. Levine. An energy-efficient architecture for DTN throwboxes. In *INFOCOM 2007. 26th IEEE International Conference on Computer Communications. IEEE*, pages 776–784. IEEE, 2007.

[7] T. Camp, J. Boleng, and V. Davies. A survey of mobility models for ad hoc network research. *Wireless communications and mobile computing*, 2(5):483–502, 2002.

[8] G. Colliat. OLAP, relational, and multidimensional database systems. *ACM Sigmod Record*, 25(3):64–69, 1996.

[9] A. Keränen, J. Ott, and T. Kärkkäinen. The ONE Simulator for DTN Protocol Evaluation. In *SIMUTools '09: Proceedings of the 2nd International Conference on Simulation Tools and Techniques*, New York, NY, USA, 2009. ICST.

[10] K. Lee, S. Hong, S. J. Kim, I. Rhee, and S. Chong. SLAW: A New Mobility Model for Human Walks. In *INFOCOM*, pages 855–863. IEEE, 2009.

[11] A. Lindgren, A. Doria, and O. Schelen. Probabilistic Routing in Intermittently Connected Networks. In P. Dini, P. Lorenz, and J. N. de Souza, editors, *SAPIR*, volume 3126 of *Lecture Notes in Computer Science*, pages 239–254. Springer, 2004.

[12] P. Meroni, S. Gaito, E. Pagani, and G. P. Rossi. CRAWDAD data set unimi/pmtr (v. 2008-12-01). Downloaded from `http://crawdad.cs.dartmouth.edu/unimi/pmtr`, Dec. 2008.

[13] A. K. Monfared, M. H. Ammar, and E. W. Zegura. Plausible mobility inference from wireless contacts using optimization. Technical report, Georgia Institute of Technology, May 2013.

[14] A. Natarajan, M. Motani, and V. Srinivasan. Understanding urban interactions from bluetooth phone contact traces. *Passive and Active Network Measurement*, pages 115–124, 2007.

[15] I. Rhee, M. Shin, S. Hong, K. Lee, S. J. Kim, and S. Chong. On the levy-walk nature of human mobility. *IEEE/ACM Trans. Netw.*, 19(3):630–643, 2011.

[16] N. Ristanovic, D. Tran, and J. Le Boudec. Tracking of mobile devices through Bluetooth contacts. In *Proceedings of the ACM CoNEXT Student Workshop*, page 4. ACM, 2010.

[17] J. Scott, R. Gass, J. Crowcroft, P. Hui, C. Diot, and A. Chaintreau. CRAWDAD data set cambridge/haggle (v. 2009-05-29). Downloaded from `http://crawdad.cs.dartmouth.edu/cambridge/haggle`, May 2009.

[18] J. Scott, R. Gass, J. Crowcroft, P. Hui, C. Diot, and A. Chaintreau. CRAWDAD trace cambridge/haggle/imote/infocom2006 (v. 2009-05-29). Downloaded from `http://crawdad.cs.dartmouth.edu/cambridge/haggle/imote/infocom2006`, May 2009.

[19] V. Srinivasan, M. Motani, and W. T. Ooi. CRAWDAD data set nus/contact (v. 2006-08-01). Downloaded from `http://crawdad.cs.dartmouth.edu/nus/contact`, Aug. 2006.

[20] P. Wang, Z. Gao, X. Xu, Y. Zhou, H. Zhu, and K. Zhu. Automatic inference of movements from contact histories. In *Proceedings of the ACM SIGCOMM 2011 conference on SIGCOMM*, pages 386–387. ACM, 2011.

[21] J. Whitbeck, M. De Amorim, V. Conan, M. Ammar, and E. Zegura. From encounters to plausible mobility. *Pervasive and Mobile Computing*, 7(2):206–222, 2011.

[22] J. Yoon, M. Liu, and B. Noble. Random Waypoint Considered Harmful. In *INFOCOM*, 2003.