Pervasive and Mobile Computing 42 (2017) 27–44

Contents lists available at ScienceDirect

Pervasive and Mobile Computing

journal homepage: www.elsevier.com/locate/pmc

Effective opportunistic dissemination of spatio-temporal contents in mobile environments

Daisuke Kasamatsu, Peizhao Hu, Mohan Kumar *

Department of Computer Science, Rochester Institute of Technology, NY, USA

ARTICLE INFO

Article history:
Received 22 July 2016
Received in revised form 7 August 2017
Accepted 11 August 2017
Available online 18 August 2017

Keywords:
Opportunistic environments
Content dissemination
Analytical framework
Spatio-temporal reachability graphs

ABSTRACT

Dissemination of spatio-temporally valid content from content providers to consumers is critical in certain application contexts as data items could lose their validity across time and space. Content sharing in challenged opportunistic environments remains a research challenge as existing solutions fail to exploit dissemination patterns across spatio-temporal limits. In this paper, we propose spatio-temporal reachability graphs to depict reachability of time- and space-sensitive content in opportunistic mobile environments. Furthermore, we develop an analytical framework to estimate content distribution in such environments and validate its feasibility over long-term datasets. We perform extensive trace-driven simulation studies to determine content dissemination properties of environments with known mobility patterns. The analytical framework estimates dissemination ratio, optimizes parameter setting, and tests transmission capacities of opportunistic environments. Proposed scheme is useful to content providers as well as receivers.

© 2017 Published by Elsevier B.V.

1. Introduction

The proliferation of mobile and wearable devices has created opportunities for disseminating information/contents via peer-to-peer communication when devices encounter each other. This type of data transmission relying on a series of spatiotemporally distributed contacts is called opportunistic networking [1]. Opportunistic communication is a low-cost, high-bitrate alternative method for disseminating content on a massive scale; for example, crowd-sensing of air quality data (measuring multiple polluting gases) every minute by sensors mounted on nearly 13 000 taxis in the city of New York. Aggregation of data from large numbers of sensors can lead to air quality assessment in a given area. In such crowd sensing applications, a single measurement can be lost or delayed without introducing deviation or distorting the outcome. Hence, it is preferred to gather this type of data cost-effectively using opportunistic communication instead of the expensive cellular networks. Data is typically shared among each other by opportunistic forwarding and uploaded when WiFi is available. Furthermore, the rapid growth of Internet-of-Things (IoT) has created an opportunity for extracting more variety of information about a place, an object, and a person. But, data produced by IoT devices (e.g., augmenting taxis with air quality sensors) will significantly increase demand for bandwidth which makes existing infrastructure networks hard to cope. Data may also have validity limits across time and space domains. A more effective design of opportunistic content sharing is needed to address the ever-increasing demand for bandwidth.

We introduce the notions of *time-to-live* (TTL) and *space-to-live* (STL) to opportunistic data/content dissemination in mobile environments so that data is forwarded only within a predefined time and space boundary. When considering TTL and STL in content dissemination, it is important to determine the appropriate values to maximize reachability and minimize

* Corresponding author.
E-mail address: mjk@cs.rit.edu (M. Kumar).

http://dx.doi.org/10.1016/j.pmcj.2017.08.001
1574-1192/© 2017 Published by Elsevier B.V.

overheads due to data caching and transmission. However, it is often difficult to identify these boundaries appropriately if the mobility pattern of users in a given area is unknown. Prior work investigates opportunistic forwarding in simulation using synthesized mobility models (e.g., random walk) [2,3]. For example, Han and Srinivasan [3] studied influential mobile users in a social network context for content distribution, and evaluated their models through simulation using random walk. The results of this type of research are meaningful in general cases when we assume user mobility. Hence their opportunistic contacts conform to a normal distribution. While the random walk mobility model has been used in many opportunistic networking studies [2,4–6], the results often represent system performance in most general cases. However, despite the diversity in individual mobilities, humans tend to follow simple mobility patterns with a high degree of temporal and spatial regularity [7,8]. By capturing and modeling the regularity of human mobility, we can leverage this insight to significantly improve data forwarding algorithms and predict data dissemination in more realistic opportunistic networking settings. In the context of opportunistic content sharing, the state-of-the-art has offered some insights into opportunistic content dissemination [9], spatiotemporal information dissemination [10], and temporal reachability [11]. However, none of the existing approaches quantify the influence of changes of spatiotemporal property on reachability in different application scenarios and provide a way to adapt parameter settings to the different application contexts. Investigations on the impact of spatiotemporal property, file size, bandwidth, and other parameters on dissemination ratio and dissemination capacity are needed. Without a priori knowledge about dissemination capability of opportunistic environments, there would be no motivation for producers and service providers of opportunistic content sharing.

In this paper, we develop spatiotemporal reachability graphs (STRG) to model spatiotemporal dissemination in mobile environments where content is distributed using opportunistic forwarding. STRG captures available source–destination node pairs that satisfy end-to-end delay and geographic distance requirements. Based on STRG, we develop an analytical framework to make following types of estimations in a given scenario: expected success in content dissemination and expected transmission capacity. In particular, the framework comprises three functions: (i) estimate dissemination ratio, (ii) test transmission capacities, and (iii) optimize parameter setting. Extensive trace-driven simulation studies are performed on two large taxicab traces collected in urban areas for a period of one month. Distinguishing weekdays/weekends and time windows, our analysis reveals spatiotemporal patterns of contact opportunities. There is a high correlation between trajectories and contacts. Leveraging the patterns, the framework aims to predict content distribution parameters based on only historical mobility traces. To demonstrate the robustness of our framework, we also test the predictability by adding different degrees of variations in mobility traces. Our simulation results show the effectiveness, that is, the framework achieves more than 75% accuracy and reduces prediction error by nearly 50%, compared to a random method. Also, the analytical framework addresses above questions for an any opportunistic environment.

The paper makes three main contributions:

- *Spatiotemporal Reachability Graphs (STRG)*. A graph model has been developed to capture spatiotemporal dissemination in opportunistic environments;
- *Analytical framework for content dissemination*. Analytic functions are developed based on STRG to estimate content distribution parameters; and
- *Extensive trace-based simulations*. Applicability of the framework is demonstrated with long-term empirical datasets collected in urban areas.

The novelty of this paper lies in a generic analytical framework that can provide guidance to application developers, service providers and users in opportunistic environments. With the use of our methodology, it is possible to achieve acceptable levels of prediction of opportunistic content dissemination in such uncertain environments. The framework would be useful for opportunistic content sharing in any mobility scenario and also enable effective dissemination with fine-tuned parameter settings.

The remainder of this paper is structured as follows. First, Section 2 presents the state-of-the-art approaches to opportunistic content sharing. Section 3 discusses network models, including the introduction of the spatiotemporal reachability graph which is an extension of the time-varying graph. Section 4 introduces an analytical framework which discusses how user mobility data can be collected and fed into systems that compute the relevant estimation on spatiotemporal property and reachability in the environment. Section 5 elaborates the predicting scheme for content dissemination. Section 6 presents results trace-based simulations and discusses the effectiveness of the proposed framework. The final section concludes the paper.

2. Related work

This section presents the state-of-the-art regarding content sharing and dissemination analysis in opportunistic environments.

For opportunistic content sharing, Hyytiä et al. [9] have proposed the concept of floating content to improve performance of geo-caching within a specific region, where nodes store and distribute local spatiotemporal floating information. Information dissemination is geographically limited within the area of interest, whereas nodes are free to delete the information outside the area. They determined the length of time information remains available in the area of interest in such synthetic mobility models as Manhattan road network model and random waypoint model. The authors showed that

the expected lifetime of the information depends on two properties of contact frequency and node arrival rate in the area of interest via a numerical simulation study. They do not consider temporal and spatial contexts simultaneously.

Wang et al. [10] investigated the inherent properties of tempo-spatial information dissemination in order to understand space and time limitations. In particular, they tried to answer two questions: (i) given a region, what is the distribution of the minimum time needed to spread information over the region? and (ii) what is the minimum time needed for the information to arrive in a region from a remote source? In the adoption of the Lévy Walk mobility model, they derived theorems on the time bounds to spread information to a given region. They validated the theoretical model via a numerical simulation study. This paper is perhaps the most closely related work to our study because the model considers TTL and STL requirements. However, the model does not estimate content distribution parameters as we do in our algorithm. Lévy Walk mobility model is far from real-world mobility traces. Typically, real-world traces possess sparse/dense regions of trajectories in a reference area. There is also a significant difference in population between daytime/midnight and weekdays/weekends. Since such heterogeneous patterns are neglected in the mobility model, there remain unrealistic assumptions on the predictability of node movements.

As an analysis of temporal property, Tang et al. [12] have proposed temporal distance metrics to capture the temporal characteristics of time-varying graphs, e.g., how fast information spreads to all nodes. Comparing the static and temporal connected components, the trace-based simulation showed that the static model overestimates the connectedness of the network, since the model ignores time order of contacts. Whitbeck et al. [11] have proposed the concept of temporal reachability graphs (TRGs) to characterize multi-hop connectivity. Given edge traversal time and the maximum journey delay, a TRG captures such temporal structures of time-varying graphs as temporal connectivity, temporal asymmetry, and temporal dominating set. Applying the concept to synthetic and real-world mobility traces, they showed bounds on communication capabilities, i.e., delivery ratio, which is very sensitive to increasing transmission time between two nodes. Pietiläinen et al. [13] have proposed a methodology to break down traces of opportunistic social networks into temporal communities, i.e., clusters of people who meet periodically during an experiment. The trace-driven analysis demonstrated that efficient content dissemination is made by the mobile nodes with high contact rate and movement among temporal communities. Kim et al. [14] have proposed an analytical framework based on continuous-time Markov chain to provide probabilistic guarantees on the time of information spread. They demonstrated efficacy of the framework by applying it to vehicular mobility traces obtained from more than a thousand taxis in Shanghai. Zhu et al. [15] have proposed temporal capacity graphs to characterize the maximum amount of data that can be transmitted between any two nodes within a specific timeframe. Applying the graph model to synthetic and real-world mobility traces, they showed the fundamental relationships and tradeoffs between network setting and system performance, e.g., transmission delay vs. amount of transmitted data and data size vs. delivery ratio. However, the focus of the above work is on the temporal property rather than on the spatial property.

On deploying mobile applications of opportunistic content sharing, there are several important issues: social centrality of users [3,16], user preferences [17,18], privacy concerns [19], incentive mechanism [20], content popularity [21], traffic offloading [22], and real-world implementation [23]. However, the above work is not closely related to this paper.

3. Spatiotemporal reachability graphs

This section defines a network model and details spatiotemporal reachability graphs. After showing an example of spatiotemporal dissemination, we define the graph model and then develop an algorithm to build the graphs.

3.1. Network model

Opportunistic environments can be modeled based on the well-known time-varying graphs [24,25]. The graph is suitable for networks, where mobile users move around freely and contacts between users appear or disappear over time.

Figs. 1a–1d illustrate instantaneous graph snapshots at four time slices, $t = 1, \ldots, 4$. Nodes correspond to mobile users, whereas edges represent contacts between users. Node 3 has a contact with Node 4 at $t = 1$, and then Node 4 has a contact with Node 1 at $t = 3$ since they satisfy the spatiotemporal constraints. If the contact between Nodes 3 and 4 remains available until the data is transferred, the transmission starts at $t = 1$ and completes at $t = 1 + t_t$, where t_t is the transmission time. Then, if the contact between Nodes 4 and 1 at $t = 3$ remains available during the data transmission, the communication from Node 3 to Node 1 via Node 4 is available over time.

Time-Varying Graphs (TVG). A graph snapshot is described as $G_t = (V_t, E_t, \zeta)$, where t is time instance. A set of nodes at time t is denoted by V_t, and $E_t \subseteq V_t \times V_t$ is the set of edges between nodes at time t. A latency function, denoted by ζ, indicates edge traversal time (i.e., transmission time) it takes to cross a given edge at a given time. $G = \{G_1, G_2, \ldots, G_n\}$ is a set of all graph snapshots for a lifetime of a system.

Journey. A journey from a source node s to a destination node d, denoted by $J_{s,d}$, is described as a sequence of tuples $\langle e_{u,v}, t\rangle$, each of which represents a contact between nodes u and v at time t. In the above example, the communication from Node 3 to Node 1 is $J_{3,1} = \{\langle e_{3,4}, 1\rangle, \langle e_{4,1}, 3\rangle\}$.

Temporal length of journey. Temporal length is an end-to-end duration, given by $J_{s,d}(t) = t_d - t_s$, where t_s is the departure time at source and t_d is the arrival time at destination. In the above example, $J_{3,1}(t) = 3$, where $t_d = 4$ and $t_s = 1$. When there are multiple journeys from a source node to a destination node at time t_s, the *earliest* journey is given by the earliest arrival time at the destination node.

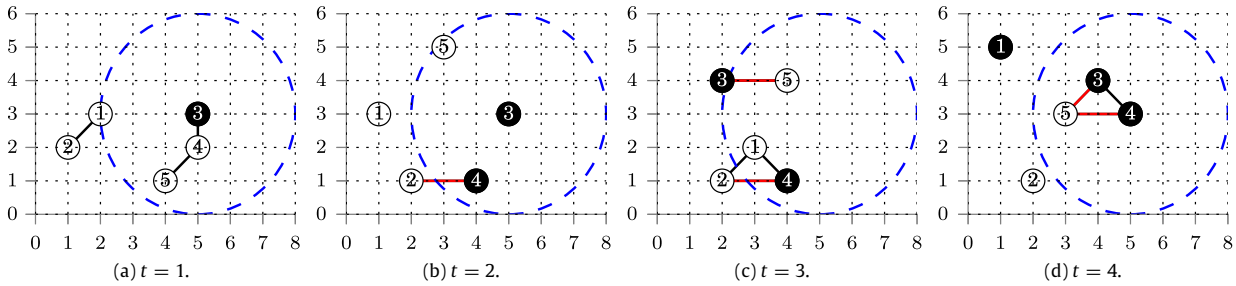


Fig. 1. An example of time-varying graphs, where (x, y)-coordinates are in units, the edges represent pair-wise contacts, the shaded nodes represent nodes that have data originated from Node 3, and the dotted circle represents the spatial-validity area that Node 3 declares when it first forwards data to Node 4 at $t = 1$.

Spatial length of journey. Let source position be the position where a source node initiates dissemination. The farthest point on the *earliest* journey is given by the maximum distance from a source position to each node on the *earliest* journey. Spatial length is the farthest distance from a source to the farthest destination. In the above example, the spatial length of $J_{3,1} = \{\langle e_{3,4}, 1\rangle, \langle e_{4,1}, 3\rangle\}$ is obtained as follows. A source position is at coordinates $(5, 3)$. The distance between two nodes u and v is given by $\sqrt{(u_x - v_x)^2 + (u_y - v_y)^2}$. For the first contact $e_{3,4}$ at $t = 1$, the distance from the source position to Node 3 at $(5, 3)$ is zero unit, whereas the distance to Node 4 at $(5, 2)$ is one unit. For the second contact $e_{4,1}$ at $t = 3$, the distance from the source position to Node 4 at $(4, 1)$ is 2.236 units, whereas the distance to Node 1 at $(3, 2)$ is 2.236 units. Thus, the farthest point is Node 4 or Node 1 at $t = 3$, and then the spatial length of $J_{3,1}$ is 2.236 units.

3.2. Spatiotemporal dissemination

In spatiotemporal dissemination, data is distributed within space and time limits.

Time-to-live (TTL). Let δ be the TTL, that specifies a temporal-validity of data by limiting the maximum delay from departure time t_s.

Space-to-live (STL). Let γ be the STL, that stipulates a spatial-validity of data with the radius γ from a source position by restricting the maximum distance from the source position. If needed, other area specifications (e.g., line, rectangle, and polygon) are also available with minor change of this definition.

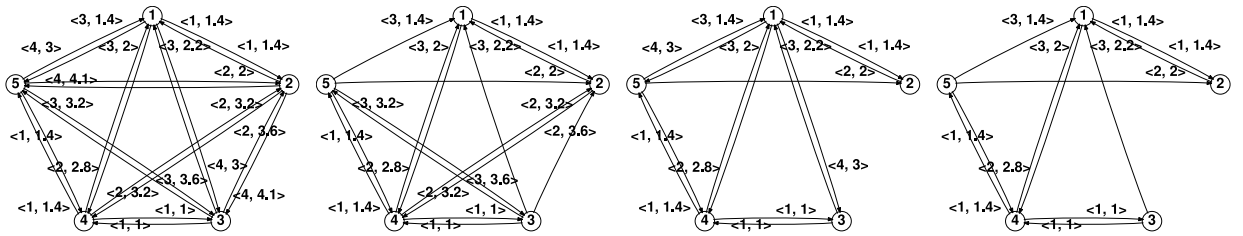
These parameters can be easily implemented as timestamp (date and time), source position (geo-coordinates) and coverage distance, attached in data headers. Data is discarded once TTL/STL expires. When two nodes get in contact, data is forwarded only if the position of the receiving node satisfies STL limit. Spatiotemporal dissemination can be regarded as journeys from a source node to all other destination nodes such that for all journeys, temporal and spatial lengths do not exceed TTL and STL.

Using the example scenario shown in Fig. 1, spatiotemporal dissemination is exemplified as follows. Suppose a source node is Node 3, departure time is $t_s = 1$, TTL and STL are three units, and edge traversal time is one unit for each edge. Note that edge traversal time can be arbitrary for each edge over time. In the figure, dotted circles correspond to the spatial-validity area with the radius γ, $STL = 3$ units, from the source position at coordinates of the point $(5, 3)$, whereas filled nodes indicate presence of data. At $t = 1$, Node 3 begins forwarding data to Node 4. The transfer is completed at $t = 2$, after one unit of traversal time. At $t = 2$, Nodes 3 and 4 have the data, and Node 4 has a contact with Node 2. However, the data is not forwarded because this spatial length becomes greater than the STL. At $t = 3$, Node 3 has a contact with Node 5, and Node 4 has contacts with Nodes 1 and 2. Node 4 forwards the data to Node 1. The data arrives at Node 1 at $t = 4$. Data is not forwarded to Node 5 as the temporal length of journeys are greater than the TTL.

3.3. Definition of STRG

Spatiotemporal reachability graphs capture end-to-end connectivity across space and time. In Fig. 1, the subgraphs within the dotted circle depict *spatio-time-varying graphs (STVG)* focusing on the spatial-validity area. Some edges may exist in TVG but not in STVG. In Fig. 2, the edges are represented by a tuple $\langle L_T, L_S\rangle$, where L_T is the temporal length and L_S is the spatial length. Fig. 2a depicts an example of reachability graph for the scenario shown in Fig. 1. Fig. 2b is an example of temporal reachability graph for the same scenario. This graph satisfies that temporal length of journeys do not exceed TTL. Fig. 2c depicts an example of spatial reachability graph for the same scenario. The graph is derived from STVG in Fig. 1. In this graph scenario, spatial length of journeys do not exceed STL. Fig. 2d is an example of spatiotemporal reachability graph (STRG) for the same scenario. In this graph, temporal and spatial lengths of journeys do not exceed TTL and STL, respectively. Each edge corresponds to a source–destination node pair that can communicate within TTL and STL.

Spatiotemporal Reachability Graphs (STRG). Given departure time, TTL, and STL, let $G_R^{\delta\gamma} = (V_R^{\delta\gamma}, E_R^{\delta\gamma})$ be STRG, where δ and γ represent TTL and STL. A directional edge $e_{s,d}$ corresponds to the *earliest* journey from a source node s to a destination node



(a) Reachability graph. (b) Temporal reachability graph. (c) Spatial reachability graph. (d) Spatio-temporal reachability graph.

Fig. 2. Reachability graphs for the example scenario in Fig. 1 (from $t_s = 1$), where TTL and STL are each three units. Due to the limitation on space, we put the $\langle L_T, L_S \rangle$ values near the destination nodes (i.e., near the arrow for every edge).

Algorithm 1 *journey_search*(G, t_s)

Input: TVG $G = \{V, E\}$ and departure time t_s
Output: A list *edgelist*, which gives all source–destination node pairs with the spatiotemporal length

for $s \in V$ **do**
  $L_T[s] \leftarrow t_s$; $L_S[s] \leftarrow 0$
  for $\{v \in V \mid v \neq s\}$ **do**
    $L_T[v] \leftarrow \infty$; $L_S[v] \leftarrow \infty$
  $Q \leftarrow \{s\}$
  while $Q \neq \emptyset$ **do**
    pull u from Q
    for $\{v \in V \mid e_{u,v} \in E)\}$ **do**
      $t_{earliest}, d \leftarrow select(e_{u,v}, L_T[u])$
      if $t_{earliest} + \zeta < L_T[v]$ **then**
        $L_T[v] \leftarrow t_{earliest} + \zeta$
        if $L_S[v] = \infty$ **then**
          $L_S[v] \leftarrow d$
        else
          $L_S[v] \leftarrow max(L_S[v], d)$
        if $v \notin Q$ already **then**
          push v into Q
          $sort(Q, L_T)$
  for $\{v \in V \mid v \neq s\}$ **do**
    $edgelist \leftarrow s, v, L_T[v] - t_s, L_S[v]$
return *edgelist*

d at departure time t_s such that for each journey, the temporal length is less than or equal to TTL, and the spatial length is less than or equal to STL. Let E_R^{inv} be the set of invalid edges whose temporal length is greater than TTL, or the spatial length is greater than STL. STRG is built up by removing the set of invalid edges from a reachability graph, G_R.

$$G_R^{\delta\gamma} = G_R - E_R^{inv} \tag{1}$$

STRG is the intersection of two graphs—the temporal reachability graph, G_R^{δ}, and the spatial reachability graph, G_R^{γ}.

$$G_R^{\delta\gamma} = G_R^{\delta} \cap G_R^{\gamma} \tag{2}$$

For example, when TTL and STL are each equal to 3 units for the reachability graph in Fig. 2a, the set of invalid edges is described as $E_R^{inv} = \{e_{1,3}, e_{1,5}, e_{2,3}, e_{2,4}, e_{2,5}, e_{3,2}, e_{3,5}, e_{4,2}, e_{5,3}\}$. STRG in Fig. 2d is obtained by removing the set of invalid edges from the reachability graph. STRG in Fig. 2d comprises the common edges to both of the temporal reachability graph in Fig. 2b and the spatial reachability graph in Fig. 2c.

3.4. Algorithm for building STRG

The algorithm first derives a reachability graph from TVG. Based on the reachability graph, STRG is constructed by satisfying space and time requirements.

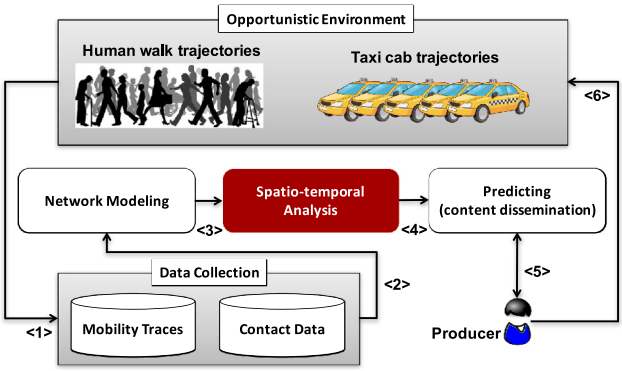


Fig. 3. Workflow of the analytical framework. The main contribution of this paper focuses on the spatio-temporal analysis as highlighted. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

Algorithm 1 details how to find available journeys with the spatiotemporal length. Given TVG containing all graph snapshots for the lifetime of a system and the departure time, the algorithm finds the *earliest* journeys and then returns an edge list, comprising all source–destination node pairs and corresponding spatiotemporal length. In Line 1, a source node is iteratively picked. Lists of spatial and temporal lengths and a min-heap priority queue are initialized in Lines 2–5. In the while loop of Lines 6–18, the algorithm finds journeys from the source node to all the other destination nodes and ends this loop when the queue becomes empty. In Line 7, a node u is pulled from the top of the queue, which has the minimum temporal length in the queue. In the for loop of Lines 8–18, each neighbor v of u is traversed. In Line 9, *select* function returns the earliest time when nodes u and v contact after the arrival time of u. The function also returns the longer of two distances between the source position and u or v at the earliest time. When nodes u and v do not have any contact after the arrival time of u, the function returns infinite values for both of them. If the earliest time plus edge traversal time is less than the temporal length to node v, then the length is updated with the time in Lines 10–11. Furthermore, if the spatial length to node v is the initial value, then the length is updated with the distance returned by *select* function in Lines 12–13; otherwise, the length is updated with the longer of the spatial length and the distance in Lines 14–15. In Lines 16–18, if node v was not in the queue already, then node v is pushed into the queue and the queue is sorted in ascending order of temporal length. After subtracting the departure time from each temporal length, the edge list is obtained in Lines 19–20. Last, the algorithm returns the edge list in Line 21. A reachability graph is derived from the edge list with the spatiotemporal length. Once the reachability graph is built up, STRG can be constructed by removing the set of invalid edges for any TTL and STL. This algorithm works like the well-known algorithm of Dijkstra, additionally considering time-varying nodes/edges and temporal/spatial length.

STRG can be constructed with any opportunistic routing protocol such as epidemic [2], Spray-and-Wait [26], PRoPHET [27], BUBBLE Rap [4], and so on. Different routing protocols construct STRG with different spatio-temporal length. In TVG, there are some routing metrics: minimum number of hops, earliest arrival time, and shortest time span between departure time and arrival time [24]. In this paper, STRG utilizes the metric of earliest arrival time, called the earliest journey, whose dissemination process is similar to a simple epidemic routing algorithm. In the epidemic routing, when a node having data comes in contact with another node not having the data, the former node transmits the data to the latter. Eventually, the data is delivered to all reachable destinations. Although the epidemic routing is expensive for network resources (e.g., bandwidth and energy), it maximizes the chances of connectivity and minimizes latency. In many cases, it can serve as an upper-bound for any other routing protocols.

4. Analytical framework

In opportunistic environments, mobile nodes can act as data ferries between disconnected domains of a highly partitioned network. Content is distributed by store-carry-forward mechanism as follows. Content generated by a producer is stored in its buffer and carried forward. When a producer node is within communication range of another node, the content can be forwarded to the latter and stored in its buffer. This process is repeated until the content is delivered to all other consumers through a series of opportunistic contacts between pairs of mobile users. In effect, the mechanism realizes content dissemination in opportunistic environments.

Fig. 3 illustrates a workflow of analytical framework for spatiotemporal content dissemination. The contribution of this article is on the "spatial–temporal Analysis" as highlighted in red box in the figure. In an opportunistic environment, mobile users move around freely. GPS data is sent from mobile-user devices to a data collection server which constructs the mobility traces ⟨1⟩. Contact data are derived from mobility traces by comparing distance between two devices and their transmission ranges. With the mobility traces and contact data, user mobility and device connectivity are modeled ⟨2⟩, which is detailed in Section 3.1. Then, the framework performs spatiotemporal analysis for content dissemination ⟨3⟩, which is detailed in Section 3. Based on the analysis, the framework predicts content distribution parameters in the mobility scenario ⟨4⟩, the

novel contribution of the paper is detailed in Section 5. To obtain such parameters, a content producer creates and requests a query to the analytical framework, and then the producer receives the response with the predicted parameters from the framework ⟨5⟩. The producer can make a decision to spread content over the opportunistic environment or not ⟨6⟩. Also, the producer can use the optimal parameter setting to spread content over the opportunistic environment.

As a key metric on spatiotemporal content dissemination in opportunistic environments, dissemination ratio is defined as the number of consumers that receive content within the given TTL and STL to the total number of consumers. The goal of this paper is to estimate content distribution parameters for the following queries:

- *Query* 1. Given TTL and STL, what is the estimated dissemination ratio?
- *Query* 2. Given file size, TTL and STL, is it possible to successfully deliver the file to consumers and what is the estimated dissemination ratio?
- *Query* 3. What are the optimal TTL and STL values to achieve a given dissemination ratio?

For these queries, we present some examples of application scenario that will be supported by the proposed framework: geographic notification for weather forecast in a theme park (Scenario A), geographically-restricted video file sharing on campus (Scenario B), and location-targeted advertisement for marketing in a city (Scenario C).

In Scenario A, a producer generates content regarding severe thunderstorm warning that says it will appear over the theme park an hour later. The producer wants to know the ratio of visitors in the park to receive the content. Query 1 is made by the producer, where parameters such as content generation time, TTL, and STL can be automatically determined by the application context. Receiving the estimated dissemination ratio from the framework, the producer spreads content out in the opportunistic environment if it is acceptable; otherwise, the producer might choose an alternative way, e.g., via cellular networks.

In Scenario B, a producer generates a 15 MB video file of a keynote speech at an academic conference. The producer wants to know the ratio of students that receive the content and the possibility of successful dissemination. Query 2 is made by the producer. In this case, STL is determined so that dissemination covers the entire campus area, whereas TTL is determined freely by the producer. Receiving the estimated transmission capacity and dissemination ratio from the framework, the producer can disseminate the content if the file size is less than the capacity; otherwise, the producer might give up doing that.

In Scenario C, a producer generates content advertising 30% saving at Restaurant XYZ in the city. The advertisement cost might rise up with increasing TTL and STL in order to avoid extra traffic. The producer decides to spread the content to 40% of visitors in the city and wants to know the optimal parameter setting for the desired dissemination ratio. Query 3 is made by the producer. Receiving the optimized parameter setting from the framework, the producer can use the parameter for content dissemination, which prevents the producer from setting too long/short TTL and STL.

Prediction accuracy depends on the collected mobility datasets and the mobility traces of entities used. However, similar to other machine learning algorithms our algorithm focuses on the mobility pattern rather than specific coordinates within a trajectory. Thus, the model learned can tolerate variations in the input dataset. We are conducting separate experiments [28] as part of our ongoing work to study impact of variation in mobility dataset on prediction accuracy.

5. Predicting scheme for content dissemination

This section details a predicting scheme for content dissemination. First, analytic functions are developed based on the spatiotemporal property captured by STRG. Then, a predicting scheme is elaborated to infer future parameters of content distribution based on only historical parameters.

5.1. Estimating dissemination ratio

A spatiotemporal property of dissemination is captured by Algorithm 1 for building STRG. The algorithm can find available source–destination node pairs (s–d pairs) with the spatiotemporal length at any departure time in any mobility scenario.

For the example scenario shown in Figs. 1, 4 depicts the spatiotemporal length distribution at departure time $t_s = 1$. For each s–d pair, temporal and spatial lengths are plotted on the graph. Thus, each dot represents an available s–d pair, which can communicate at the estimated temporal and spatial lengths. If s–d pairs have infinite temporal or spatial length, the s–d pairs are not plotted on the graph. For example, as described in Section 3.1, the temporal and spatial lengths of $J_{3,1}$ are three units and 2.236 units. Some s–d pairs overlap, e.g., two dots at $(1.0,\ 1.0)$. As seen in the figure, some dots are in a line on the temporal length. In opportunistic networks, data are disseminated by store-carry-and-forward mechanism. Let us consider the following example, where there are n nodes within the same cluster and one isolated node in an area. At initial time $t = t_s$, all source nodes disseminate their data at different source positions. During some time period, these n nodes exchange data within the same cluster, whereas the isolated node has no contacts with anyone. Then, one node, say n_k, collects $n - 1$ data from the other nodes within the same cluster. When n_k encounters an isolated node, say n_i, $(n - 1)$ data are received at the latter. For these $n - 1$ data received at n_i, the temporal lengths are same, whereas the spatial lengths are different because source positions are different at $t = t_s$. In contrast, dots in a line on the spatial length are fewer. It happens only when maximum distances traveled by data from source positions are same but their arrival times are different. However, this case is less frequent compared to the former.

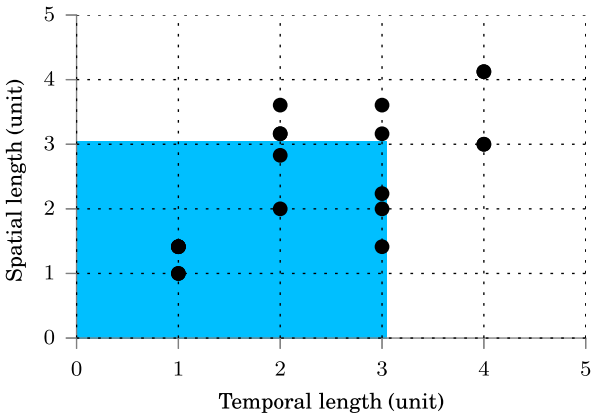


Fig. 4. Spatiotemporal length distribution for the example scenario in Fig. 1, where each dot corresponds to a source–destination node pair at departure time $t_s = 1$ and a rectangle area represents spatio-temporal frame with $TTL = STL = 3$ (units).

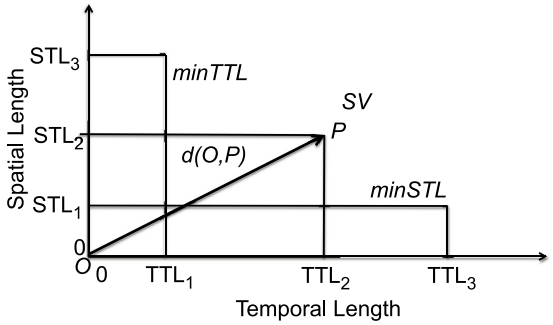


Fig. 5. Optimization criteria, where three rectangles represent three types of the optimized $\delta \times \gamma$ STF.

To facilitate response to user queries, we develop an analytic function for content dissemination. The following function relates to Query 1: given TTL and STL, determine the estimated dissemination ratio (R). The function can be broadly defined as $R = f(t_s, TTL, STL)$, where t_s is the start time of dissemination.

Let $\delta \times \gamma$ be the spatiotemporal frame (STF), which is a rectangle area bounding the spatiotemporal length. Let $|P_{\delta\times\gamma}|$ be the number of available s–d pairs within $\delta \times \gamma$ STF. For each s–d pair within STF, the temporal (spatial) length is less than or equal to TTL (STL). The dissemination ratio is estimated as the ratio of the number of s–d pairs within the $\delta \times \gamma$ STF to the total number of possible s–d pairs. The total number of possible s–d pairs is given by $|V|(|V| - 1)$, where $|V|$ is the number of nodes in a mobility scenario.

$$R = \frac{|P_{\delta\times\gamma}|}{|V|(|V| - 1)}. \tag{3}$$

The dissemination ratio is the average value for all source nodes, instead of specifying a source node. Thus, the bottom part in Eq. (3) is the number of possible end-to-end journeys via *store-carry-forward* mechanism, not via direct transmission.

A sample of spatiotemporal frame is depicted as a rectangle area in Fig. 4. For the example scenario of Fig. 1, considering from $t_s = 1$ where for every node $TTL = STL = 3$ units the reachability R of all nodes is estimated as $11/(5 \times 4) = 0.55$. Note, $|P_{\delta\times\gamma}| = 11$ in Fig. 2d and there are five nodes considered in the example. Of course, this is an example of computing R for the node pairs that satisfy the spatiotemporal constraint. We can also compute R for a specific node by changing $|P_{\delta\times\gamma}|$ to the number of available s–d pairs where s is the specific node.

5.2. Optimizing parameter setting

This analytic function relates to Query 3: determine optimal TTL and STL values to achieve a desired dissemination ratio. The function optimizes the parameter setting at t_s, where t_s is the start time of dissemination.

Fig. 5 illustrates optimization criteria, where three rectangles represent three types of the optimized $\delta \times \gamma$ STF: (i) *minTTL*, (ii) *minSTL*, and (iii) *SV*, where $SV = dist(O, P)$ such that P is the point that corresponds to the shortest vector. Temporal and

Algorithm 2 *parameter_optimization*(*edgelist*, R^D)

Input: *edgelist*, desired dissemination ratio R^D
Output: Tuples *minTTL*, *minSTL*, and *SV* to be optimized by the minimum TTL, the minimum STL, and the shortest vector

```
1: minTTL ← ∞, ∞; minSTL ← ∞, ∞; SV ← ∞, ∞, ∞
2: L_T^max, L_S^max ← max(edgelist)
3: L ← min(L_T^max, L_S^max)
4: for k = 1 to L do
5:     if R(k, k) ≥ R^D then
6:         update_parameters(k, k)
7:         break
8:     for i = k to L_S^max do
9:         if R(k, i) ≥ R^D then
10:            update_parameters(k, i)
11:    for j = k to L_T^max do
12:        if R(j, k) ≥ R^D then
13:            update_parameters(j, k)
14: return minTTL, minSTL, SV
15: procedure UPDATE_PARAMETERS(TTL, STL)
16:    if TTL < minTTL[0] then
17:        minTTL ← TTL, STL
18:    if STL < minSTL[1] then
19:        minSTL ← TTL, STL
20:    if dist(O, P) < SV[2] then
21:        SV ← TTL, STL, dist(O, P)
```

spatial lengths are normalized with their corresponding maxima, i.e., L_T^{max} and L_S^{max}. Then, $dist(O, P)$ is calculated by,

$$dist(O, P) = \sqrt{\left(\frac{TTL}{L_T^{max}}\right)^2 + \left(\frac{STL}{L_S^{max}}\right)^2} \tag{4}$$

Algorithm 2 details how to optimize parameter setting. Given an edge list obtained by Algorithm 1 and a desired dissemination ratio, Algorithm 2 returns three pairs of TTL and STL optimized by three criteria: *minTTL*, *minSTL*, and *SV*. First, three pairs of parameter settings are initialized in Line 1, where the first and second variables are used for TTL and STL, and the third variable of *SV* is used for the distance. In Line 2, *max* function returns the maximum spatial and temporal lengths in the edge list, and then *min* function returns the shorter of these lengths in Line 3. In the for loop of Lines 4–13, $\delta \times \gamma$ STF is expanded incrementally on the diagonal of the area $L_T^{max} \times L_S^{max}$. If the current R calculated by Eq. (3) is greater than the desired dissemination ratio, then the procedure *update_parameters* is called with the corresponding TTL and STL in Lines 5 and 6. In the update procedure, if the current TTL is less than TTL of *minTTL*, then *minTTL* is updated with the current TTL and STL in Lines 16 and 17. If the current STL is less than STL of *minSTL*, then *minSTL* is updated with the current TTL and STL in Lines 18 and 19. If the current distance calculated by Eq. (4) is less than the distance of *SV*, then *SV* is updated with the current TTL, STL, and distance in Lines 20 and 21. When Line 5 is true, the algorithm terminates the for loop of Lines 4–13 in Line 7 and then returns the optimized parameter settings in Line 14. In the for loop of Lines 8–10 or 11–13, $\delta \times \gamma$ STF is expanded incrementally to the maximum spatial or temporal length. In these for loops, if the current R is greater than the desired dissemination ratio, then *update_parameters* is called in Lines 9 and 10, or 12 and 13.

5.3. Testing transmission capacity

This analytic function relates to Query 2: given file size, TTL and STL, test the possibility to successfully deliver the file to consumers and determine the estimated dissemination ratio. The function estimates the transmission capacity within a certain time window between t_s and $t_s + TTL$, where t_s is the start time of dissemination. If the file size is not greater than the capacity, the whole content is completely disseminated; otherwise, the content is partially disseminated utilizing available capacity. The later query is estimated by Eq. (3) in Section 5.1.

In opportunistic environments, mobile devices get connected with each other intermittently. Data are transferred by exploiting direct contact opportunities between two devices. Hence, the environments can be characterized by combined individual mobility and pairwise-contact patterns. Contact and inter-contact times are well-known properties in opportunistic networks [29,30]. In this function, contact time is measured to estimate a transmission capacity within a specific time window.

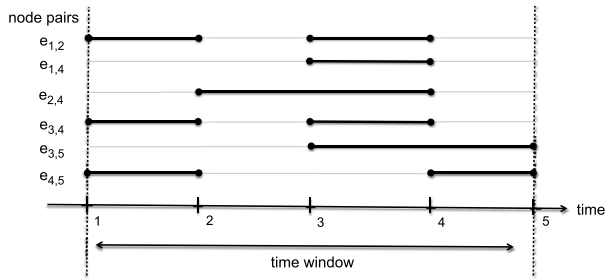


Fig. 6. Contact time for the example scenario in Fig. 1, where a line corresponds to the contact time for a node pair.

Contact time. Contact time, also called link lifetime or link duration is the time interval during which a given pair of nodes are in each other's transmission range. The value is useful to estimate the volume of transferable data in each connection opportunity.

Let CT be the contact time within a given time window. A transmission capacity (TC) is estimated as the product of the contact time and the bottleneck bandwidth (BW), which is the slowest data rate on a series of contacts between source and destination nodes.

$$TC = CT \times BW. \tag{5}$$

In a conservative estimate, contact time is the modal value within the time window, where the statistical term "mode" refers to the most frequently occurring number found in a set of numbers. In more aggressive estimate, contact time is the mean value within the time window.

Fig. 6 illustrates the contact time for the example scenario shown in Fig. 1. Suppose $t_s = 1$ and $TTL = 4$ units. In the given time window, there are total nine contacts. There are seven contacts of duration one unit while two contacts are of two units. For these contacts, minimum, modal, median, mean, and maximum contact times are 1, 1, 1, 1.22, and 2 in units, respectively. In a conservative estimate, contact time is 1 unit. In an aggressive estimate, contact time is 1.22 units.

5.4. Predicting scheme

A predicting scheme is developed to infer future parameters of content distribution based on historical parameters.

In real-world traces, contact opportunities depend highly on the mobility of individual users. The predicting scheme distinguishes weekdays/weekends and time windows for the historical parameters. In content dissemination, date and time are given by applications. To infer future parameters, historical parameters are chosen according to the following history-based methods.

- *RDRT*. Prediction is based on a random day of the last week at random time window.
- *SDST*. Prediction is based on the same day of the last week at the same time window as given time.
- *PDST*. Prediction is based on the previous day at the same time window as given time.
- *AWST*. Prediction is based on the average value of the last week at the same time window as given time.

In RDRT, differences between weekdays/weekends and time windows are neglected. In this paper, RDRT serves as a baseline in order to quantify the superiority of the proposed predicting scheme over the baseline. In PDST and AWST, weekdays and weekends are differentiated, (e.g., the previous day of Monday is last Friday). These predicting methods use only historical parameters and do not require any other knowledge in future.

6. Evaluation

This section assesses the predictive power of the analytical framework for opportunistic content dissemination. First, long-term empirical datasets are introduced. Then, a basic analysis is performed to characterize a spatiotemporal pattern of contact opportunities. After presenting a validation scheme, extensive simulations are performed to validate the predicting scheme with three analytic functions.

6.1. Long-term empirical datasets in urban areas

We are interested in long-term mobility traces with a set of nodes to examine a spatiotemporal pattern in opportunistic environments. We utilize two large taxicab traces collected in urban areas, San Francisco [31] and Rome [32,33] . Each trace contains tens of million trajectory points for a period of one month, where a taxicab's trajectory is a sequence of positions with the corresponding timestamps. Taxicab trajectories provide an extensive coverage on city streets, unlike other

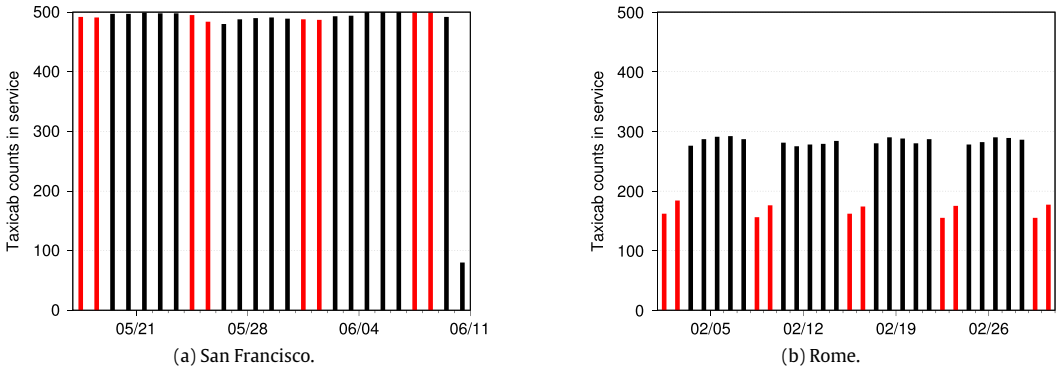


Fig. 7. Taxicab counts in service on each day through entire trace durations.

transportation vehicles, e.g., buses move mainly on main streets. A taxi service also works 24/7 in shifts. A taxicab can be in either moving to serve customers (i.e., pick up or drop off) or staying at parking spaces (i.e., idle) to wait for new customers. Each taxicab visits locations independently according to the random nature of customer's demand for destinations. In a future scenario of smart city, a fleet of taxicabs could provide a service of data dissemination/collection within a specific city without requiring big expenses.

San Francisco traces. GPS trajectories are collected from 536 taxicabs in the San Francisco Bay area, California, USA. The dataset contains 11,219,955 trajectories (0.4 GB file size) for a period of 25 days from May 17th to June 10th in 2008. Each trajectory includes position (i.e., GPS coordinates of latitude and longitude), occupancy (i.e., ride or idle), and timestamp (i.e., date and time) while each file corresponds to a taxicab ID.

Rome traces. GPS trajectories are collected from a fleet of 370 taxicabs moving around the city of Rome, Italy. The dataset contains 21,817,850 trajectories (1.6 GB file size) with fine sample rate of 7-s intervals for a period of 30 days from February 1st to March 2nd in 2014. Each trajectory includes taxicab ID, timestamp, and position.

6.2. Spatiotemporal property of contact opportunities

Different cities have different properties in area size, population, road infrastructure, and so forth. For San Francisco and Rome traces, a basic analysis is performed to characterize spatiotemporal patterns of taxicab trajectories and contact opportunities, e.g., how taxicabs move in the city, how contact counts change over time of the day, and how contacts are distributed in the city. Assuming that all taxicabs are equipped with WiFi Direct to communicate with each other, their transmission ranges are set to 75 m. The whole traces are divided into discrete time slots of 30 s to compare these traces.

Taxicab counts in service on each day through the entire trace durations are shown in Fig. 7. In San Francisco traces, there are always around 500 taxicabs everyday except for the last day on June 10th. In Rome traces, there are around 290 taxicabs on weekdays while there are around 170 taxicabs in weekends. Naturally, Rome traces have a significant decrease in taxicab counts during weekends.

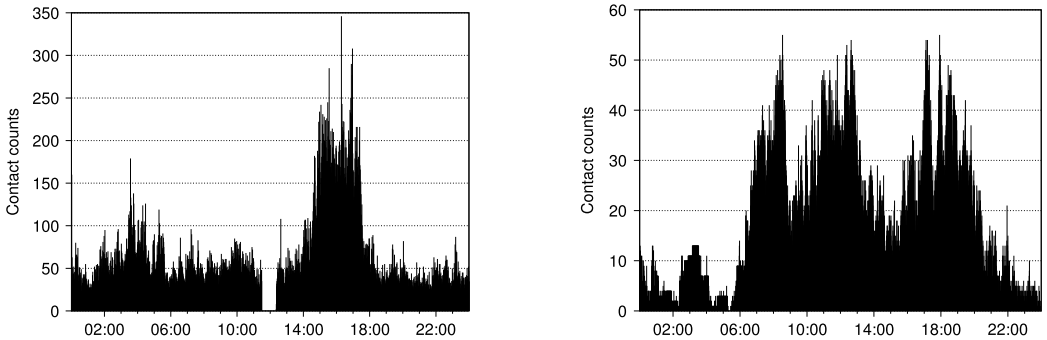
Samples of taxicab trajectories are shown in Fig. 8. Total trajectory counts in a day for San Francisco traces are 433,461 on Wednesday May 21st, 2008. Rome traces have 235,430 trajectories on Wednesday February 5th, 2014. San Francisco traces are two times more than Rome traces because taxicab counts are different between these traces. In both traces, most trajectories are concentrated on a certain region as shown in the topright (center) region for San Francisco (Rome) traces. From a geographic viewpoint, the degree of concentration increases from suburban regions to the city center. Taxicabs move with somewhat random paths while geographic distributions of trajectories varies during the day and across different weeks. Such concentrations are observed at the identical geo-locations everyday in each trace. This is because workplaces and residential places are not frequently changed in a city.

Fig. 9 shows samples of contact counts at different times of the day. San Francisco traces show 153,207 contacts in a day compared Rome traces that show 52,902 contacts. This is expected as there are more taxicabs in the San Francisco traces. We count the number of contact opportunities for every 30 s. In San Francisco traces, a peak time when most contacts happen is in the evening. In Rome traces, the corresponding peak time happens at morning, noon, and evening on weekdays while it happens in the night on weekends. In each trace, there is a significant difference in counts between daytime and midnight because most people go to work on weekdays during daytime and stay at home during the night. Thus, density of connectivity between nodes is changes over time, whereas the peak time depends on a specific city.

Fig. 10 shows samples of contact counts across space in a day. Considering different area sizes, we divide the entire area of San Francisco and Rome traces into 300×300 and 100×100 regions, respectively. Then, we count the number of taxicabs falling into the same region. As seen in the figure, we can observe a region of contact hotspots, where most contacts happen. We find that contact hotspots are highly correlated with geo-locations of dense trajectories, e.g., around the city center in

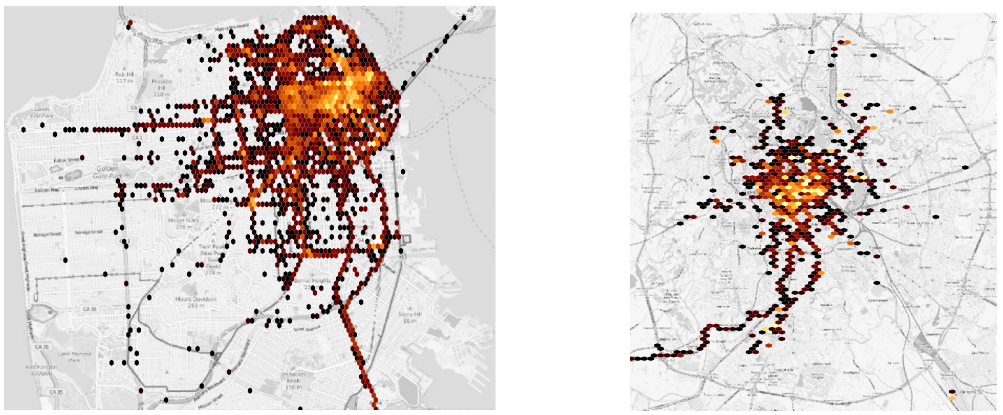
(a) San Francisco on Wed. May 21. (b) Rome on Wed. February 5.

Fig. 8. Samples of taxicab trajectories in a day.



(a) San Francisco on Wed. May 21. (b) Rome on Wed. February 5.

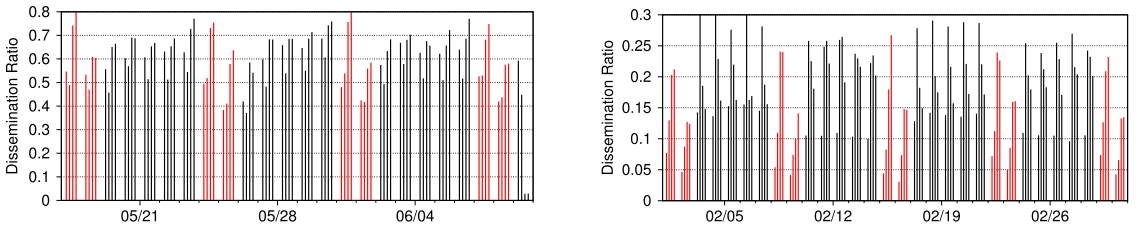
Fig. 9. Samples of contact counts according to time in a day, where the x-axis represents time of the day.



(a) San Francisco on Wed. May 21. (b) Rome on Wed. February 5.

Fig. 10. Samples of contact counts across space in a day, where dark/light-colored regions represent a small/large number of contact opportunities. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

these traces. The contact hotspots are located at popular places where people visit most frequently. From a traffic viewpoint, such places increase the probability of contacts due to high traffic volume (i.e., traffic jam) and lower vehicle speeds. We also observe the stability of contact hotspots within the entire period in each trace.



(a) San Francisco from May 17 to June 9.

(b) Rome from Feb 1 to March 2.

Fig. 11. Observation results of dissemination ratio with TTL = 4 (h) and STL = 16 (km) through entire trace durations, where the *x*-axis represents date and time.

6.3. Validation scheme

In the following simulation, we apply analytic functions on each day to observe how content distribution parameters change for both traces. For each day, content dissemination is generated from each source node to all other nodes at different times of the day: 0600 h, 1000 h, 1400 h, and 1800 h. We evaluate the effectiveness of the framework, comparing history-based predicting methods with a random method that serves as a baseline. For each method, prediction errors are measured to examine how each method is robust against uncertain future mobility. We utilize a well-known validation scheme in machine learning: cross-validation and mean absolute error (MAE). In cross-validation (sometimes called rotation estimation), the original sample dataset is split into training ad testing sets. MAE is a validation metric to measure overfitting (i.e., over/under-estimate).

$$MAE = \frac{1}{n}\sum_{i=1}^{n}\left|v_{pred} - v_{true}\right| \tag{6}$$

n is the sample size, v_{pred} represents predicted values, and v_{true} indicates ground-truth values obtained from a specific date and time given by applications. A training set is determined by each method, whereas a testing set is randomly picked for a pair of date and time. In San Francisco traces, date and time are chosen from a period of 21 days between May 26th and June 9th. In Rome trace, the corresponding period is 15 days between February 10th and March 2nd.

This paper proposes an analytical framework to estimate content distribution parameters, where any routing protocol is acceptable for the framework as mentioned in Section 3.4. Some performance metrics such as delivery ratio and latency are examined as dissemination ratio and temporal length, respectively. Other metrics for routing protocols (e.g., routing overhead and energy efficiency) are beyond the scope of this paper.

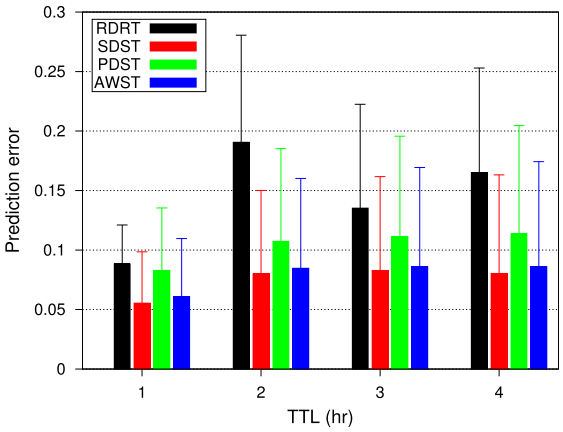
6.4. Estimating dissemination ratio

Observation results of dissemination ratio through the entire trace duration are shown in Fig. 11, where $TTL = 4$ h and $STL = 16$ km. The *x*-axis shows date and time while the *y*-axis shows dissemination ratio. Each line in 4-line clusters corresponds to 4 different time windows: 0600, 1000, 1400 and 1800 h. In general, San Francisco traces show higher dissemination ratios compared to those of Rome traces. The average (standard deviation) dissemination ratio is 0.57 (0.17) in San Francisco traces and 0.17 (0.07) in Rome traces. In San Francisco traces, a peak time window with the highest dissemination ratio is between 1400 and 1800 h, everyday. In Rome traces, the corresponding peak time window is between 1000 and 1400 h on weekdays and between 1400 and 1800 h on weekends. In both traces, there is a significant difference between weekdays and weekends while the peak time window is repeated. In general, Eq. (3) depicts the dissemination ratio for any TTL and STL.

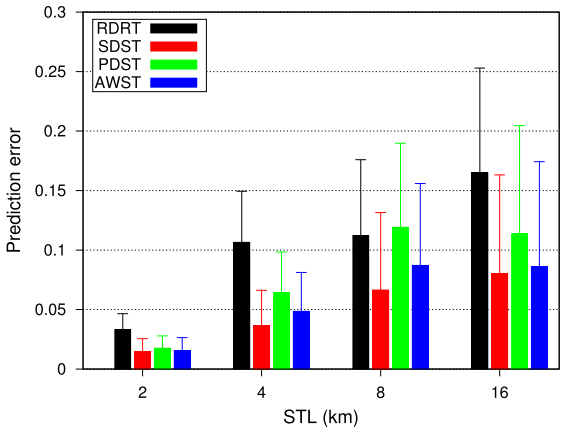
Figs. 12 and 13 show the validation results of predicting dissemination ratio in San Francisco and Rome traces. The results are the average of 21 or 15 independent simulation runs, shown with 95% confidence intervals. San Francisco traces show higher prediction errors than Rome traces because of different dissemination ratio. Also, the prediction error increases with the TTL/STL values. SDST outperforms the other prediction methods in San Francisco traces while PDST outperforms the others in Rome traces. This is because San Francisco traces exhibit a weekly-periodic behavior compared to Rome traces as seen in Fig. 11. With the best predicting method, the error value at TTL = 4 (h) and STL = 16 (km) is 0.08 in San Francisco traces and 0.03 in Rome traces. That means more than 80% accuracy for the average dissemination ratio in the above observation. The predicting method achieves more than 50% reduction of prediction error, compared to RDRT.

6.5. Optimizing parameter setting

Observation results of the parameter setting optimized by the shortest vector method are shown in Figs. 14 and 15. The desired dissemination ratio is set to 0.5 in San Francisco traces and 0.2 in Rome traces because these traces do not often

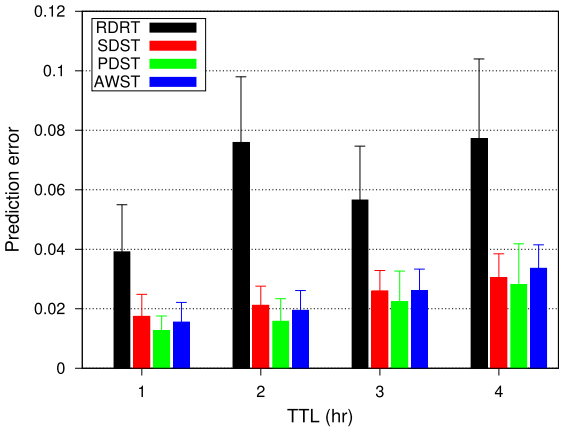


(a) Impact of TTL on prediction errors with STL = 16 (km).

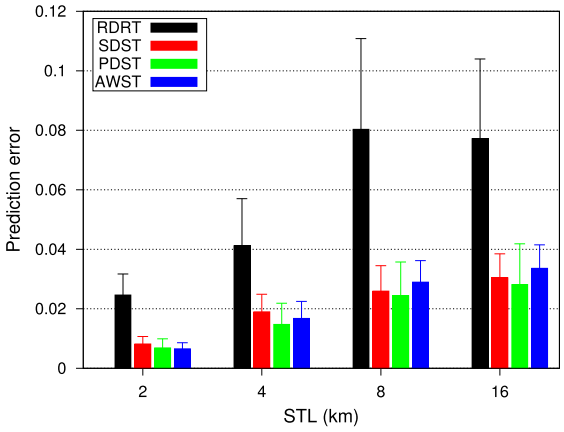


(b) Impact of STL on prediction errors with TTL = 4 (h).

Fig. 12. Validation results of predicting dissemination ratio in San Francisco.



(a) Impact of TTL on prediction errors with STL = 16 (km).



(b) Impact of STL on prediction errors with TTL = 4 (h).

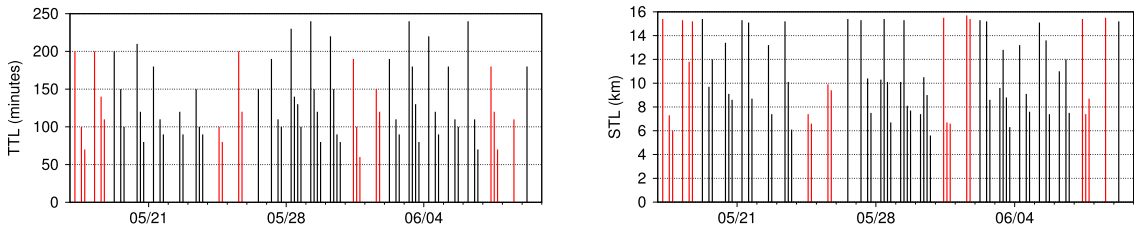
Fig. 13. Validation results of predicting dissemination ratio in Rome.

reach more than these values. The *x*-axis is date and time while the *y*-axis is TTL/STL. Each line in 4-line clusters corresponds to 4 different time windows. Through the entire duration, Rome traces are less frequently optimized than San Francisco traces because Rome traces cannot often achieve the desired dissemination ratio with any parameter setting within the time window. In both traces, the optimized parameter setting exhibits a similar pattern weekly while there is a difference between weekdays and weekends. In general, Algorithm 2 can optimize the parameter setting for any desired dissemination ratio.

Fig. 16 shows the validation results of predicting optimal parameter setting in San Francisco and Rome traces. The results are the average of 40 independent simulation runs, shown with 95% confidence intervals. For each trial, the optimized parameter setting is predicted for a given date and time, and then dissemination ratio is obtained by applying the parameter to the same date and time. In both traces, the prediction error gets larger according to larger desired dissemination ratio. SDST outperforms the other prediction methods at the desired dissemination ratio of 0.6 in San Francisco traces while PDST outperforms the others at the corresponding ratio of 0.2 in Rome traces. With the predicting method, the error value is 0.13 in San Francisco traces and 0.05 in Rome traces. That means more than 75% accuracy for the desired dissemination ratio. The predicting method achieves around 50% reduction of prediction error, compared to RDRT.

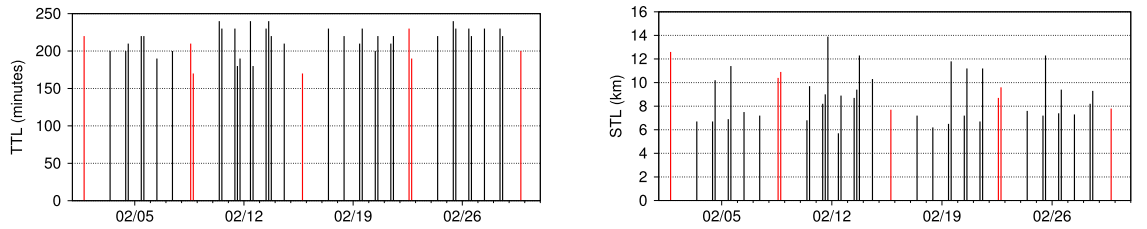
6.6. Testing transmission capacity

Samples of cumulative distribution function (CDF) of aggregate contact times in a day are shown in Fig. 17. The *x*-axis is the contact time while the *y*-axis is the probability. In San Francisco traces, almost all contact times are 30 s in every



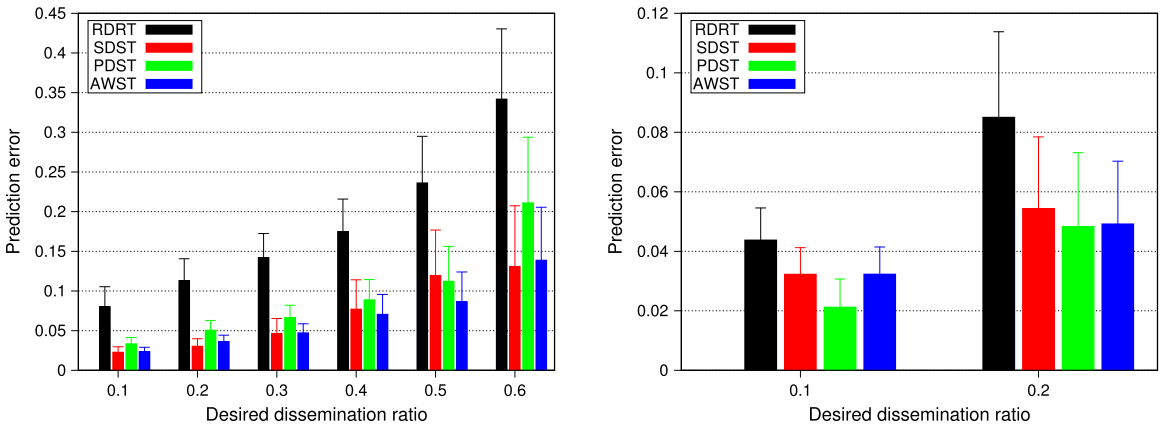
(a) Optimized TTL values. (b) Optimized STL values.

Fig. 14. Observation results of optimal parameter setting in San Francisco traces from May 17 to June 9, where desired dissemination ratio is 0.5 and the *x*-axis represents date and time.



(a) Optimized TTL values. (b) Optimized STL values.

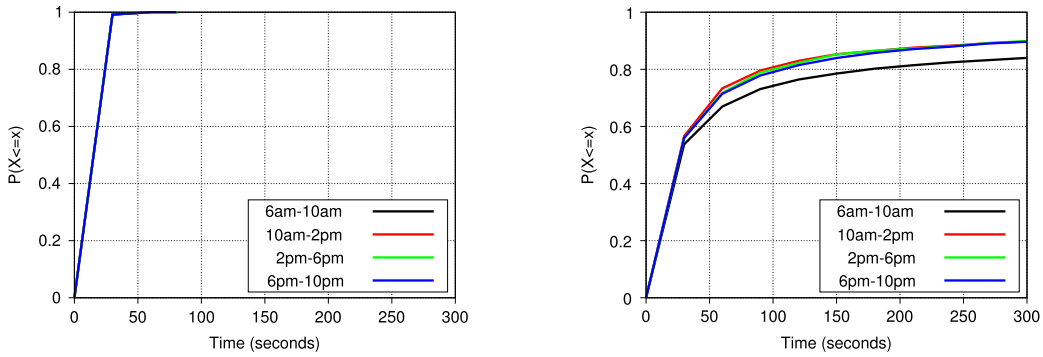
Fig. 15. Observation results of optimal parameter setting in Rome traces from Feb 1 to March 2, where desired dissemination ratio is 0.2 and the *x*-axis represents date and time.



(a) San Francisco. (b) Rome.

Fig. 16. Validation results of predicting parameter setting with impact of desired dissemination ratio on prediction errors.

time window. This is mainly because the traces have coarse sample rate of time intervals for consecutive trajectory points and are therefore not suitable for a contact time analysis. Thus, we focus on only Rome traces in the following simulation. In Rome traces, the modal contact time is 30 s in every time window, whereas the mean contact time is slightly different, e.g., around 220 s in the time window of 0600–1000 h and around 130 s in the other time windows. Based on Eq. (5), the effect of the bottleneck bandwidth is proportional to the transmission capacity. We assume the bandwidth is 1 Mbps to estimate the capacity, which can be arbitrary (e.g., 1, 6, and 24 Mbps) and measurable in the real-world experiments [34]. In a conservative estimate, the transmission capacity is 3.75 MB by applying the modal contact time to the equation. In more aggressive estimate, the capacity is 27.5 MB in the morning time window and 16.25 MB in the other time windows by applying the modal contact time to the equation. For example, the average file size for each type of files has been introduced to assess the downloaded contents: 0.5 MB for an image file, 0.75 MB for a text file, 6.3 MB for an audio file, and 15 MB for a video file [35]. If file size is not greater than the estimated transmission capacity, the whole content is most likely to be disseminated completely. In general, Eq. (5) can obtain the contact time for any time window.



(a) San Francisco on Wed. May 21. (b) Rome on Wed. February 5.

Fig. 17. Samples of CDF of aggregate contact times in a day.

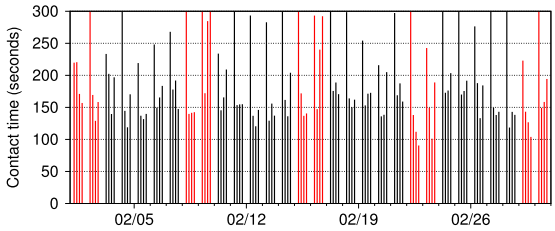


Fig. 18. Observation result of contact time through the entire trace duration from Feb 1 to March 2 in Rome traces, where the *x*-axis represents date and time.

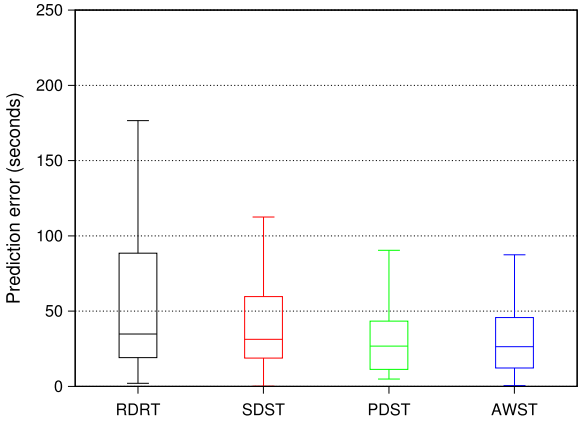


Fig. 19. Validation results of predicting contact time in Rome traces, where the *x*-axis represents predicting methods.

In Rome traces, an observation result of contact time through the entire trace duration is shown in Fig. 18, where contact time is obtained by the aggressive estimate. The *x*-axis shows date and time while the *y*-axis shows the contact time. For the entire duration, the average contact time is 196 s with standard deviation of 72 s. A peak time window with the longest contact time is between 0600 and 1000 h everyday because most taxicabs are perhaps idle at parking spaces. The peak time window is repeated while the periodic behavior of contact time is slightly different between weekdays and weekends.

Fig. 19 shows the validation result of predicting contact time in Rome traces. The result is the average of 40 independent simulation runs, shown with 95% confidence intervals. PDST outperforms the other prediction methods and its error value is 30 s. That means around 85% accuracy for the average contact time in the above observation. The predicting method achieves around 50% reduction of prediction error range, compared to RDRT. In a conservative estimate, there is almost no prediction error because modal contact time is 30 s every day for each time window.

Low prediction errors in our experiments indicate that the proposed method is accurate. Furthermore, the analytical framework has been used to estimate content distribution for taxi cab traces. Each trace covers a city extensively, depicts movement of people, exhibits varying mobility patterns over each day and across nearly 30 days.

7. Conclusion

Opportunistic content sharing among mobile users is expected to impact a number of emerging mobile applications in the near future. In this paper, Spatiotemporal reachability graphs are developed to depict spatiotemporal dissemination in opportunistic environments. We provide an analytical framework to estimate content distribution parameters with a goal to identify favorable characteristics of typical opportunistic environments for effective content dissemination. The validity of the analytical framework is investigated for known empirical datasets. The proposed framework would be useful for content sharing in any mobility scenario and enable effective dissemination in opportunistic environments.

Future work will explore other types of queries and develop necessary analytical models. Another direction of work is to implement applications of opportunistic content sharing on smartphones and facilitate interaction with the analytical framework. Key insights on traffic offloading can be explored through the real-world implementation. It is planned to steer this research in two directions to further enhance the potential of opportunistic environments—distributed computing potential and service composition.

References

[1] M. Conti, M. Kumar, Opportunities in opportunistic computing, vol. 43, no. 1, Jan 2010, pp. 42–50. http://dx.doi.org/10.1109/MC.2010.19 ISSN: 0018-9162.

[2] M.J.F. Alenazi, Y. Cheng, D. Zhang, J.P.G. Sterbenz, Epidemic routing protocol implementation in Ns-3, in: ACM Proc. Workshop on Ns-3, 2015, pp. 83–90. http://dx.doi.org/10.1145/2756509.2756523 ISBN: 978-1-4503-3375-7 http://doi.acm.org/10.1145/2756509.2756523.

[3] B. Han, A. Srinivasan, Your Friends Have More Friends Than You Do: Identifying Influential Mobile Users Through Random Walks, in: Proc. ACM Int. Symp. on MobiHoc, 2012, pp. 5–14. http://dx.doi.org/10.1145/2248371.2248376, ISBN: 978-1-4503-1281-3 http://doi.acm.org/10.1145/2248371.2248376.

[4] P. Hui, J. Crowcroft, E. Yoneki, BUBBLE Rap: Social-based forwarding in delay-tolerant networks, vol. 10, no. 11, Nov 2011, pp. 1576–1589. http://dx.doi.org/10.1109/TMC.2010.246, ISSN: 1536-1233.

[5] R. Pathak, P. Hu, J. Indulska, M. Portmann, Protocol for efficient opportunistic communication, in: The 38th IEEE Conference on Local Computer Networks, Sydney, Australia, October 2013.

[6] W. Rao, K. Zhao, Y. Zhang, P. Hui, S. Tarkoma, Towards maximizing timely content delivery in delay tolerant networks, vol. 14, no. 4, April 2015, pp. 755–769. http://dx.doi.org/10.1109/TMC.2014.2330296, ISSN: 1536-1233.

[7] M.C. González, C.A. Hidalgo, A.-L. Barabási, Understanding individual human mobility patterns, Nature 453 (7196) (2008) 779–782.

[8] C. Song, Z. Qu, N. Blumm, A.-L. Barabási, Limits of predictability in human mobility, Science 327 (5968) (2010) 1018–1021.

[9] E. Hyytiä, J. Virtamo, P. Lassila, J. Kangasharju, J. Ott, When does content float? Characterizing availability of anchored information in opportunistic content sharing, in: Proc. IEEE INFOCOM, April 2011, pp. 3137–3145. http://dx.doi.org/10.1109/INFCOM.2011.5935160 ISSN: 0743-166X.

[10] S. Wang, X. Wang, X. Cheng, J. Huang, R. Bie, The tempo-spatial information dissemination properties of mobile opportunistic networks with Levy mobility, in: Proc. IEEE ICDCS, June 2014, pp. 124–133. http://dx.doi.org/10.1109/ICDCS.2014.21, ISSN: 1063-6927.

[11] J. Whitbeck, M. Dias de Amorim, V. Conan, J.-L. Guillaume, Temporal reachability graphs, in: Proc. ACM MobiCom, 2012, pp. 377–388, http://dx.doi.org/10.1145/2348543.2348589, ISBN: 978-1-4503-1159-5 , http://doi.acm.org/10.1145/2348543.2348589.

[12] J. Tang, M. Musolesi, C. Mascolo, V. Latora, Characterising temporal distance and reachability in mobile and online social networks, ACM SIGCOMM Comput. Commun. Rev. 40 (1) (2010) 118–124.

[13] A.-K. Pietilänen, C. Diot, Dissemination in opportunistic social networks: The role of temporal communities, in: Proc. ACM Int. Symp. on MobiHoc, 2012, pp. 165–174. http://dx.doi.org/10.1145/2248371.2248396, ISBN: 978-1-4503-1281-3 http://doi.acm.org/10.1145/2248371.2248396.

[14] Y. Kim, K. Lee, N. Shroff, I. Rhee, Providing probabilistic guarantees on the time of information spread in opportunistic networks, in: Proc. IEEE INFOCOM, April 2013, pp. 2067–2075. http://dx.doi.org/10.1109/INFCOM.2013.6567008, ISSN: 0743-166X.

[15] X. Zhu, Y. Li, D. Jin, P. Hui, Temporal capacity graphs for time-varying mobile networks, in: Proc. Int. Conf. World Wide Web (WWW) Workshop on Simplifying Complex Networks for Practitioners (SIMPLEX), 2014, pp. 723–726. http://dx.doi.org/10.1145/2567948.2579362, ISBN: 978-1-4503-2745-9 http://dx.doi.org/10.1145/2567948.2579362.

[16] J. Qin, H. Zhu, Y. Zhu, L. Lu, G. Xue, M. Li, POST: Exploiting dynamic sociality for mobile advertising in vehicular networks, in: Proc. IEEE INFOCOM, April 2014, pp. 1761–1769. http://dx.doi.org/10.1109/INFOCOM.2014.6848114.

[17] W. Gao, G. Cao, User-centric data dissemination in disruption tolerant networks, in: Proc. IEEE INFOCOM, April 2011, pp. 3119–3127. http://dx.doi.org/10.1109/INFCOM.2011.5935157 ISSN: 0743-166X.

[18] K.-J. Lin, C.-W. Chen, C.-F. Chou, Preference-aware content dissemination in opportunistic mobile social networks, in: Proc. IEEE INFOCOM, March 2012, pp. 1960–1968. http://dx.doi.org/10.1109/INFCOM.2012.6195573, ISSN: 0743-166X.

[19] L. Guo, C. Zhang, H. Yue, Y. Fang, A privacy-preserving social-assisted mobile content dissemination scheme in DTNs, in: Proc. IEEE INFOCOM, April 2013, pp. 2301–2309. http://dx.doi.org/10.1109/INFCOM.2013.6567034, ISSN: 0743-166X.

[20] T. Ning, Z. Yang, H. Wu, Z. Han, Self-Interest-Driven incentives for ad dissemination in autonomous mobile social networks, in: Proc. IEEE INFOCOM, April 2013, pp. 2310–2318, http://dx.doi.org/10.1109/INFCOM.2013.6567035, ISSN: 0743-166X.

[21] P. Sermpezis, T. Spyropoulos, Not all content is created equal: Effect of popularity and availability for content-centric opportunistic networking, in: Proc. ACM Int. Symp. on MobiHoc, 2014, pp. 103–112. http://dx.doi.org/10.1145/2632951.2632976, ISBN: 978-1-4503-2620-9, http://doi.acm.org/10.1145/2632951.2632976.

[22] J. Ahn, M. Sathiamoorthy, B. Krishnamachari, F. Bai, L. Zhang, Optimizing content dissemination in vehicular networks with radio heterogeneity, vol. 13, no. 6, June 2014, pp. 1312–1325. http://dx.doi.org/10.1109/TMC.2013.100, ISSN: 1536-1233.

[23] N. Belblidia, M. Sammarco, L. Costa, M. Dias de Amorim, EPICS: Fair opportunistic multi-content dissemination, vol. 14, no. 9, Sept 2015, pp. 1847–1860. http://dx.doi.org/10.1109/TMC.2014.2368558, ISSN: 1536-1233.

[24] A. Casteigts, P. Flocchini, B. Mans, N. Santoro, Measuring temporal lags in delay-tolerant networks, vol. 63, no. 2, Feb 2014, pp. 397–410. http://dx.doi.org/10.1109/TC.2012.208, ISSN: 0018-9340.

[25] B.-M. Bui-Xuan, A. Ferreira, A. Jarry, Computing shortest, fastest, and foremost journeys in dynamic networks, Internat. J. Found Comput. Sci. 14 (2) (2003) 267–285.
[26] T. Spyropoulos, K. Psounis, C.S. Raghavendra, Spray and Wait: An Efficient Routing Scheme for Intermittently Connected Mobile Networks, in: Proc. ACM SIGCOMM Workshop on Delay-Tolerant Networking, WDTN'05, ACM, New York, NY, USA, 2005, pp. 252–259 ser..
[27] A. Lindgren, A. Doria, O. Schelén, Probabilistic routing in intermittently connected networks, SIGMOBILE Mob. Comput. Commun. Rev. 7 (3) (2003) 19–20.
[28] D. Kasamatsu, M. Kumar, P. Hu, Service compositions in challenged mobile environments under spatiotemporal constraints, in: SmartComp, Hong Kong, China, 2017.
[29] A. Chaintreau, P. Hui, J. Crowcroft, C. Diot, R. Gass, J. Scott, Impact of human mobility on opportunistic forwarding algorithms, vol. 6, no. 6, June 2007, pp. 606–620. http://dx.doi.org/10.1109/TMC.2007.1060, ISSN: 1536-1233.
[30] T. Karagiannis, J.-Y. Le Boudec, M. Vojnović, Power law and exponential decay of intercontact times between mobile devices, vol. 9, no. 10, Oct 2010, pp. 1377–1390. http://dx.doi.org/10.1109/TMC.2010.99, ISSN: 1536-1233.
[31] M. Piorkowski, N. Sarafijanovic-Djukic, M. Grossglauser, A parsimonious model of mobile partitioned networks with clustering, in: Int. Communication Systems and Networks and Workshops, Jan 2009, pp. 1–10. http://dx.doi.org/10.1109/COMSNETS.2009.4808865, ISSN: 2155-2487.
[32] M. Bonola, L. Bracciale, P. Loreti, R. Amici, A. Rabuffi, G. Bianchi, Opportunistic communication in smart city: Experimental insight with small-scale taxi fleets as data carriers, Ad Hoc Networks 43 (2016) 43–55.
[33] A. Benslimane, R. Amici, M. Bonola, L. Bracciale, A. Rabuffi, P. Loreti, G. Bianchi, Performance assessment of an epidemic protocol in VANET using real traces, in: Proc. Int. Conf. on Selected Topics in Mobile and Wireless Networking (MoWNet), in: ser. Procedia Computer Science, vol. 40, 2014, pp. 92–99.
[34] K. Sankaran, A.L. Ananda, M.C. Chan, L.-S. Peh, Dynamic framework for building highly-localized mobile web dtn applications, in: Proc. ACM MobiCom Workshop on Challenged Networks, CHANTS, 2014, pp. 43–48. http://dx.doi.org/10.1145/2645672.2645675, ISBN: 978-1-4503-3071-8, http://doi.acm.org/10.1145/2645672.2645675.
[35] S. Qayyum, M. Shahriar, M. Kumar, S. Das, PCV: Predicting contact volume for reliable and efficient data transfers in opportunistic networks, in: Proc. IEEE Conf. Local Computer Networks, LCN, Oct 2013, pp. 801–809, http://dx.doi.org/10.1109/LCN.2013.6761335, ISSN: 0742-1303.