Towards an Extensive Map-oriented Trace Basis for Human Mobility Modeling

Matthias Schwamborn, Nils Aschenbruck
University of Osnabrück - Institute of Computer Science,
Wachsbleiche 27, 49090 Osnabrück, Germany
{schwamborn, aschenbruck}@uos.de

***Abstract*—Human mobility analysis and modeling is a very interdisciplinary field of research. Mobility models play an important role particularly in assessing the simulative performance of mobile and opportunistic networks. The mobility traces, which most of these models are based on, however, mostly suffer from several shortcomings. In this paper, we use the extensive Lausanne Data Collection Campaign (LDCC) mobility trace as basis for further map-oriented processing. Map-matching and sensible addition of optimal routes between points mitigate problems like GPS spatial noise, anonymization, and data gaps. Moreover, stay point extraction is performed as a preparation for the analysis of elemental statistical mobility characteristics. An exemplary impact evaluation of contact statistics shows that the resulting map-oriented trace basis is indeed suited for large-scale mobility analysis and simulation.**

Keywords*—*Trace Processing, Map-Matching, Stay Point Extraction, Mobility Analysis, Human Mobility, Opportunistic Networks

I. Introduction

For several years now, the statistical analysis of human mobility has been a considerably active research domain. Fields of application range from transportation system planning and travel demand forecasting (cf. [13], [17]) to biological and wireless virus spreading (cf. [5], [9], [14] and [19], [40], respectively). In the wireless and mobile networking research community, however, human mobility analysis is most commonly applied to and known in the context of mobility modeling for simulative performance evaluation of mobile and, particularly, opportunistic networks (for surveys, see [16], [27], [31], [38]). Mobility models characterize the nodes' mobility patterns in a network simulation and have long been found to have a significant impact on simulation results.

Opportunistic Networks (OppNets) [6], [27] are seen in challenged environments and are typically characterized by high-delay communication and rare end-to-end connectivity. Node mobility is not considered as a problem but rather as an opportunity to transport data with the *store-carry-forward* principle: Store data on the communication device, carry it to another position, and forward it to the next user as soon as the opportunity arises. This allows OppNets to bridge gaps in a partitioned network and forward the data from source to destination. Opportunistic communication is also regarded as an alternative solution (apart from Wi-Fi or femtocells) to *data offloading* [1], [10], where selected data is offloaded on to the OppNet, thus, saving valuable traffic resources of the Public Land Mobile Network. Evidently, human mobility is vital to data offloading using opportunistic communication and to the performance of OppNets in general [4].

Mobility models can be divided into *trace-based* and *synthetic* models [2]. Synthetic models, such as Random Waypoint (RWP) or Random Walk, mostly rely on arbitrary statistical assumptions about mobility but are easy to handle both in terms of trace generation for simulation as well as analytical evaluation. Trace-based models, on the other hand, rely on findings from statistical analysis of real trajectories (traces) measured by some device. These are usually more complex but also much more realistic. However, most trace-based mobility models make use of *flights*, where nodes move on a straight line to the target position. *Map-based* mobility models instead try to increase credibility by restricting the nodes' movements to the road network, thus, accounting for geographic constraints (see [34]).

Trace-based mobility models can only be as realistic as their trace basis and analysis thereof. Most mobility traces found in the literature suffer from scenario dependency, low granularity, a small user base, or short time span of the measurement campaign. If the traces were collected in the context of a certain scenario (e.g., campus), the validity of the analysis results might be limited to this scenario. Low granularity might suffice in the case of *macro- or meso-scopic* mobility, where the mobility of multiple users is aggregated and metrics such as cell-change rate are of interest. But in the case of *micro-scopic* mobility, which we focus on, mobility of single users is modeled and Global Positioning System (GPS)-level granularity is needed. A large user base and long measurement campaign results in a higher quality trace basis and is necessary to add credibility to the statistical analysis and derived model(s). In this paper, the trace basis is given by the Lausanne Data Collection Campaign (LDCC) data set [18], composed of GPS data (among others) collected from a total of 185 users for 19 months. This should, however, not be used in its raw form due to shortcomings like GPS measurement noise, anonymization of selected points, and intermittently missing positional data.

We have already shown in earlier work that integrating geographic restrictions in the form of digital map data into a human mobility model [34] and the road network's structure [35] show a significant impact on the performance evaluation of OppNets. These results collectively motivated us to go back in the trace-based modeling design chain and incorporate this map-oriented approach from the ground up, i.e., the trace analysis. Several incremental processing steps such as map-matching are involved and finally end in the extraction of so-called *stay points*: A stay point in the context of mobility modeling is a position where the user changes the direction and/or speed, while also pausing for a certain amount of time. Stay points are important for the analysis as they are the basis for the calculation of typical movement properties. This approach to trace processing yields traces which are—at least to some extend—rebuilt in the sense that they are closer to how the tracked user (likely) has moved in reality. Thus, it mitigates the above-mentioned shortcomings and adds more credibility to working with these traces. Furthermore, traces processed this way can be used more easily for *trace-driven* simulation where real traces are directly taken as input for the simulator. A big challenge, however, is the validation of the processed traces, especially if there is no ground-truth data to compare with. Since there is no ground-truth available in the case of the LDCC data set, we compare the data bases resulting from individual processing steps in the context of metrics commonly used for human mobility analysis.

The contribution of this paper is three-fold: (1) We detail the steps involved in processing the raw trace data in a map-oriented manner. (2) We evaluate the goodness of the resulting data bases after stay point extraction in spite of the lack of ground-truth data. (3) We inspect contact statistics for the best-fitting data basis as an indication for impact on OppNet performance.

II. Related Work

A. Trace-based Mobility Modeling

Numerous trace-based mobility models have emerged over the past years. A quite comprehensive, though by now somewhat outdated, survey was assembled by Aschenbruck *et al.* in [2]. They also survey several sources for publicly available mobility traces. A more recent survey by Hess *et al.* [12] focuses more on the engineering side of mobility modeling. They explain and give guidance on all parts of the modeling design chain, from mobility trace measurement to model validation. Treurniet [38] instead focuses on micro-scopic mobility models and presents a thorough taxonomy. Human mobility modeling in the context of OppNets is surveyed in [16], [27], [31].

B. Map-Matching

Research in the context of map-matching has yielded a plethora of algorithms [32]. However, since the most common application areas are transportation and navigation systems, these algorithms were engineered while keeping the on-line version of the map-matching problem in mind. Transportation and navigation systems require a real-time solution, and, therefore, the algorithm mostly has to manage on current and past input positions. Solving this on-line version of the problem when faced with post-processing of a trace is ineffective, though, as accuracy suffers from this limited knowledge. Thus, an off-line algorithm should be applied, which can incorporate knowledge about future positions to increase accuracy of the matching. These off-line algorithms can also better cope with challenges like low sampling rates or data sparseness.

Lou *et al.* [24] proposed an off-line map-matching algorithm for GPS traces with low sampling rates (sampling around every other minute). Their algorithm especially considers spatial geometric and topological structures of the road network. Furthermore, speed constraints are utilized to choose between different types of roads (e.g., motorways and residential roads). Pereira *et al.* [30] focused on improving incomplete map databases and proposed a hybrid genetic algorithm, which combines map-matching and identification of missing or erroneous road network data. An off-line algorithm robust to GPS spatial noise as well as temporal sparseness was proposed by Newson *et al.* [28]. Using a Hidden Markov Model (HMM) and the Viterbi algorithm, the most probable route among several candidates is computed for a given input trip. If an outlier in the input data would result in unlikely routes, this point is discarded and the computation is rerun on the remaining points. More details are given in Section IV-B.

C. Stay Point Extraction

Ashbrook *et al.* [3] proposed one of the first approaches to place extraction, where GPS signal loss was used as an indicator for a stay point. According to them, a stay point occurs when the user stays for a minimum of t seconds within a building, where no GPS signals can be received and, thus, no positions can be calculated. GPS noise is accounted for and stay points are aggregated to places by applying a variant of k-means clustering. More intelligent approaches choose a maximum roaming distance in addition to a minimum sojourn time as a parameter for stay point extraction [11], [15], [39]. Main differences can be found in the clustering algorithm, such as agglomerative clustering [11], on-line time-based clustering [15], or density-based clustering [39]. Montoliu *et al.* [25], [26] extend the approach by Ye *et al.* [39] by an upper time limit for stay points, such that data gaps are also accounted for. An improved grid-based clustering algorithm generalizes stay points to stay regions. Pavan *et al.* [29] make use of the accuracy and instant speed values provided by GPS to discard unreliable points and further improve the extraction. However, in our experience, these values are quite unreliable themselves.

III. Trace Basis

As mentioned in Section I, our trace basis is the GPS data of the LDCC data set [18], collected from a total of 185 users for 19 months between 2009/09 and 2011/03. This data set was also the main subject of research in the Nokia Mobile Data Challenge (MDC) [21], [22]. The collection campaign was performed within the area of Lausanne, Switzerland, and participants were recruited in a viral manner, eventually leading to a heterogeneous population of socially connected users from mixed backgrounds. Based on a state machine for optimizing power consumption of the mobile device, among other information, location data was logged for every user on a regular basis. Since sufficient GPS signal quality is not always available, GPS receiver data was complemented by GPS positions of known Wireless Local Area Network (WLAN) access points and cellular network information. However, we decided to ignore the cellular network information as these were not geo-localized and the estimated amount of work invested in data fusion outweighed the (for our purposes) potentially small overall gain in quality of the location data.

In order to protect the privacy of participants, they were able to access and free to delete their own data in part or even completely since the ownership of the data remained with the respective user. Additionally, the raw data was anonymized before given out to researchers [22]: Based on the principle of *k-anonymity* (cf. [37]), positional data, which would have led to the identification of users with given precision, was manually truncated in terms of (longitude, latitude) decimal places. The resulting anonymity rectangle was chosen such that it contains enough inhabitants, mitigating the risk of singling out a specific user. Obviously, anonymization of location data reduces the value for analysis purposes, however, it is a necessary step to protect the privacy of the users participating in the data collection campaign and is a general challenge with trace data.

IV. Trace Processing

In order to obtain a map-oriented trace basis, we apply a series of different processing steps. One of these steps is called *map-matching*, where raw positional data is basically matched with and snapped to the road network underlying a given digital map. In a further step, we extend this map-matched data by computing optimal routes to fill data gaps. Note that the map-oriented methodology proposed here can, in general, be applied to other GPS traces. However, the initial filtering step and, e.g., parameter values for further steps, need to be adapted to each trace basis individually.

The trace data, as described in the previous section, is basically a spatio-temporal sequence of quadruples $(user, time, latitude, longitude)$, composed of raw GPS and geo-located WLAN access point positions. After the removal of duplicate data points, a total of roughly 22 million samples for all users remains for further processing. In the following, we will detail the processing steps taken before the actual stay point extraction. As no ground-truth is available to evaluate the validity of the processing results, we will take the data bases resulting from each of these three steps as input for the stay point extraction algorithm and compare the output concerning commonly used metrics. In the first processing step, we filter user traces that are heavily anonymized or contain too few samples. Secondly, we apply a map-matching algorithm. In the last step, we fill data gaps with optimal routes.

A. Filtering

As an initial explorative inspection of the given trace data, we want to investigate how much of the location data is anonymized and to which degree. The degree of anonymization here is defined inversely proportional to the number of decimal places in the latitude/longitude decimal degree values (here called *scale* for short) since these have been truncated during the process of k-anonymization (cf. Section III). Figure 1 shows the scale distribution over all samples. The highest scale of 10 was the default of raw location data. Based on the distribution, a clear distinction can be made between anonymized and non-anonymized data: A scale up to 3 means that the corresponding location data has been anonymized. Samples with 3 or less decimal places amount to 41.8% of the data and correspond to a precision of around 100 m or less[1]. Decimal degree latitude/longitude values with 4 decimal places or more correspond to around 10 m precision or higher, which is sufficient for our purposes.

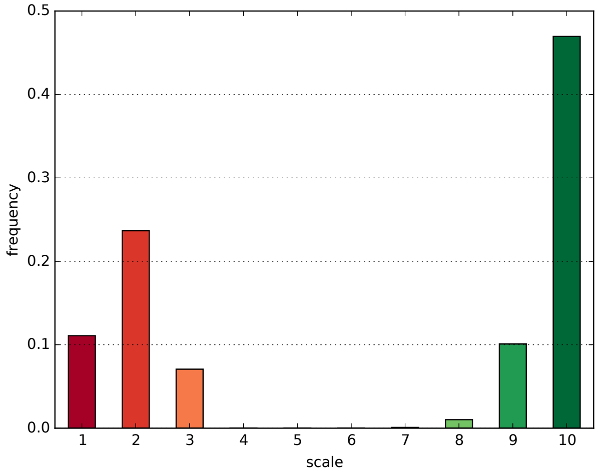


Figure 1. Normalized histogram of the number of decimal places (here called *scale*) over all samples. Lower scale means higher anonymization.

Defining a scale of 4 or more as “sufficient”, we can investigate the trace data quality per user (see Fig. 2). There are quite a few user traces which have been heavily anonymized. We decided to discard all user traces where the fraction of samples with insufficient precision is above 70% (cf. blue horizontal line in Fig. 2). Additionally, user traces with a total sample size of less than 10,000 were also discarded. After applying these filters to the trace data, 138 users with a total of 18,211,204 samples remain and we denote this as *data basis (a)*.

B. Map-Matching

The next step and first towards map-oriented processing is map-matching. Originally coming from the domain of vehicle navigation systems, map-matching solves the problem of matching measured location data to a given road network. As one of our original motivation aspects is based on the idea that humans nowadays mostly plan their routes with navigation systems optimizing distance or time taken instead of minimizing the flight length (cf. [34]), it should come as no surprise that other problems and solutions from that domain can be applied to human mobility modeling. Taking data basis (a) as input, we apply the map-matching algorithm by Newson and Krumm [28]. This algorithm is based on a HMM and is designed to be robust to GPS spatial noise and temporal sparseness in the input data. Since matching raw location data to the nearest road is too error-prone, the HMM lattice is build up by reading a sequence of points and an optimal path through this lattice is chosen with the Viterbi algorithm. States in the HMM represent road segments of the road network, state measurements represent the location measurements from the input trace data. Based on the connectivity of the road network and other factors, probabilities are assigned to state transitions, such that the most likely (reasonable) route is computed. Transition probabilities are set to zero in the following cases (cf. [28]):

- If a state represents a road segment that is too far away (200 m) from the measured point.
- If a route’s length is much greater than the great circle distance (greater by 2000 m).

[1] cf. https://en.wikipedia.org/wiki/Decimal_degrees

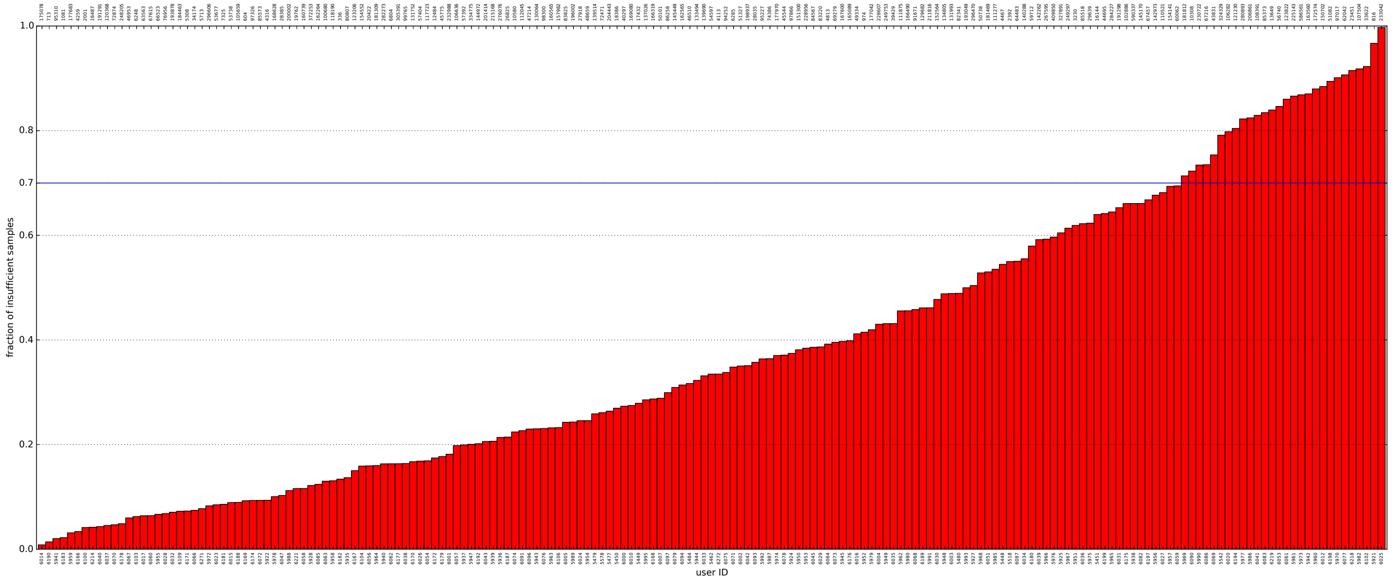


Figure 2. Distribution of anonymized location data per user. Samples with 3 or less decimal places are declared as “insufficient”. Vertical numbers on top denote the absolute number of samples for that user.

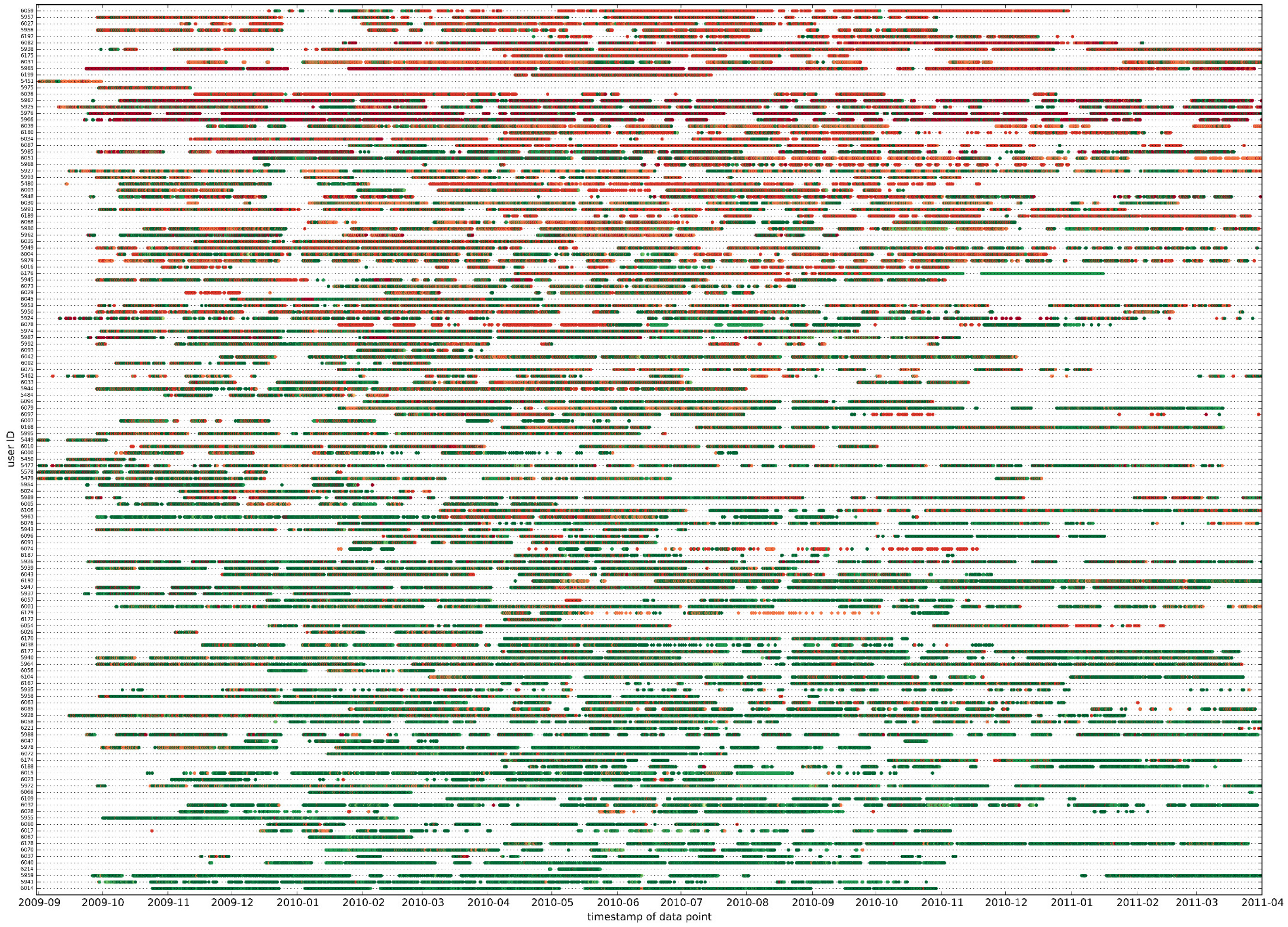


Figure 3. Availability scatter plot of GPS data over time after filtering heavily anonymized user traces. Users are sorted according to fraction of sufficiently accurate positions (color-coded with red-yellow-green).

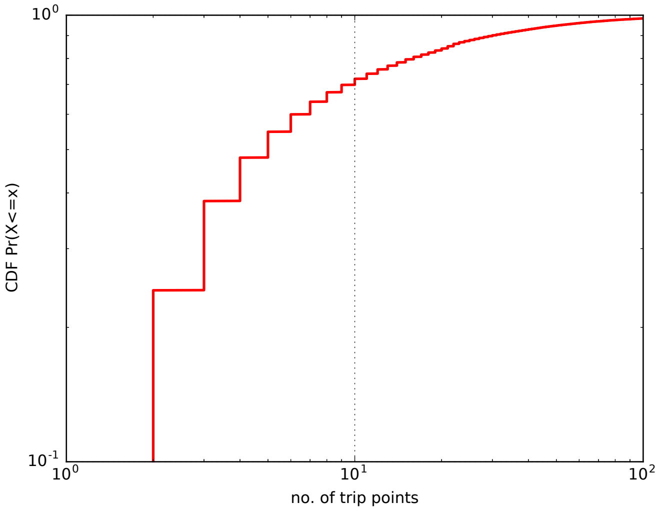


Figure 4. Distribution of trip sizes as input for map-matching.

- If a route would require an unreasonable speed (more than 180 km/h or three times the speed limit).

These conditions might lead to unmatchable points if there are no reasonable options among the candidates. The map-matching algorithm now tries to heal these breaks in the HMM by removing the problematic measured points and trying to reconnect the points before and after within the HMM. If a break could not be healed after incrementally trying to heal it this way and exceeding a time threshold of 180 s, the data is split into separate trips, and map-matching is applied to each one individually.

The map-matching algorithm has two tuning parameters: Gaussian GPS noise standard deviation σ_z and an indicator for the tolerance for non-direct routes β. Newson and Krumm estimate these parameters for their test data with the help of ground-truth data (see [28]). However, as ground-truth data is not available in our case, we had to manually adjust the parameters. Since there is no reliable metric for assessing the quality of the map-matching results without ground-truth, we had to resort to manually checking the results on a visual basis. For this purpose, we focused on the results for three representative user traces from different anonymization levels: Users 6014, 6000, and 6059 from the "low", "medium", and "high" anonymization levels (cf. Fig. 3, where user traces are sorted from low to high anonymization, i.e., low to high fraction of insufficiently accurate location data). The implementation of the map-matching algorithm we used is part of Open Source Routing Machine (OSRM)[2] (`node-osrm`) v4.6.1 and was running on Switzerland OpenStreetMap (OSM) data from 2015/04/24. Before actually running the map-matching, we dropped all remaining anonymized points since the algorithm cannot cope with too many of these in a meaningful way. Moreover, we divided user traces into trips, where a *trip* is a sequence of location data, in which each two consecutive points are at most 180 s apart, thereby abiding the above-mentioned time threshold defined by Newson and Krumm. Trips consisting of merely one point were also discarded. The resulting distribution of trip sizes (number of points in a trip) is shown in Fig. 4. The majority of trips contains only a few points which means the trace data is quite sparse, which can also be seen in Fig 3.

Using OSRM and OSM map data also requires defining which type of (OSM-)*ways* should be considered. After experimenting with the map-matching, it turned out that almost one third (30.6%) of all used (OSM-)ways were of the type "motorway" and 4.2% of the type "railway". This side result indicates that a speed model for human mobility should feature multi-modal transport in some fashion. Particularly note that our choice of (OSM-)way types basically supports multiple transport modes, i.e., pedestrian, bicycle, vehicle, train, etc.

The parameter value ranges tested were $\sigma_z \in \{4.07,\ 6,\ 8,\ 10\}$ and $\beta \in \{\frac{i}{2} : i \in \{1, \ldots, 10\}\}$. On average, the best visual results were achieved with $\sigma_z = 4.07$ (estimated value in [28]) and $\beta = 1.5$. An example trip is shown in Fig. 5. In order to at least get an idea of how well the input location data was suited for the map-matching, we can inspect the number of sub-matchings resulting from the map-matching of a trip, which is related to the number of HMM breaks (see above) that occured. The more sub-matchings, the more problematic points were removed by the map-matching algorithm. In Fig. 6, we can see that more than 50% of all map-matched trips were not sub-divided, i.e., resulted in exactly one sub-matching. Around 40% of all trips resulted in 2 to 6 sub-matchings, which is quite a lot considering that most of the trips contained only a few points. After applying map-matching to data basis (a), a total of 11,835,017 samples remain and we denote this as *data basis (b)*.

C. Route Filling

In addition to map-matching, it might also make sense to extend the location data by route segment points to fill some of the spatio-temporal data gaps. The rationale here is that we tend to plan our trips to the destination based on optimal routes. Thus, even if there is a gap in our data between two points, we can assume in certain cases that the tracked user traversed along an optimal route and fill the corresponding gap. However, if, for example, the temporal gap between these two consecutive points is too large, the uncertainty about the movement during this time is too high. Also, if the speed derived for the tentative route is too high, filling the gap in this manner makes no sense. Inspecting the time deltas (temporal gaps) between two consecutive points (cf. Fig. 7), we find that almost 90% of all time deltas are shorter than 120 s (cf. intersection of blue lines). We chose 300 s as an upper limit (corresponding to roughly 97%), which is still a realistic value for assuming the user did not make a significant detour during this time span. Furthermore, we set the upper limit for derived route speed to 120 km/h, which is the speed limit on motorways in Switzerland. This also matches well with the route speeds derived for shortest routes between consecutive points in data basis (b) as about 95% are below this limit (cf. Fig. 8). We opted for shortest instead of fastest routes since the considered time gaps are fairly small (cf. Fig. 7). This also enables us to perform a feasibility check on derived speeds: If the destination cannot be reached in the *predetermined* time on a shortest route, then even less on a fastest (usually longer) route and filling the corresponding data

[2] http://project-osrm.org/

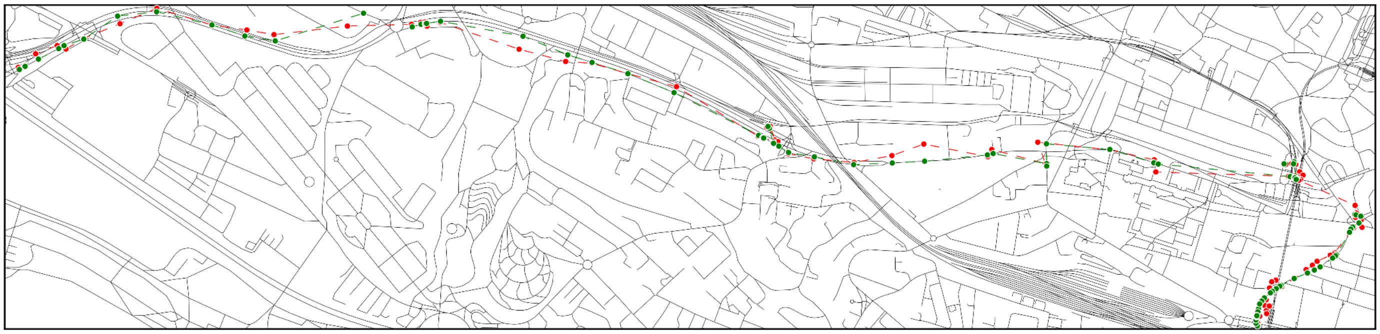


Figure 5. Map-matching results for an example trip. Red indicates input points, green indicates matched points.

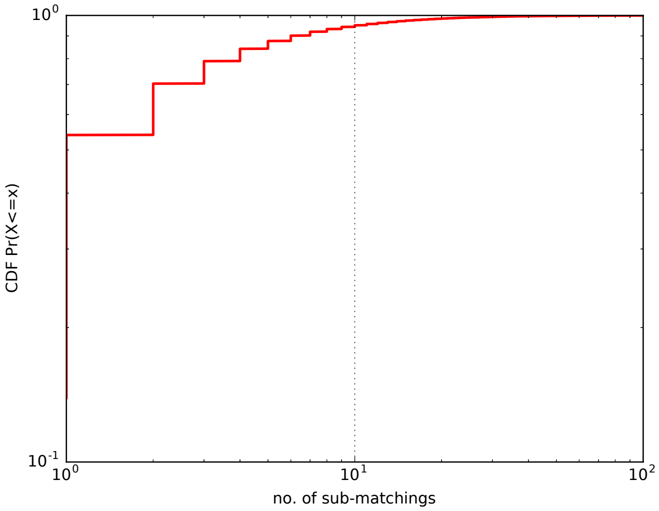


Figure 6. Distribution of number of sub-matchings for a trip after map-matching.

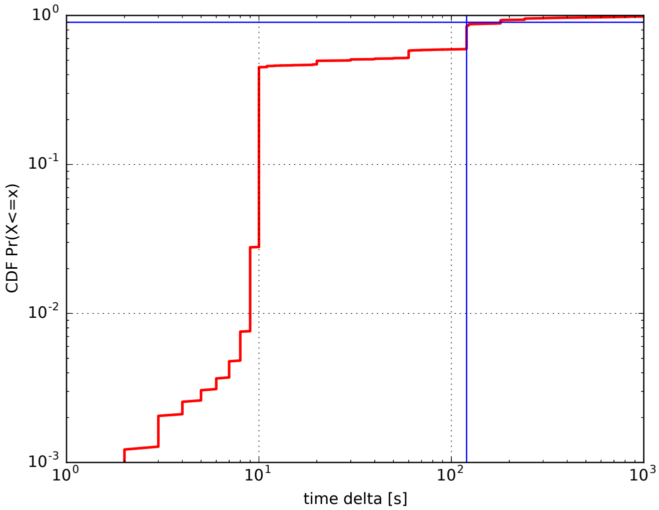


Figure 7. Distribution of time gaps between two consecutive points in data basis (b). Blue lines indicate 120 s and 0.9-quantile, respectively.

gap is likely infeasible. The computation of optimal routes was performed using OSRM (`osrm-backend`) v4.7.0, running on an OSM digital map for Switzerland (see above). Extending data basis (b) by route segment points results in a total of 43,247,962 samples and we denote this as *data basis (c)*.

V. Stay Point Extraction

After the map-oriented processing, we still have no (direct) information about the stay points, i.e., where did the users change direction and/or speed, while also staying for a period of time. This is due to the measurement method of GPS traces, where the tracking device usually logs the position every few seconds. However, stay points are important for human mobility analysis as they are the basis for the calculation of typical movement properties like pause times, flight lengths, speeds, etc. Stay points can be extracted (or *estimated*) from the traces with an extraction algorithm. Note that while a lot of work in this context is about finding stay regions or attraction points by basically generalizing or aggregating multiple stay points, this would be one abstraction level too much for our purposes.

The algorithm we use was proposed by Montoliu *et al.* [25] as an extension of the algorithm by Ye *et al.* [39] and has even been applied by Do *et al.* [7] to a subset of the trace basis we use here. According to this algorithm, three conditions must be fulfilled for a sequence of location points $(p_s,\ \dots,\ p_e)$ to be declared as a stay point:

1) $\texttt{geoDistance}(p_s,\ p_k) \leq D_{max},\ \forall k \in [s+1,\ e]$
2) $\texttt{timeDifference}(p_s,\ p_e) \geq T_{min}$
3) $\texttt{timeDifference}(p_k,\ p_{k+1}) \leq T_{max},\ \forall k \in [s,\ e-1]$

D_{max} is the maximum distance that may be covered by the user within the stay point's spatial bounds. T_{min} is the minimum pause time at the stay point and T_{max} limits the time two successive location points may be apart. A stay point for location points $(p_s,\ \dots,\ p_e)$ meeting these conditions is then defined as the corresponding centroid. Note that we replaced the strict inequalities in the above conditions with simple inequalities (cf. [25]). This might seem like a minor detail but can make a difference in output depending on the input trace data. We also explicitly use the wording *geo*-distance, denoting the computation of geodesic distances (more on geodesic vs. Euclidean distances in [35]). Furthermore, we fixed two problems with the original pseudocode in [25, Alg. 1]: On one hand, the original made a premature break (cf. [25, Alg. 1, line 9]) in the inner loop which results in discarding potential stay points (i.e., for $(p_i,\ \dots,\ p_{j-1})$). On

Algorithm 1: Stay point extraction (based on [25, Alg. 1]).

```
Input  : Temporally ordered list of $N$ location points $l_p = (p_0, \ldots, p_{N-1})$.
         Tuning parameters $T_{min}, T_{max}, D_{max}$.
Output: List of extracted stay points $l_{sp}$.
$i \leftarrow 0$;
$l_{sp} \leftarrow \emptyset$;
while $i < N - 1$ do
    $j \leftarrow i + 1$;
    while $j < N$ do
        $t_{succ} \leftarrow$ timeDifference($p_{j-1}, p_j$);
        $d_{total} \leftarrow$ geoDistance($p_i, p_j$);
        if $(t_{succ} > T_{max}) \vee (d_{total} > D_{max})$ then        // check upper time and distance bounds
            /* we know that $(p_i, \ldots, p_{j-1})$ meet the above conditions */
            $t_{total} \leftarrow$ timeDifference($p_i, p_{j-1}$);
            if $t_{total} \geq T_{min}$ then        // check lower time bound
                $sp \leftarrow$ createStaypoint($p_i, \ldots, p_{j-1}$);
                $l_{sp} \leftarrow l_{sp} \cup sp$;
            $i \leftarrow j$;
            break;
        if $j$ = $N$-$1$ then        // include last stay point for $(p_i, \ldots, p_{N-1})$ if applicable
            $t_{total} \leftarrow$ timeDifference($p_i, p_j$);
            if $t_{total} \geq T_{min}$ then
                $sp \leftarrow$ createStaypoint($p_i, \ldots, p_j$);
                $l_{sp} \leftarrow l_{sp} \cup sp$;
            $i \leftarrow j + 1$;
        $j \leftarrow j + 1$;
return $l_{sp}$;
```

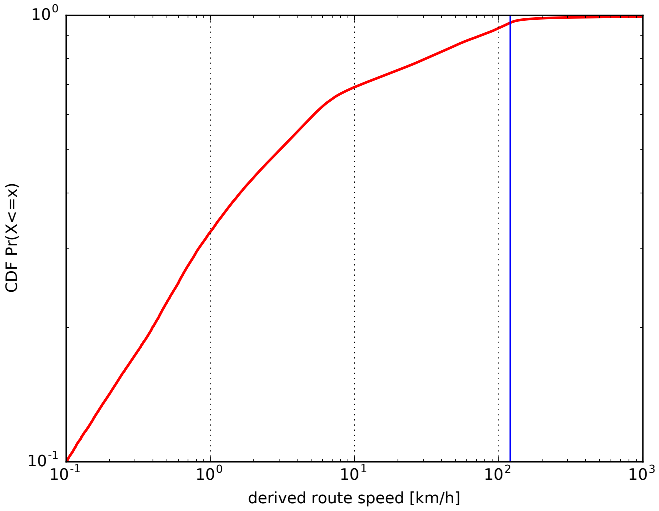


Figure 8. Distribution of derived route speeds. The speed limit of 120 km/h on motorways in Switzerland is marked by the blue line.

Table I. Sample sizes resulting from all three data bases.

Type	DB (a)	DB (b)	DB (c)
Location points	18,211,204	11,835,017	43,247,962
Stay points	42,126	24,885	31,764
Flights	9,375	7,343	8,598
SP IAT	31,990	16,462	21,611

the other hand, the inner loop variable runs one index too short which results in discarding a potential last stay point. The pseudocode fixing these problems can be found in Alg. 1.

As mentioned in Section IV, we took all three data bases (abbreviated as *DB* in the following) (a), (b), (c) as input for the stay point extraction. The evaluation of the stay point extraction results is commonly performed with the help of ground-truth data, allowing the definition and usage of metrics such as *precision*, *recall*, etc. which account for false positives and false negatives (cf. [25]). However, ground-truth was neither available nor collectible in our case. Therefore, we compared results for all three data bases in the context of human mobility metrics that are commonly found in the literature. Two prominent properties of human mobility are self-similarity of flights (cf., e.g., [33]) and periodicity of visited waypoints (cf., e.g., [8]). As indicators for these properties, we inspected the distribution of flights, i.e., step lengths between stay points, and the Inter-Arrival Time (IAT) between two visits of the same stay point. The tuning parameters for the stay point extraction were set as follows: $D_{max} = 200m$, $T_{min} = 5min$, and $T_{max} = 1min$. Note that values, especially for the time bounds, in the original paper were chosen significantly higher ($D_{max} = 250m$, $T_{min} = 30min$, and $T_{max} = 10min$ [25]), but their goal was different from ours since they were ultimately looking for stay *regions* instead of stay points. See Table I for the resulting number of extracted stay points. Most stay points were extracted from DB (a), but these also include stay points based on anonymized location points. Anonymized location points were dropped before the map-matching, leading to considerably less but more credible stay points extracted from DB (b). Due to the sensible addition of route segment points by the route filling processing step, a lot more location points are present in DB (c), also leading to more stay points.

Flights denote direct lines from one position to the next and are a common statistic to describe the distances between two successive stay points—although their usefulness in the context of human mobility is debatable [34]. As our data bases contain some data gaps, we extended the stay point extraction algorithm as described above to implement a stricter notion of successiveness: Two chronologically successive stay points sp_1 and sp_2 are *strictly successive* if $\texttt{timeDifference}(sp_1,\ sp_2) \leq 300s$. Note that this value was chosen to be equal to the upper limit for time deltas in Section IV-C for the same reason: We assume that the tracked user did not make a detour significantly deviating from the flight within this time. Computing flights for two chronologically successive stay points with a higher time difference might not be feasible anymore as the uncertainty about detours increases. Table I lists the number of (strictly successive) flights for all three stay point data bases.

The stay point IAT is defined as the time difference between two successive visits of the same stay point and is an indicator for periodicity. A post-processing step for the spatially fine-grained stay point extraction output was necessary to cluster stay points that might denote the same stay point in a semantic sense. Therefore, we transitively clustered all stay points within a range of 100 m, i.e., for each stay point in a cluster, there is at least one other stay point in the same cluster which is at most 100 m away (cf. waypoint clustering in [23]). The number of computed IAT samples can be gathered from Table I.

A comparison of the flight and stay point IAT distributions is shown in Fig. 9 (note the logarithmic axes). We chose the Complementary Cumulative Distribution Function (CCDF) as representation since we are more interested in higher values, i.e., the distribution's tail, for both metrics. As motivated in Section I, a map-oriented trace basis has several benefits, thus, we prefer data bases (b) and (c), where map-matching and the additional route filling, respectively, were applied. Comparing flights of up to 200 m, DB (b) is closer to DB (a) (cf. Fig. 9a). However, flights of this order are of lesser interest as mobility within these ranges is quite limited and can be considered as intra-stay-point movement due to the above-mentioned clustering range of 100 m. Comparing longer flights as well as stay point IATs (cf. Fig. 9b), DB (c) is closer to DB (a). DB (c) also has the additional benefit of a larger sample size since some data gaps were filled by sensibly adding optimal routes. We conclude that DB (c) is preferable over DB (b) and that there is no significant loss of statistical properties. Thus, overall, the proposed map-oriented approach to trace processing has not only sensibly rebuilt the original data, but also mostly preserved the statistical properties, which is mandatory for further mobility analysis.

VI. Impact Evaluation

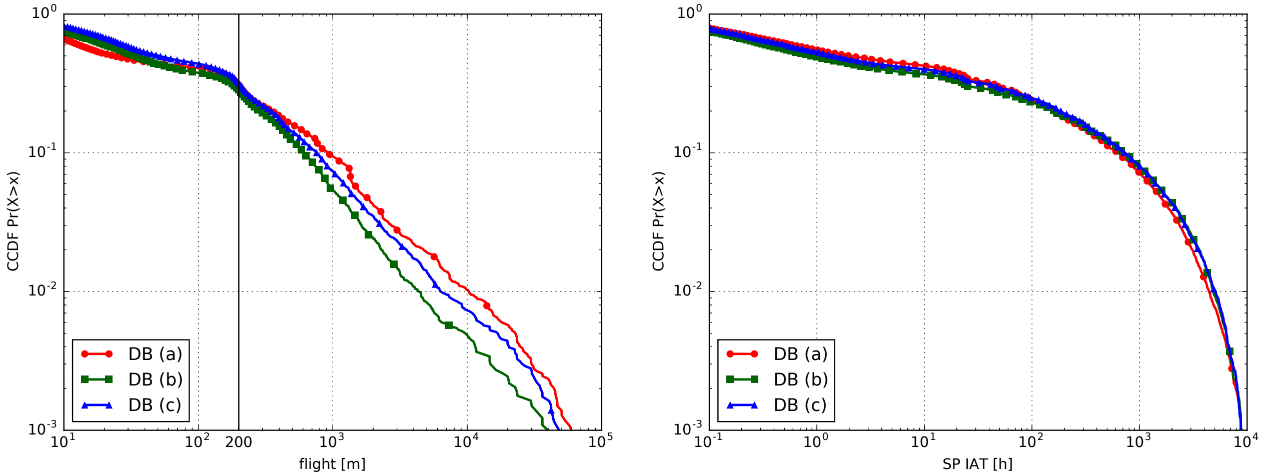
Having chosen DB (c) for further consideration (simply referred to as *LDCC* in the following), we can now exemplarily evaluate the impact on OppNet performance. For this purpose, contact statistics are most common since OppNets forwarding protocols heavily rely on contacts between users. In case of mobility traces, such as the LDCC trace, contact establishment and break-off must be defined by a signal propagation model. In order to minimize computation complexity, a simple unit disk propagation model is the most common choice. Typical contact-related metrics are the *Inter-Contact Time (ICT)* and *Contact Duration (CD)*. The ICT is the time between the break-off and re-establishment of two consecutive contacts between the same pair of users (cf. [4]). This metric is an estimate for the time it takes for two users to meet again and is directly related to communication delay. The CD is the time between establishment and break-off of a contact. It is used to estimate the amount of data (for a given data rate) which can be transmitted during a single contact.

Sensibly comparing contacts to other trace data is challenging, to say the least, since there is no data available which would be comparable to the LDCC trace. Nevertheless, we used the INFOCOM2006 contact trace [4], [36] as an example for contact statistics found in other available trace data. The INFOCOM2006 trace has also been used for model evaluation in, e.g., [20], [33]. These measurements were performed during the student workshop of the INFOCOM conference in 2006 from April 24th to 27th. 70 students and researchers participated and carried iMote devices, regularly scanning on the Bluetooth interface for other devices within a range of around 30 m.

CD and ICT distributions are shown in Fig. 10. In accordance with the INFOCOM2006 trace, we computed contacts for the LDCC trace with a *tx* range of 30 m. Overall, contacts last significantly longer in the LDCC trace (cf. Fig. 10a), which is, however, not necessarily to be expected: Attendants of such a workshop usually share a conference room where presentations are in progress for a certain amount of time. Still, around half of all contacts lasted only up to around 2 min, whereas half of all contacts in the LDCC trace lasted up to around 8 min. This difference increases with higher quantiles. ICTs in the LDCC trace are longer by multiple orders of magnitude (cf. Fig. 10b). This was to be expected due to both the much longer time span and much bigger area of the LDCC trace.

VII. Conclusion & Future Work

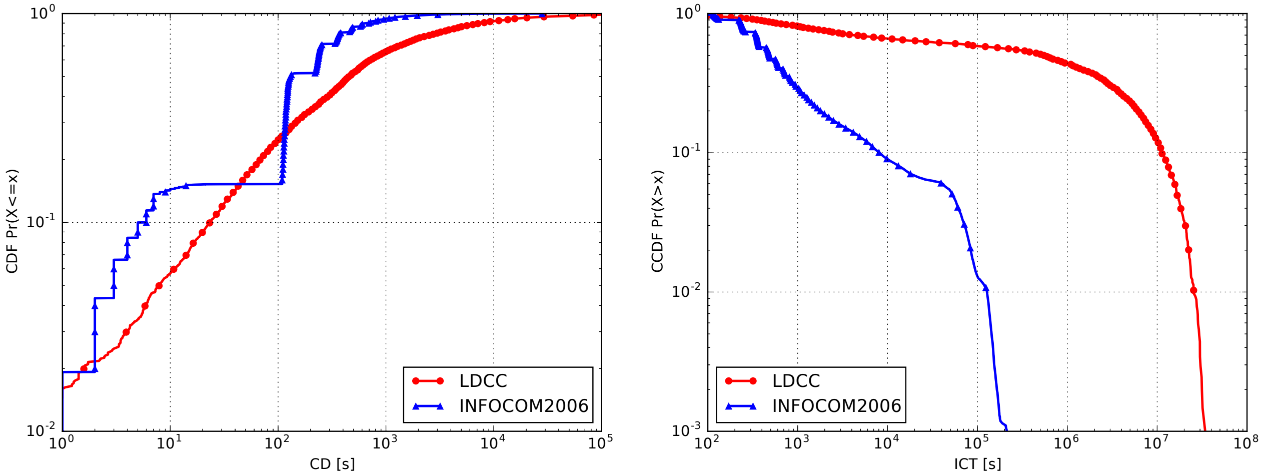
In this paper, we have shown how to perform several processing steps to achieve a map-oriented trace basis, ready for further human mobility analysis. Map-matching allowed to alleviate the problem of GPS spatial noise, inherent to any GPS-based measurements, and matched raw points to the underlying road network defined by a digital map. Temporal data gaps were further filled by sensibly adding optimal routes between two consecutive trace points. In a final step, we extracted stay points, which is necessary to analyze mobility characteristics such as pause times, flights, route lengths, etc. A comparison of stay point data resulting from the individual processing steps has shown that there is no significant loss of



(a) Flights

(b) SP IATs

Figure 9. Comparison of flights and SP IATs.



(a) Contact Duration

(b) Inter-Contact Time

Figure 10. Contact Duration and Inter-Contact Time distributions for *tx* range 30 m.

statistical properties, particularly concerning further analysis of self-similarity and periodicity. In a further impact evaluation, we inspected contact statistics as indicators for Opportunistic Network performance. Contacts mostly last from several minutes to a few hours, while Inter-Contact Times range from several minutes to a few months. Thus, the LDCC trace is suited for large-scale mobility analysis as well as simulation.

We conclude that even though we might have been able to cope with some of the shortcomings in the original data set, obviously, not all data gaps or anonymized points can be sensibly mitigated. Therefore, there is still some work to do in the area of trace collection in order to enhance the quality even before processing. However, we still believe that the LDCC trace is a step towards more extensive trace bases and that we were able to increase its quality by performing the proposed map-oriented processing. As future work, we plan to further utilize the map-oriented LDCC trace by analyzing statistical properties for human mobility modeling.

Acknowledgment

The authors would like to thank the anonymous reviewers for their invaluable comments and suggestions that helped improving the paper. The research in this paper used the MDC Database made available by Idiap Research Institute, Switzerland and owned by Nokia. Further trace data was also gratefully provided by the Community Resource for Archiving Wireless Data At Dartmouth (CRAWDAD). This work was supported in part by the German Research Foundation (Deutsche Forschungsgemeinschaft, DFG), Project No. AS341/3-1 as well as the “Stifterverband für die Deutsche Wissenschaft” (H170 5701 5020 20951).

References

[1] A. Aijaz, H. Aghvami, and M. Amani, “A Survey on Mobile Data Offloading: Technical and Business Perspectives,” *IEEE Wireless Communications*, vol. 20, no. 2, pp. 104–112, 2013.

[2] N. Aschenbruck, A. Munjal, and T. Camp, “Trace-based Mobility Modeling for Multi-hop Wireless Networks,” *Elsevier Computer Communications*, vol. 34, no. 6, pp. 704–714, 2011.

[3] D. Ashbrook and T. Starner, “Using GPS to Learn Significant Locations and Predict Movement Across Multiple Users,” *Personal and Ubiquitous Computing*, vol. 7, no. 5, pp. 275–286, 2003.

[4] A. Chaintreau, P. Hui, J. Crowcroft, C. Diot, R. Gass, and J. Scott, “Impact of Human Mobility on Opportunistic Forwarding Algorithms,” *IEEE Transactions on Mobile Computing*, vol. 6, no. 6, pp. 606–620, 2007.

[5] V. Colizza, A. Barrat, M. Barthelemy, A.-J. Valleron, and A. Vespignani, “Modeling the Worldwide Spread of Pandemic Influenza: Baseline Case and Containment Interventions,” *PLoS Medicine*, vol. 4, no. 1, pp. 95–110, 2007.

[6] M. Conti and S. Giordano, “Mobile Ad Hoc Networking: Milestones, Challenges, and New Research Directions,” *IEEE Communications Magazine*, vol. 52, no. 1, pp. 85–96, January 2014.

[7] T. Do and D. Gatica-Perez, “The Places of Our Lives: Visiting Patterns and Automatic Labeling from Longitudinal Smartphone Data,” *IEEE Transactions on Mobile Computing*, vol. 13, no. 3, pp. 638–648, 2014.

[8] F. Ekman, A. Keränen, J. Karvo, and J. Ott, “Working day movement model,” in *Proc. of the 1st Workshop on Mobility Models (MobilityModels ’08)*, Hong Kong, China, 2008, pp. 33–40.

[9] S. Eubank, H. Guclu, V. S. Anil Kumar, M. V. Marathe, A. Srinivasan, Z. Toroczkai, and N. Wang, “Modelling Disease Outbreaks in Realistic Urban Social Networks,” *Nature*, vol. 429, no. 6988, pp. 180–184, 2004.

[10] B. Han, P. Hui, V. Kumar, M. Marathe, J. Shao, and A. Srinivasan, “Mobile Data Offloading through Opportunistic Communications and Social Participation,” *IEEE Transactions on Mobile Computing*, vol. 11, no. 5, pp. 821–834, 2012.

[11] R. Hariharan and K. Toyama, “Project Lachesis: Parsing and Modeling Location Histories,” in *Proc. of the 3rd Int. Conference on Geographic Information Science (GIScience ’04)*, Adelphi, MD, USA, 2004, pp. 106–124.

[12] A. Hess, K. Hummel, W. Gansterer, and G. Haring, “Data-driven Human Mobility Modeling: A Survey and Engineering Guidance for Mobile Networking,” *ACM Comput. Surv.*, vol. 48, no. 3, pp. 38:1–38:39, 2015.

[13] M. Horner and M. O’Kelly, “Embedding Economies of Scale Concepts for Hub Network Design,” *Journal of Transport Geography*, vol. 9, no. 4, pp. 255–265, 2001.

[14] L. Hufnagel, D. Brockmann, and T. Geisel, “Forecast and Control of Epidemics in a Globalized World,” *Proc. of the National Academy of Sciences of the United States of America (PNAS)*, vol. 101, no. 42, pp. 15 124–15 129, 2004.

[15] J. Kang, W. Welbourne, B. Stewart, and G. Borriello, “Extracting Places from Traces of Locations,” *SIGMOBILE Mob. Comput. Commun. Rev.*, vol. 9, no. 3, pp. 58–68, 2005.

[16] D. Karamshuk, C. Boldrini, M. Conti, and A. Passarella, “Human Mobility Models for Opportunistic Networks,” *IEEE Communications Magazine*, vol. 49, no. 12, pp. 157–165, 2011.

[17] R. Kitamura, C. Chen, R. Pendyala, and R. Narayanan, “Microsimulation of Daily Activity-travel Patterns for Travel Demand Forecasting,” *Transportation*, vol. 27, no. 1, pp. 25–51, 2000.

[18] N. Kiukkonen, J. Blom, O. Dousse, D. Gatica-Perez, and J. Laurila, “Towards Rich Mobile Phone Datasets: Lausanne Data Collection Campaign,” in *Proc. of the 7th Int. Conference on Pervasive Services (ICPS ’10)*, Berlin, Germany, 2010, p. 7 pages.

[19] J. Kleinberg, “Computing: The Wireless Epidemic,” *Nature*, vol. 449, no. 7160, pp. 287–288, 2007.

[20] S. Kosta, A. Mei, and J. Stefa, “Large-Scale Synthetic Social Mobile Networks with SWIM,” *IEEE Transactions on Mobile Computing*, vol. 13, no. 1, pp. 116–129, Jan 2014.

[21] J. Laurila, D. Gatica-Perez, I. Aad, J. Blom, O. Bornet, T. Do, O. Dousse, J. Eberle, and M. Miettinen, “The Mobile Data Challenge: Big Data for Mobile Computing Research,” in *Proc. of the Mobile Data Challenge by Nokia Workshop*, Newcastle, UK, June 2012.

[22] ——, “From Big Smartphone Data to Worldwide Research: The Mobile Data Challenge,” *Pervasive and Mobile Computing*, vol. 9, no. 6, pp. 752–771, 2013.

[23] K. Lee, S. Hong, S. J. Kim, I. Rhee, and S. Chong, “SLAW: Self-Similar Least-Action Human Walk,” *IEEE/ACM Transactions on Networking*, vol. 20, no. 2, pp. 515–529, 2012.

[24] Y. Lou, C. Zhang, Y. Zheng, X. Xie, W. Wang, and Y. Huang, “Map-Matching for Low-Sampling-Rate GPS Trajectories,” in *Proc. of the 17th ACM SIGSPATIAL Int. Conference on Advances in Geographic Information Systems (GIS ’09)*, Seattle, WA, USA, 2009, pp. 352–361.

[25] R. Montoliu, J. Blom, and D. Gatica-Perez, “Discovering Places of Interest in Everyday Life from Smartphone Data,” *Multimedia Tools and Applications*, vol. 62, no. 1, pp. 179–207, 2013.

[26] R. Montoliu and D. Gatica-Perez, “Discovering Human Places of Interest from Multimodal Mobile Phone Data,” in *Proc. of the 9th Int. Conference on Mobile and Ubiquitous Multimedia (MUM ’10)*, Limassol, Cyprus, 2010, pp. 12:1–12:10.

[27] V. Mota, F. Cunha, D. Macedo, J. Nogueira, and A. Loureiro, “Protocols, Mobility Models and Tools in Opportunistic Networks: A Survey,” *Elsevier Computer Communications*, vol. 48, pp. 5–19, 2014.

[28] P. Newson and J. Krumm, “Hidden Markov Map Matching Through Noise and Sparseness,” in *Proc. of the 17th ACM SIGSPATIAL Int. Conference on Advances in Geographic Information Systems (GIS ’09)*, Seattle, WA, USA, 2009, pp. 336–343.

[29] M. Pavan, S. Mizzaro, I. Scagnetto, and A. Beggiato, “Finding Important Locations: A Feature-Based Approach,” in *Proc. of the 16th Int. Conference on Mobile Data Management (MDM ’15)*, Pittsburgh, PA, USA, 2015, pp. 110–115.

[30] F. Pereira, H. Costa, and N. Pereira, “An Off-line Map-Matching Algorithm for Incomplete Map Databases,” *European Transport Research Review*, vol. 1, no. 3, pp. 107–124, 2009.

[31] P. Pirozmand, G. Wu, B. Jedari, and F. Xia, “Human Mobility in Opportunistic Networks: Characteristics, Models and Prediction Methods,” *Elsevier Journal of Network and Computer Applications*, vol. 42, no. 0, pp. 45–58, 2014.

[32] M. Quddus, W. Ochieng, and R. Noland, “Current Map-Matching Algorithms for Transport Applications: State-of-the-Art and Future Research Directions,” *Transportation Research Part C: Emerging Technologies*, vol. 15, no. 5, pp. 312–328, 2007.

[33] I. Rhee, M. Shin, S. Hong, K. Lee, S. J. Kim, and S. Chong, “On the Levy-Walk Nature of Human Mobility,” *IEEE/ACM Transactions on Networking*, vol. 19, no. 3, pp. 630–643, 2011.

[34] M. Schwamborn and N. Aschenbruck, “Introducing Geographic Restrictions to the SLAW Human Mobility Model,” in *Proc. of the 21st Int. Symposium on Modeling, Analysis and Simulation of Computer and Telecommunication Systems (MASCOTS ’13)*, San Francisco, CA, USA, 2013, pp. 264–272.

[35] ——, “On Modeling and Impact of Geographic Restrictions for Human Mobility in Opportunistic Networks,” in *Proc. of the 23rd Int. Symposium on Modeling, Analysis and Simulation of Computer and Telecommunication Systems (MASCOTS ’15)*, Atlanta, GA, USA, 2015, pp. 178–187.

[36] J. Scott, R. Gass, J. Crowcroft, P. Hui, C. Diot, and A. Chaintreau, “CRAWDAD dataset cambridge/haggle (v. 2009-05-29),” May 2009. [Online]. Available: http://crawdad.org/cambridge/haggle/20090529

[37] L. Sweeney, “k-anonymity: A Model for Protecting Privacy,” *Int. Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, vol. 10, no. 5, pp. 557–570, 2002.

[38] J. Treurniet, “A Taxonomy and Survey of Microscopic Mobility Models from the Mobile Networking Domain,” *ACM Computing Surveys*, vol. 47, no. 1, pp. 14:1–14:32, 2014.

[39] Y. Ye, Y. Zheng, Y. Chen, J. Feng, and X. Xie, “Mining Individual Life Pattern Based on Location History,” in *Proc. of the 10th Int. Conference on Mobile Data Management (MDM ’09)*, Taipei, Taiwan, 2009, pp. 1–10.

[40] E. Yoneki, P. Hui, and J. Crowcroft, “Wireless Epidemic Spread in Dynamic Human Networks,” in *Proc. of the 1st Workshop on Bio-Inspired Design of Networks (BIOWIRE ’07)*, Cambridge, UK, 2007, pp. 116–132.