IET Networks

Research Article

Contact duration aware cache refreshing for mobile opportunistic networks

ISSN 2047-4954
Received on 22nd September 2015
Revised on 24th May 2016
Accepted on 13th June 2016
doi: 10.1049/iet-net.2015.0086
www.ietdl.org

Fang Wang[1], *Zhaocheng Wang*[1], *Zhixing Yang*[1], *Sheng Chen*[2,3] ✉

[1]Department of Electronic Engineering, Tsinghua National Laboratory for Information Science and Technology, Tsinghua University, Beijing 100084, People's Republic of China
[2]Electronics and Computer Science, University of Southampton, Southampton SO17 1BJ, UK
[3]Faculty of Engineering, King Abdulaziz University, Jeddah 21589, Saudi Arabia
✉ E-mail: sqc@ecs.soton.ac.uk

Abstract: Caching is a practical data access approach in mobile opportunistic networks, which enables mobile terminals to obtain data from adjacent nodes and to reduces accessing latency, whereby the updating files should be delivered to the caching nodes on time when the source is refreshed in order to maintain the freshness of the cached data. Due to the node mobility and short communication range, the contact duration in mobile opportunistic networks is short and this limits the amount of data transmitted in one contact. However, current cache refreshing schemes often ignore this critical issue. In this study, the authors study the contact duration aware cache refreshing in mobile opportunistic networks. Specifically, they establish a two-state Markov chain to describe the data delivered from one node to another. Based on the proposed model, they investigate a tree structure to refresh the cached data, where each node is responsible to refresh the data cached at its children and it may opportunistically rely on some relays outside the tree to improve the refreshing efficiency. Extensive trace-driven simulations validate the effectiveness of the authors' scheme in terms of improving cache refreshing in challenging mobile environments.

Nomenclature

t_S	last time when the source data was refreshed
t_B	last time the data cached at the parent node was refreshed
t_E	last time the data cached at the child node was refreshed
t_{R1}	time the parent node selected the relay R
t_{R2}	time the updating file was delivered to the relay R
λ_l	link perishing rate
μ_l	link generating rate
γ	source refreshing rate
$f(t)$	probability density function (PDF) of source refreshing interval
L	data lifetime
θ	cumulative contact duration threshold

1 Introduction

In order to provide communication services in the challenging wireless networks where there exist no direct end-to-end paths between the source and destination pairs, mobile opportunistic networks are conceived, which exploit the opportunistic connectivity and node mobility to relay and carry messages. When the next hop is not immediately available for the current node to deliver a message, it will store the message in its buffer, carry the message along its movement, and forward the data to other nodes when a communication opportunity is available. Since this store-carry-and-forward mechanism can transmit data even in the challenging environment without end-to-end path between the source and destination, many related interesting applications are experimented, including mobile social networks based on human mobility, sensor networks for wildlife tracking and habitat monitoring, vehicular *ad-hoc* networks for road safety and commercial applications, and deep-space interplanetary networks. Data access and transmission is critically important and challenging in all these applications.

There are various approaches to provide data access in mobile opportunistic networks, including epidemic routing [1] as well as relay and community assisted data access [2–6]. Epidemic routing is a robust but inefficient way to disseminate data to mobile nodes [1]. In relay based algorithms, some nodes are selected as relay nodes according to their nodes' interest and mobility patterns to deliver data from the source to requesters [2, 3]. In the so-called publish/subscribe systems, the network is divided into communities where one or several brokers are elected to help propagating and collecting data as well as to connect other communities [4–6].

Due to the intermittent network connectivity, caching is a practical approach in mobile opportunistic networks, wherein data is cached by mobile nodes according to the query history, so that future queries can be rapidly satisfied. Since the source may be refreshed periodically, it is essential to keep the consistency of these data copies. Specifically, when the source is refreshed, the data cached at other nodes is out-of-date and should be refreshed as soon as possible, so that it can provide the latest information. Therefore, it is imperative to maintain the cache freshness.

Current works address the problem of cache refreshing by modelling the contact process as a Poisson distribution [7]. However, the existing works ignore the limitation of contact duration and simply assume that the contact duration is sufficiently long to transmit the whole content. In mobile opportunistic networks, contact duration is usually limited due to the mobility and short transmission range of wireless terminals. For example, when users communicate with each other via hand-held devices through the Bluetooth, whose communication range is usually within 10 m, the contact duration tends to be only several seconds if the devices are moving at a normal walking speed. An even more extreme example of short contact duration is the communication between peer-to-peer moving vehicles, where users are in high speed vehicles. Even if they communicate via WiFi, which has a relatively long transmission range, the contact duration is still very short. Unfortunately, a plurality of multimedia contents needs a relatively long time to complete the transmission, which indicates that more than one contact is needed. Therefore, contact duration limitation should be considered in data transmission models for cache refreshing in mobile opportunistic networks.

In this paper, we study contact duration aware cache refreshing. In particular, we establish a more realistic data transmission model assuming that the complete data can be delivered from one node to another via one or several intermittent contacts as long as the cumulative contact duration is sufficiently long. Furthermore, we investigate a hierarchical cache refreshing scheme, where nodes interested in the same data are organised as a tree structure during data access. Each node in the tree is responsible for refreshing the data cached at its children. To improve the refreshing efficiency, the nodes opportunistically replicate the updating files to the nodes outside the tree, referred to as the relays, to help forward the data. Our contributions are as follows:

- We consider the limited contact duration in cache refreshing by modelling data transmission as a two-state Markov chain and describing the distribution of cumulative contact duration.
- We analyse the utility of updating files for refreshing the cached data based on the proposed model. These utilities are then used to opportunistically replicate updating files and to satisfy the requirements of cache freshness.
- We use extensive real trace-driven simulations to illustrate the good performance of our contact duration aware cache refreshing scheme in terms of improving the refreshing ratio and reducing the refreshing overhead.

2 Related work

Caching enables clients to obtain their requested data from nodes nearby rather than directly from the data source via multiple hops, which greatly saves bandwidth and reduces accessing latency. Research efforts have focused on caching policy design and cache consistency maintenance. In the former, clients' interest, buffer capacity and node mobility patterns have been exploited to determine the number of replicated copies and the way of caching [8, 9]. The latter deals with how to maintain the versions of the cached data to be consistent with the source data in order to provide up-to-date information. There exist many works investigating the maintenance of cache consistency [10–14]. However, these end-to-end based cache consistency strategies are originated from the designs for wired networks, and they are impractical for wireless environments with frequent link disconnection [15]. The authors of [7] studied how to maintain the freshness of the cached data in mobile opportunistic networks, where the data source proactively disseminates updating files to the caching nodes through a hierarchical structure. However, they simply assumed that the contact duration is sufficiently long so that an updating file can be transmitted from one node to another in one contact. In this paper, we consider the impact of contact duration on data delivery in cache refreshing and propose a practical data transmission model for this application.

The limited contact duration is an important factor in mobile opportunistic networks. Some researchers have studied the contact duration patterns in human mobility environments [16, 17] and vehicular networks [18, 19]. Others employed a Markovian model to evaluate the effect of contact duration and message size on the performance of data dissemination, especially in terms of delivery delay and delivery ratio [20, 21]. Based on the uncovered contact duration patterns, the authors of [22, 23] proposed some contact duration aware data caching schemes, where data is encoded into packets and the number of packets that can be transmitted are proportional to the contact duration. Unlike these existing works, to the best of our knowledge, we are the first to study contact duration limitation in cache refreshing and to propose a realistic contact duration aware cache refreshing scheme.

3 Overview of data access tree (DAT)-based cache refreshing

A hierarchical tree structure is typically adopted for cache refreshing. Queries of nodes are satisfied when they contact nodes carrying their interested data, which forms a DAT [7]. In the DAT, the parent node is defined as the first node from which the child nodes receive their desired content. Whenever the source data is refreshed, updating information is disseminated through the DAT. An updating file contains the difference between the current and previous versions of the data concerned. Using delta encoding [24], cached data can be refreshed by such an updating file instead of replacing the completed data. Since the size of the updating file is usually much smaller than the complete file and it is exactly what the caching node needs, delivering the updating file, rather than the whole data, will save wireless bandwidth and reduce transmission interference with other mobile devices. Therefore, it is preferred to deliver updating files for meeting the cache freshness requirements. Each node in the DAT has knowledge of the data versions at its child nodes and is able to prepare updating files accordingly.

Denote by d_i is the version i of the data concerned and $v_{i,j}$ is the updating file required to refresh the data from version i to version j. Fig. 1*a* illustrates a DAT, where node S is the source and nodes A to E are mobile terminals. Two nodes connected by a solid-line arrow are a parent node and its child node, respectively. The data cached at nodes A to E are d_3, d_4, d_1, d_2 and d_3, respectively. A generates updating file $v_{1,3}$ to refresh d_1 at C, while B generates updating files $v_{2,4}$ and $v_{3,4}$ to refresh d_2 at D and d_3 at E, respectively. Transmission of updating files or cache refreshing occurs when the next time the corresponding nodes come into contact.

Each cached data is associated with a finite lifetime, which may be reset each time the data is refreshed. Once its lifetime expires, the data is removed from the caching node and this caching node leaves the DAT. Later, this node will re-connect to the DAT and get refreshed when it contacts another node in the DAT. Fig. 1*b* illustrates this scenario of DAT evolution. At the time instant of its caching data lifetime expired, node D has not received the updating file and leaves the DAT. When it comes in contact with node A, node D gets data d_3 from A and sets A as its new parent node.

It is noted that the leaf nodes in the tree are changeable and the parent–child relations between nodes are dynamic. Therefore, a child node may become a parent node, while a parent node may become a child node. Spatial and temporal dynamics of a DAT can be explained using the example shown in Figs. 1*a* and *c*–*f*. Denote t_A and t_C as the last times that the cached data are refreshed in nodes A and C, respectively, where $t_A < t_C$. Further denote L as data lifetime. Fig. 1*a* shows the original structure of the DAT. As shown in Fig. 1*c*, at time $t_A + L$, node A leaves the DAT since its cached data has not been refreshed by then. After this, node C cannot receive any updating file from node A and its cached data is removed at time $t_C + L$, as shown in Fig. 1*d*. The parent–child relation between nodes A and C is thus broken, and both A and C are now outside the DAT. Fig. 1*e* illustrates that at time $t'_C(> t_C + L)$, node C contacts source S and gets data d_4, while node A has not made any contact with any node in the DAT and, therefore, remains outside the DAT. The next time node A meets node C, it re-connects to the DAT and gets its desired data from node C, as depicted in Fig. 1*f*, where node C becomes the parent node and node A becomes the child node. Therefore, the DAT is evolving in space and time, and the parent–child relation is dynamic. Over a long period of time, the data access and refreshing process may be considered as a complete dynamic graph.

Due to intermittent connectivity, it is very likely that a node is unable to refresh its children's cached data within their lifetimes. To ensure the freshness of cached data, a node opportunistically replicates the updating files to other nodes outside the DAT, which act as relays or helpers to deliver the files. For example, as shown in Fig. 2*a*, node A caches data d_3 and knows the data cached at its child node C is d_1. To increase the probability that d_1 is refreshed

IET Netw., 2016, Vol. 5, Iss. 4, pp. 93–103

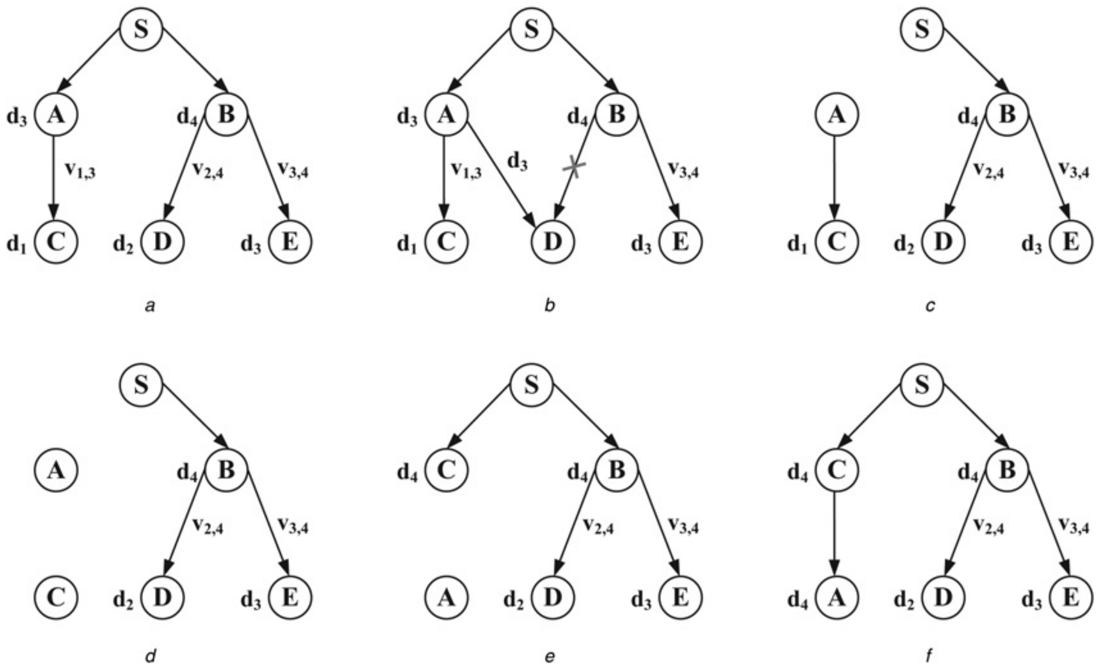

Fig. 1 *Illustration of*

a DAT
b–f DAT evolution
Specifically, (c) is associated with time instant $t_A + L$, (d) with time instant $t_C + L$ where $t_C > t_A$, (e) with time instant t'_C where $t'_C > t_C + L$, and (f) with time instant t'_A where $t'_A > t'_C$

before the lifetime expiration, A selects relays R_1 and R_2, and replicates $v_{1,3}$ to them to forward the file to C. Note that the selected relays are only able to refresh the specific data version cached at the specific node, but unable to refresh data cached at other nodes in the DAT or provide data access to nodes outside the DAT. How to select relays will be discussed in the following section.

When the data has been refreshed multiple times, there may be various updating files for the same data coexisting in the network. We assume that an updating file cannot be merged from consecutive messages, i.e. $v_{i,j} \oplus v_{j,k} \neq v_{i,k}$. Moreover, d_j cannot be refreshed to d_k by $v_{i,k}$ if $j \neq i$. When a node contacts its child node whose cached data has been refreshed by relays, the node re-generates the updating file for the child node. As depicted in Fig. 2*b*, when node A has its cached data refreshed to d_4, it generates $v_{1,4}$ for C since it is unaware of the current data version at C. The next time A contacts C whose cached data has been refreshed to d_3 by R_1, they will exchange information and A will re-generate $v_{3,4}$ for C.

4 Modelling and relay selection

In this section, we first use a contact graph to describe the network and a two-state Markov chain to model data transmission in the graph, and then describe our cache refreshing scheme by explaining how the proposed model is used in selecting relays to forward updating files.

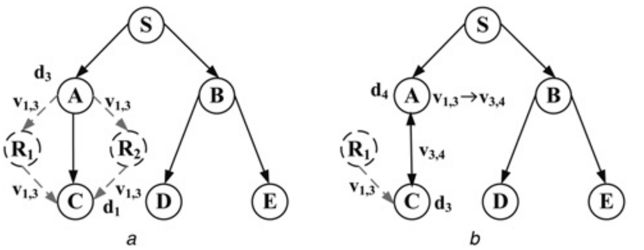

Fig. 2 *Illustration of*

a Relay selection
b Regenerating appropriate updating file

4.1 Contact graph and source refreshing

We describe the intermittent contacts among nodes by a contact graph $G(\mathcal{V}, \mathcal{E}, \xi, \eta)$, where $\mathcal{V}$ is the set of mobile nodes and $\mathcal{E}$ is the set of contact processes. Each link $l = (i, j) \in \mathcal{E}$ is associated with an inter-contact interval $\xi(l)$ [also denoted as $\xi(i, j)$] and a contact duration $\eta(l)$ [also denoted as $\eta(i, j)$]. Data transmission only occurs in link connection. We consider $\xi(l)$ and $\eta(l)$ as exponentially distributed with the intensities of μ_l (also denoted as $\mu_{i,j}$) and λ_l (also denoted as $\lambda_{i,j}$), respectively. We now justify this assumption.

Exponentially distributed inter-contact interval and contact duration are widely adopted in modelling mobile opportunistic networks [25]. More specifically, recent work [26] has found that the inter-contact interval exhibits the exponential distribution over a large range of time scales. In terms of contact duration, empirical studies [16, 27] in human mobility environments have indicated that the contact duration of human mobility follows a power law distribution, and the Pareto distribution can be used to model it. However, in an extensive empirical study involving real-world vehicular mobility traces [18], it has been validated that the contact duration of vehicular opportunistic networks obeys an exponential distribution that includes at least 80% of the whole distribution, while beyond a characteristic time point it decays as a power law one. Based on these general observations, we may justify that $\xi(l)$ and $\eta(l)$ can be modelled by the two exponential distributions with the intensities μ_l and λ_l, respectively. Clearly, there exists a consent that $\xi(l)$ is exponentially distributed, while there are different opinions regarding the true distribution of $\eta(l)$. Regardless what is the true distribution of the contact duration, however, modelling it as an exponential distribution with a certain intensity is much more realistic and more accurate than simply assuming that the contact duration is always sufficiently long for transmitting the whole content, which in fact is the assumption adopted by all the existing works on cache refreshing, e.g. [7].

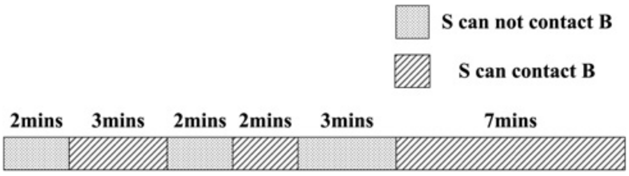

Fig. 3 *Intermittent contact patterns between nodes S and B, which S utilises to deliver data to B*

We also assume that the inter-refreshing interval of the source data is exponentially distributed with average refreshing rate γ, which has been validated by the work [7]. Specifically, the authors of [7] demonstrated that the inter-refreshing interval follows an exponential distribution within a time boundary and a generalised Pareto distribution out of the boundary. The notations and their physical meanings used through this paper are summarised in the Nomenclature section.

4.2 Data transmission

In mobile opportunistic networks, nodes intermittently contact each other. When two nodes are located within the communication range, data transmission can occur but there is no guarantee that the data can be delivered in one contact. Let us assume a fixed data transmission rate. When the contact duration is too short, only part of the data can be transmitted, and the node records which part of the data has been delivered and which part has not. The undelivered part is transmitted in the next contact. As long as the cumulative contact duration (CCD) is longer than the required time duration for transmitting the whole data, the complete data can be delivered. For the example illustrated in Fig. 1*a*, assume that node *S* is delivering data to node *B*. Fig. 3 depicts the intermittent contacts and the contact duration of each contact between *S* and *B*. If it takes 10 min to transmit the data, the complete data can be delivered from *S* to *B* after three contacts. This process can be modelled as a two-state Markov chain. As shown in Fig. 4, the two states arrive alternately, where State 1 represents that two nodes are out of communication range (out of contact) while State 2 represents that they are in communication range (in contact). Thus the time duration of the process staying in State 1 is modelled by $\xi(l)$ and the time duration of the process staying in State 2 is modelled by $\eta(l)$. Denote the cumulative distribution function (CDF) of $\xi(l)$ by $P(\xi(l) \le x) = H(x)$ and the CDF of $\eta(l)$ by $P(\eta(l) \le x) = G(x)$.

Let us further denote $\xi_n(l)$ and $\eta_n(l)$ as the nth States 1 and 2 of this process, respectively. Suppose that the initial state of this process is State 1. Then $\xi_1(l),\ \eta_1(l); \xi_2(l),\ \eta_2(l); \xi_3(l),\ \eta_3(l); \cdots$ represent the state sequence of this process. Note that $\{\xi_n(l),\ n = 1, 2, 3, \ldots\}$ are independently identically distributed (i.i.d.) and $\{\eta_n(l),\ n = 1, 2, 3, \cdots\}$ are i.i.d. Let $\nu_n = \xi_1(l) + \xi_2(l) + \cdots + \xi_n(l)$ and $\rho_n = \eta_1(l) + \eta_2(l) + \cdots + \eta_n(l)$ be the sums of the first n States 1 and 2, respectively. Denote $\alpha(t)$ and $\beta(t)$ as the total amounts of the time durations in States 1 and 2, respectively, during the time interval $(0,\ t]$, where $t > 0$. Clearly, $\beta(t)$ is the CCD over the time interval $(0,\ t]$. Define $\Phi(x,\ t) = P(\beta(t) \le x)$ as the distribution of $\beta(t)$. The event $\beta(t) \le x$ occurs in the following two scenarios: (i) at the time instant t the process is in State 1 and there have been n contacts between the two nodes with $\rho_n \le x$; and (ii) at the time instant t the process is in State 2 and there have been n inter-contact intervals between the two nodes with $\nu_n \ge t - x$. Therefore, $\Phi(x,\ t)$ is expressed as

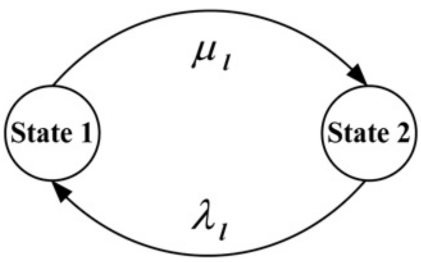

Fig. 4 *State transition process representing intermittent contact patterns between two nodes*

$$\Phi(x,\ t) = \sum_{n=0}^{\infty} P(\rho_n \le x,\ \nu_n + \rho_n \le t < \nu_{n+1} + \rho_n) + \sum_{n=1}^{\infty} P(\nu_n \ge t - x,\ \nu_n + \rho_{n-1} \le t < \nu_n + \rho_n). \quad (1)$$

According to Theorem 1 in [28], (1) can be rewritten as

$$\begin{aligned} \Phi(x,\ t) &= \sum_{n=0}^{\infty} (P(\rho_n \le x,\ \nu_n < t - x) - P(\rho_n \le x,\ \nu_{n+1} < t - x)) \\ &= \sum_{n=0}^{\infty} P(\rho_n \le x)(P(\nu_n < t - x) - P(\nu_{n+1} < t - x)) \\ &= \sum_{n=0}^{\infty} G_n(x)(H_n(t - x) - H_{n+1}(t - x)),\ 0 \le x < t, \end{aligned} \quad (2)$$

where $G_n(x) = P(\rho_n \le x)$, $G_0(x) = 1$, $H_n(x) = P(\nu_n < t - x)$, and $H_0(x) = 1$.

Furthermore, as mentioned previously, $G(\cdot)$ and $H(\cdot)$ are exponentially distributed. Hence, $G_n(\cdot)$ and $H_n(\cdot)$ are the CDFs of two Gamma distributions, and we have [28]

$$G_n(x) = 1 - \sum_{k=0}^{n-1} \frac{1}{k!} \mathrm{e}^{-\lambda_l x} (\lambda_l x)^k, \quad (3)$$

$$H_n(x) - H_{n+1}(x) = \frac{1}{n!} \mathrm{e}^{-\mu_l x} (\mu_l x)^n. \quad (4)$$

Then (2) can be expressed as follows

$$\begin{aligned} \Phi(x,\ t) &= \sum_{n=0}^{\infty} \left(1 - \sum_{k=0}^{n-1} \frac{1}{k!} (\lambda_l x)^k\right) \frac{1}{n!} \mathrm{e}^{-\mu_l (t-x)} (\mu_l (t - x))^n \\ &= 1 - \sum_{n=1}^{\infty} \sum_{k=0}^{n-1} \frac{1}{k!} \mathrm{e}^{-\lambda_l x} (\lambda_l x)^k \frac{1}{n!} \mathrm{e}^{-\mu_l (t-x)} (\mu_l (t - x))^n \\ &= 1 - \sum_{k=0}^{\infty} \frac{1}{k!} \mathrm{e}^{-\lambda_l x} (\lambda_l x)^k \sum_{n=k+1}^{\infty} \frac{1}{n!} \mathrm{e}^{-\mu_l (t-x)} (\mu_l (t - x))^n \\ &= 1 - \sum_{k=0}^{\infty} \frac{1}{k!} \mathrm{e}^{-\lambda_l x} (\lambda_l x)^k \int_0^{\mu_l (t-x)} \frac{y^k}{\Gamma(k + 1)} \mathrm{e}^{-y}\, \mathrm{d}y \\ &= 1 - \mathrm{e}^{-\lambda_l x} \int_0^{\mu_l (t-x)} \sum_{k=0}^{\infty} \frac{1}{k!} (\lambda_l x)^k \frac{1}{\Gamma(k + 1)} y^k \mathrm{e}^{-y}\, \mathrm{d}y \\ &= 1 - \mathrm{e}^{-\lambda_l x} \int_0^{\mu_l (t-x)} I_0(2\sqrt{\lambda_l x y})\, \mathrm{e}^{-y}\, \mathrm{d}y, \end{aligned} \quad (5)$$

where

$$I_0(x) = \sum_{j=0}^{\infty} \frac{1}{j!\Gamma(j + 1)} \left(\frac{x}{2}\right)^{2j}$$

is the first kind modified Bessel function of order 0, and the Gamma function $\Gamma(k + 1) = k!$ for non-negative integer k.

The above data transmission model will be referred to as the intermittent transmission model (ITM). In the following, we use the ITM to measure the updating utility and to select relays to forward updating files to the caching nodes.

IET Netw., 2016, Vol. 5, Iss. 4, pp. 93–103

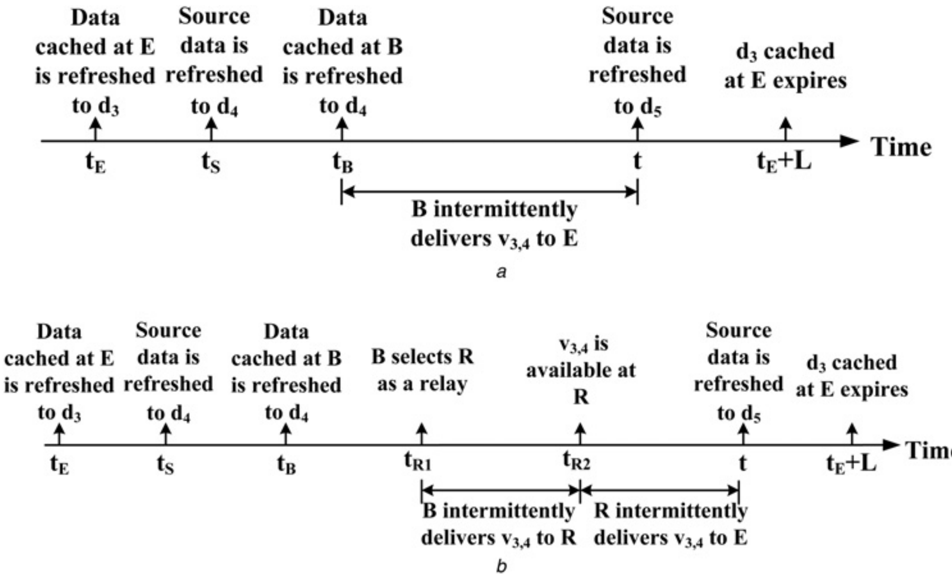

Fig. 5 *Illustration of cache refreshing using the ITM*

a Parent node refreshes child node
b Parent node selects relay to help refreshing child node

4.3 Relay selection

Generally speaking, more relays may enhance the performance of cache refreshing. However, each time a parent node selects a new relay, it replicates the updating file to the relay node, which consumes energy and wireless resource. Moreover, without careful selection, a parent node may replicate an updating file to some inappropriate relays whose average contact durations with the child node are very short and the contact frequencies are very low, resulting in very little performance improvement at a considerable cost. Therefore, it is imperative to select appropriate relay nodes to achieve as much as possible performance enhancement with as little as possible expenditure. We now detail how to select relays.

First, we introduce the utility of an updating file, denoted as U, which is defined as the probability that the cache freshness requirement is satisfied. Specifically, given an updating file, we calculate the utilities of the parent node and each node outside the DAT that the parent node contacts, to determine whether this parent node needs help to refresh its child node and which nodes should be selected as relays. If the utility of the parent node is lower than the required value U_{cr}, where $U_{cr} \in (0, 1)$, it selects relays to help forwarding the updating file. Whenever the parent node contacts another node outside the DAT, if the node's utility is larger than the maximum utility of the already selected relays, or that of the parent itself if no selected relay yet, the parent node will select this node as a new relay and replicate the updating file to it.

Denote K as the number of selected relays and U_k, $1 \le k \le K$, as the utility of relay k. Further denote the parent node as node 0 and let U_0 be its utility. Note that U_k, $0 \le k \le K$, represents the probability that node k can refresh the data cached at the child node. Then

$$\widetilde{U} = 1 - \prod_{k=0}^{K} (1 - U_k) \qquad (6)$$

represents the probability that at least one of these $K+1$ nodes can refresh the data cached at the child node. When $\widetilde{U} \ge U_{cr}$, the cache freshness requirement is satisfied and the parent node may stop selecting new relays. Data refreshing follows the ITM introduced in Section 4.2, and relays just carry and forward updating files. For updating file $v_{i,j}$, the cached data d_i is successfully refreshed when the CCD is longer than a threshold, denoted as $\theta_{i,j}$. For notational simplicity, we drop the indices and use θ in the following analysis. From (5), we can readily obtain the probability of the CCD being longer than θ, which is calculated as

$$P(\beta(t) > \theta) = 1 - \Phi(\theta, t) = e^{-\lambda_l \theta} \int_0^{\mu_l (t-\theta)} I_0(2\sqrt{\lambda_l \theta y})\, e^{-y}\, dy. \qquad (7)$$

Consider the example of Fig. 1*a* where node B is refreshing the data cached at node E from d_3 to d_4, i.e. node B is a parent node. The process is illustrated in Fig. 5*a* to show how to calculate U. The utility measures the probability at time t that B has refreshed E's cached data within its lifetime, given that the source data is refreshed to d_5 at time t. Using (7) with $\lambda_{B,E}$ and $\mu_{B,E}$ for the process of B refreshing E, the utility of parent node B, denoted by U_B, can be expressed by (see (8))

where $T = t_E + L$ and $t_0 = t_B + \theta$.

$$\begin{aligned} U_B &= \int_{t_B+\theta}^{\infty} P(B \text{ refreshes } E \text{ within the data's lifetime}) P(S \text{ is refreshed at time } t)\, dt \\ &= \int_{t_B+\theta}^{t_E+L} P(\beta(t-t_B) > \theta) f(t-t_S)\, dt + P(\beta(t_E+L-t_B) > \theta) \int_{t_E+L}^{\infty} f(t-t_S)\, dt \\ &= \int_{t_0}^{T} P(\beta(t-t_B) > \theta) f(t-t_S)\, dt + P(\beta(T-t_B) > \theta) \int_{T}^{\infty} f(t-t_S)\, dt \\ &= \int_{t_0}^{T} \int_0^{\mu_{B,E}(t-t_0)} \gamma e^{-(\lambda_{B,E}\theta + y + \gamma(t-t_S))} I_0\left(2\sqrt{\lambda_{B,E}\theta y}\right) dy\, dt + e^{-(\lambda_{B,E}\theta + \gamma(T-t_S))} \int_0^{\mu_{B,E}(T-t_0)} I_0\left(2\sqrt{\lambda_{B,E}\theta y}\right) e^{-y}\, dy, \end{aligned} \qquad (8)$$

Next, assume that the parent node B meets a potential relay node R and is deciding whether to select R as a relay node. The calculation of R's utility is more complicated and the process is illustrated in Fig. 5*b*, where t_{R1} is the time instance that B selects R as a relay, while t_{R2} denotes the time instance that B has successfully delivered updating file $v_{3,4}$ to R. Then the probability density function (PDF) of the process that the updating file is available at R at time instance t_{R2} can be expressed as (see (9))

where $t_R = t_{R2} - t_{R1}$ and

$$I_1(x) = \sum_{j=0}^{\infty} \frac{1}{j!\Gamma(j+2)} \left(\frac{x}{2}\right)^{2j+1}$$

is the first kind modified Bessel function of order 1. The utility of R, denoted as U_R, measures the probability that R has refreshed E's cached data within its lifetime, under the conditions that the updating file is available at R at t_{R2} and the source data is refreshed to d_5 at time t. The three events that B delivers the updating file to R, R refreshes E's cached data and the source data is refreshed are independent with each other. Using (7) with the parameters $\lambda_{R,E}$ and $\mu_{R,E}$ for the process of R refreshing E as well as the PDF (9), we can express U_R as follows (see (10))
Note that U_R explicitly depends on $\lambda_{R,E}$ and $\mu_{R,E}$ via $P(\beta(t - t_{R2}) > \theta)$ of (7).

Remarks: It can be seen that when the parent node B is calculating the utility of refreshing the child node E by itself or by using relay R to refresh E, it requires the values of the parameters t_B, t_E, γ, t_S, L, θ, or t_{R1}, $\lambda_{B,E}$ or $\lambda_{B,R}$ and $\lambda_{R,E}$ as well as $\mu_{B,E}$ or $\mu_{B,R}$ and $\mu_{R,E}$. Note that t_E, L and θ are the local information stored in the parent node, while t_B is the time when the updating file is available and is known to the parent node too. Similarly, the parent node B knows t_{R1} as this is the time instant that it selects the relay R. The source data information, γ and t_S, are spread over the network and, therefore, it is reasonable to assume that the parent node has these parameters. The statistics $\lambda_{B,E}$ and $\mu_{B,E}$ may be obtained from the contact history between the parent node B and the child node E. In the same way, the statistics $\lambda_{B,R}$ and $\mu_{B,R}$ may be obtained. Similarly, the parameters $\lambda_{R,E}$ and $\mu_{R,E}$ are available at the potential relay R. As regarding how this information ($\lambda_{R,E}$ and $\mu_{R,E}$) can be obtained by the parent node B, we assume that nodes are willing to cooperate and, therefore, when the parent node B contacts a potential relay node R, R will pass $\lambda_{R,E}$ and $\mu_{R,E}$ to the parent node.

In practice, nodes are identified by their unique identity (ID) numbers, which can be their MAC addresses or any other specific numbers. Thus, each node can create a table to record its contact history with others, where each item records its intermittent contacts with another node. As revealed in [29], the movement of a mobile node often follows a certain pattern. Although a node may encounter many other nodes in its daily activity, the number of the nodes that it frequently contacts is not very large. If the

$$\begin{aligned}
f_\Phi(\theta, t_R) &= \frac{\partial \Phi(\theta, t_R)}{\partial \theta} = \frac{\partial}{\partial \theta}\left(1 - \sum_{n=1}^{\infty}\sum_{k=0}^{n-1} \frac{1}{k!} e^{-\lambda_{B,R}\theta} (\lambda_{B,R}\theta)^k \frac{1}{n!} e^{-\mu_{B,R}(t_R-\theta)} (\mu_{B,R}(t_R-\theta))^n\right) \\
&= (\lambda_{B,R} - \mu_{B,R}) \sum_{n=1}^{\infty}\sum_{k=0}^{n-1} \frac{(\lambda_{B,R}\theta)^k}{k!} e^{-\lambda_{B,R}\theta} \frac{(\mu_{B,R}(t_R-\theta))^n}{n!} e^{-\mu_{B,R}(t_R-\theta)} - \lambda_{B,R} \sum_{n=2}^{\infty}\sum_{k=0}^{n-2} \frac{(\lambda_{B,R}\theta)^k}{k!} e^{-\lambda_{B,R}\theta} \frac{(\mu_{B,R}(t_R-\theta))^n}{n!} e^{-\mu_{B,R}(t_R-\theta)} \\
&\quad + \mu_{B,R} \sum_{n=0}^{\infty}\sum_{k=0}^{n} \frac{(\lambda_{B,R}\theta)^k}{k!} e^{-\lambda_{B,R}\theta} \frac{(\mu_{B,R}(t_R-\theta))^n}{n!} e^{-\mu_{B,R}(t_R-\theta)} \\
&= (\lambda_{B,R} - \mu_{B,R}) \sum_{k=0}^{\infty} \frac{(\lambda_{B,R}\theta)^k}{k!} e^{-\lambda_{B,R}\theta} \sum_{n=k+1}^{\infty} \frac{(\mu_{B,R}(t_R-\theta))^n}{n!} e^{-\mu_{B,R}(t_R-\theta)} - \lambda_{B,R} \sum_{k=0}^{\infty} \frac{(\lambda_{B,R}\theta)^k}{k!} e^{-\lambda_{B,R}\theta} \sum_{n=k+2}^{\infty} \frac{(\mu_{B,R}(t_R-\theta))^n}{n!} e^{-\mu_{B,R}(t_R-\theta)} \\
&\quad + \mu_{B,R} e^{-\lambda_{B,R}\theta} + \mu_{B,R} \sum_{k=1}^{\infty} \frac{(\lambda_{B,R}\theta)^k}{k!} e^{-\lambda_{B,R}\theta} \sum_{n=k}^{\infty} \frac{(\mu_{B,R}(t_R-\theta))^n}{n!} e^{-\mu_{B,R}(t_R-\theta)} \\
&= (\lambda_{B,R} - \mu_{B,R}) \sum_{k=0}^{\infty} \frac{(\lambda_{B,R}\theta)^k}{k!} e^{-\lambda_{B,R}\theta} \int_0^{\mu_{B,R}(t_R-\theta)} \frac{y^k}{\Gamma(k+1)} e^{-y}\, dy - \lambda_{B,R} \sum_{k=0}^{\infty} \frac{(\lambda_{B,R}\theta)^k}{k!} e^{-\lambda_{B,R}\theta} \int_0^{\mu_{B,R}(t_R-\theta)} \frac{y^{k+1}}{\Gamma(k+2)} e^{-y}\, dy \\
&\quad + \mu_{B,R} e^{-\lambda_{B,R}\theta} + \mu_{B,R} \sum_{k=1}^{\infty} \frac{(\lambda_{B,R}\theta)^k}{k!} e^{-\lambda_{B,R}\theta} \int_0^{\mu_{B,R}(t_R-\theta)} \frac{y^{k-1}}{\Gamma(k)} e^{-y}\, dy \\
&= (\lambda_{B,R} - \mu_{B,R}) e^{-\lambda_{B,R}\theta} \int_0^{\mu_{B,R}(t_R-\theta)} I_0(2\sqrt{\lambda_{B,R}\theta y}) e^{-y}\, dy - \lambda_{B,R} e^{-\lambda_{B,R}\theta} \int_0^{\mu_{B,R}(t_R-\theta)} I_1(2\sqrt{\lambda_{B,R}\theta y}) e^{-y} \frac{\sqrt{y}}{\sqrt{\lambda_{B,R}\theta}}\, dy \\
&\quad + \mu_{B,R} e^{-\lambda_{B,R}\theta} + \mu_{B,R} e^{-\lambda_{B,R}\theta} \int_0^{\mu_{B,R}(t_R-\theta)} I_1(2\sqrt{\lambda_{B,R}\theta y}) e^{-y} \frac{\sqrt{\lambda_{B,R}\theta}}{\sqrt{y}}\, dy \\
&= e^{-\lambda_{B,R}\theta} \int_0^{\mu_{B,R}(t_R-\theta)} \left((\lambda_{B,R} - \mu_{B,R}) I_0(2\sqrt{\lambda_{B,R}\theta y}) + \frac{\lambda_{B,R}\mu_{B,R}\theta - \lambda_{B,R} y}{\sqrt{\lambda_{B,R}\theta y}} I_1(2\sqrt{\lambda_{B,R}\theta y})\right) e^{-y}\, dy + \mu_{B,R} e^{-\lambda_{B,R}\theta},
\end{aligned} \tag{9}$$

$$\begin{aligned}
U_R &= \int_{t_{R1}+2\theta}^{\infty} P(R\text{refreshes } E \text{ within data's lifetime}|\text{updating file is available at } R \text{ at } t_{R2}, S \text{ is refreshed at time } t)\, dt \\
&= \int_{t_{R1}+2\theta}^{T} \int_{t_{R1}+\theta}^{t-\theta} P(\beta(t - t_{R2}) > \theta) f_\Phi(\theta, t_R) f(t - t_S)\, dt_{R2}\, dt + \int_{t_{R1}+\theta}^{T-\theta} P(\beta(T - t_{R2}) > \theta) f_\Phi(\theta, t_R)\, dt_{R2} \int_T^{\infty} f(t - t_S)\, dt \\
&= \int_{t_{R1}+2\theta}^{T} \int_{t_{R1}+\theta}^{t-\theta} \gamma e^{-\gamma(t-t_S)} P(\beta(t - t_{R2}) > \theta) f_\Phi(\theta, t_R)\, dt_{R2}\, dt + e^{-\gamma(T-t_S)} \int_{t_{R1}+\theta}^{T-\theta} P(\beta(T - t_{R2}) > \theta) f_\Phi(\theta, t_R)\, dt_{R2}.
\end{aligned} \tag{10}$$

IET Netw., 2016, Vol. 5, Iss. 4, pp. 93–103

contact frequency or the average contact duration between a pair of two nodes is very low over a period, either node will delete the corresponding item in its contact table, since the other node is just a 'stranger'. It can be seen that with this deletion mechanism, the number of items in the contact table of a node is very limited, even though the total number of nodes in the network may be very large. Therefore, our proposed cache refreshing scheme is practical and scalable.

As explained previously, the DAT is a dynamic structure which is evolving in both space and time. Therefore, the data access and refreshing process, modelled as a dynamic graph, is consistent with the opportunistic nature of the mobile network. At any specific time instance, such a structure is implicit to any node connected to it, and no node has the complete information of the whole structure, as it is impossible and unnecessary to keep a complete track of such an evolving structure. Thus, the proposed cache refreshing scheme is a distributed algorithm, and the state information kept at each node is very limited. Specifically, a node is only required to keep relevant local information, namely, the information of the nodes that 'connect' to it, rather than the state of the whole network. In particular, child node E keeps the record of connection to its parent node B. As for parent node B, it keeps the information of the child node E, including the version of the data cached at E and the last time the cached data is refreshed, so that it can generate the appropriate updating file for its child node E and calculate its utility of refreshing E. Such information is updated when B contacts E. To opportunistically selecting relay to enhancing cache refreshing performance, parent node B also needs to keep the record of potential relay nodes. As explained in the previous paragraph, the number of appropriate potential relays is actually quit small. The mobility statistics of these potential relays can be saved at the memory of B without frequent update. When B has chosen a relay R to help delivering the updating file to its child node E, it passes the updating file together with the ID of E to R. Note that such record will be deleted from the memory of R when relay R has successfully delivered the updating file to E or it finds that the data cached at E has been refreshed by other nodes. Therefore, the information kept at each relay node is also very limited. It can be seen that since the state information kept at each node is very limited, the proposed cache refreshing scheme with an opportunistic selection of relay is cost efficient.

5 Performance evaluation

We evaluate the performance of our proposed **ITM-based refreshing** scheme, which uses (8) and (10) to measure updating utilities and selecting relays. In order to demonstrate its effectiveness, we compare the performance of our scheme with the following schemes:

- *OTM-based refreshing [7]:* Updating file utility is calculated based on the assumption that data can be delivered in one connection, which we call One-time Transmission Model (OTM). Adopting the formula (4) in [7], the utility measures the probability of pairwise contact before the source is refreshed, and based on which the parent node selects relays. The major difference with our ITM-based refreshing scheme is in selecting relays. Specifically, the OTM-based refreshing calculates the utility of a potential relay assuming that the updating file is guaranteed to be delivered in one contact, which is generally untrue.
- *Passive refreshing:* Updating files are disseminated through the DAT. A node can only get updating information from its parent node. When the CCD is longer than θ, the complete data is delivered. It can be seen that the difference with the ITM-based and OTM-based schemes is that the passive refreshing scheme does not rely on relay.
- *Active refreshing [30]:* Each time when the source is refreshed, it disseminates the updating file to the whole network via epidemic routing [30]. In epidemic routing, a node simply 'infects' any node it contacted with the data. Therefore, the active refreshing scheme attains the highest possible refreshing performance in terms of the percentage of caching nodes receiving their required updating files, but the cost may become prohibitively large, particularly for the networks with a large number of nodes.

We use the following three metrics to evaluate the performance of each scheme:

- *Refreshing ratio:* The percentage of caching nodes receiving the corresponding updating files.
- *Refreshing overhead:* The average number of nodes (including parents and relays) that are used to deliver an updating file from a parent to a child. For example, as depicted in Fig. 2*a*, in order to deliver the updating file to node C, node A selects nodes R_1 and R_2 as relays. Therefore, the refreshing overhead of refreshing C is 3.
- *Refreshing delay:* The average delay for updating files being delivered from the data source to the caching nodes.

5.1 Simulation setup

Our evaluations are conducted on two realistic mobile opportunistic network traces, which cover two types of mobile network environments, urban area (Cambridge) [31] and conference site (Infocom) [32]. In Cambridge trace mobile devices are carried by students of Cambridge University, while in Infocom trace mobile devices are carried by students attending the student workshop. The information of the two traces is summarised in Table 1.

Queries at all nodes are randomly generated following a Zipf distribution, which is widely used for modelling web data access [33]. Denote $p_i \in [0, 1)$ as the probability that the data item i is requested and M as the number of data items in the network. Similar to [7], M is set to 4, and we have $p_i = (1/i)/(\sum_{j=1}^{M}(1/j))$. Nodes who are interested in the same data item comprise a node set. In each set, the node having the highest contact frequency is selected as the data source. We generate inter-refreshing intervals of the source data following an exponential distribution. Considering the long inter-contact interval and short contact duration, we expand the average inter-refreshing intervals to 14 and 32 times of those used in [7] for Infocom trace and Cambridge trace, respectively, in our simulations. Thus the average intervals of the four data items are 10.03, 7.60, 11.55 and 11.91 h for Infocom trace, and they are 22.93, 17.37, 26.39 and 27.22 h for Cambridge trace. In the simulations we assume that θ is the same for all the updating files, although our scheme is equally applicable with different θs for different updating files.

5.2 Results and discussion

We first compare the performance of our proposed refreshing scheme with those of the other three schemes by varying the data lifetime L of the cached data while fixing the CCD threshold value θ and setting $U_{cr} = 50\%$. To increase the probability of successfully delivering updating files, a parent node will continue selecting new relays even if $\bar{U} \geq U_{cr}$. Since Infocom and Cambridge traces represent the two networks of different spatial dimensions with very different mobility statistics, see Table 1, values of L and θ used for these two network simulations are very different. The evaluation results are shown in Figs. 6 and 7 for Infocom trace and Cambridge trace, respectively. It can be seen from both Figs. 6 and 7 that with the increase of L, the refreshing ratio is increased and the refreshing overhead is reduced. This is because with a larger L, a larger amount of data are cached in the network and the caching nodes can wait for a longer time to be refreshed. This lowers the average delivery cost, while increasing the opportunities for the updating files to be delivered to the caching nodes before the data expiration. It is obvious that in general the epidemic routing based active refreshing scheme will achieve the highest refreshing ratio at the cost of the highest refreshing overhead. It is also obvious that in general the passive refreshing scheme will achieve the lowest refreshing ratio while imposing the lowest

Table 1 Trace summary

Trace	Infocom	Cambridge
duration, days	3	11
number of devices	41	36
number of contacts	5370	3497
average inter-contact interval, hours	5.7106	16.9585
average pairwise node contact duration, minutes	8.1564	28.9537

refreshing overhead. Significantly, our proposed refreshing scheme attains the second highest refreshing ratio (below the active scheme) and achieves the second lowest refreshing overhead (above the passive scheme). In particular, observe that the proposed ITM-based scheme requires less relays than the OTM-based scheme and yet attains better refreshing ratio. Evidently, our contact-duration aware scheme is capable of selecting more appropriate relays to aid cache refreshing. This demonstrates the effectiveness of our proposed method.

For the Infocom case, it can be seen from Fig. 6*a* that at the data lifetime of $L = 7$ h and the CCD threshold of $\theta = 20$ min, the proposed scheme attains the refreshing ratio of 12%, which may appear to be very low. However, under this extremely difficult environment, even the active scheme only attains around 27% of refreshing ratio. Although for this small-scale network with only around 40 nodes, the active scheme may more than double the refreshing ratio at this operation environment, it also requires nearly three times of the refreshing overhead, i.e. the cost, as can be seen from Fig. 6*b*. Given that the nodes' mobility statistics are specified by the trace and the set of the inter-refreshing intervals of the source data is also specified, the attainable cache refreshing performance is obviously depends on the data lifetime L and the CCD threshold θ. For the Infocom case with $L = 7$ h and $\theta = 5$ min, the proposed scheme, for example attains a refreshing ratio of 33%, as will be seen from Fig. 8*a*.

We next evaluate the performance of the four cache refreshing schemes with the different requirements for the CCD threshold value θ while fixing the data lifetime L. The results obtained are shown in Figs. 8 and 9, respectively, for Infocom trace and Cambridge trace. It can be seen from the results of Figs. 8 and 9 that increasing the CCD threshold value θ leads to a decrease in the cache refreshing performance and an increase in the cache refreshing cost, for all the four schemes. This is because with a larger CCD threshold value θ, it needs more contact times to deliver updating files and moreover, due to the intermittent network connectivity, the parent nodes replicate more updating file copies to satisfy the cache freshness requirements and there are more cached data expired before being refreshed. As expected, in general the active refreshing scheme achieves the highest cache refreshing performance at the highest cost, while the passive refreshing scheme obtains the lowest cache refreshing performance at the lowest cost. Again, it can be observed from Figs. 8 and 9 that our proposed contact-duration-aware cache refreshing scheme attains the second highest refreshing ratio, while imposing the second lowest refreshing overhead. This further confirms the effectiveness of our proposed scheme.

Finally, we investigate the refreshing delay performance of the four schemes. Specifically, by varying the data lifetime L while fixing the CCD threshold value θ, we compare the refreshing delay performance of the four schemes in Fig. 10 for both Infocom and Cambridge traces. Similarly, Fig. 11 compares the refreshing delay performance of the four schemes, obtained by changing the CCD threshold value θ while fixing the data lifetime L. The first observation from both Figs. 10 and 11 is that the active scheme generally has the worst delay performance. This is not surprising, as the epidemic routing adopted by the active scheme relies purely on random encounter patterns of the nodes. By contrast, the other three schemes all exploit the hierarchical DAT structure to aid the cache refreshing process, which is beneficial for speeding up the delivery process. It can also be seen that with the fixed data lifetime L, increasing the CCD threshold θ increases the delay of the active scheme, while the refreshing delay of the active scheme is insensitive to L. This is because the data delivery of the active scheme depends mainly on the mobility patterns, and therefore L has little influence on its delay performance. With increasing θ, often the data cannot be delivered in a single contact, and this leads naturally to a longer delay in delivering the updating file.

The third observation from both Figs. 10 and 11 is that the proposed cache refreshing scheme attains a competitive performance in terms of refreshing delay, compared to the OTM-based and passive schemes. Moreover, with a fixed CCD threshold θ, increasing the data lifetime L generally leads to increase in refreshing delay for these three schemes, as can be seen from Fig. 10. This is because, with a larger L, a parent node can afford to wait for longer to deliver the updating file and this results in an increase in delay. However, with a fixed L, the influence of θ to the delay performance of these three schemes is more complicated, which appears to also depend on the spatial dimension of the network and the mobility statistics. Specifically, for the Cambridge case, increasing the CCD threshold generally leads to increase in the delay performance of these three schemes, as can be clearly seen from Fig. 11*b*. For the Infocom case, by contrast, with the increase of θ, the refreshing delays of these three schemes first increase slightly, then stay almost flat and finally decrease slightly, as can be seen from Fig. 11*a*. We believe that

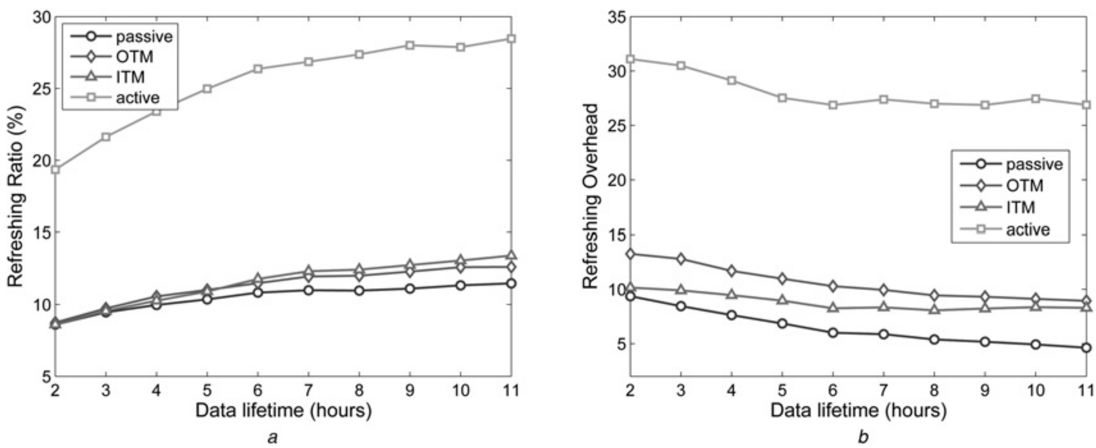

Fig. 6 *Performance comparison of maintaining cache freshness given different lifetimes of cached data using four schemes for Infocom trace. CCD threshold value is given by $\theta = 20$ min*

a Refreshing ratio
b Refreshing overhead

IET Netw., 2016, Vol. 5, Iss. 4, pp. 93–103

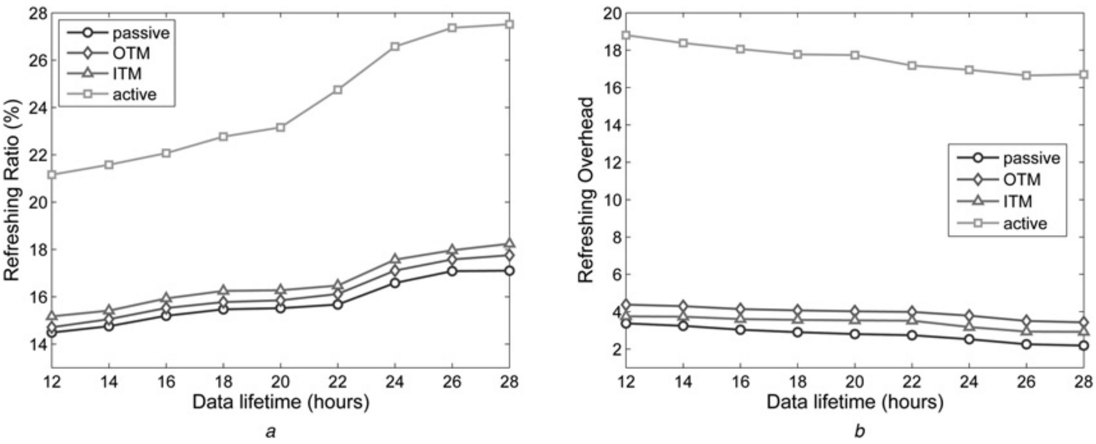

Fig. 7 *Performance comparison of maintaining cache freshness given different lifetimes of cached data using 4 schemes for Cambridge trace. CCD threshold value is given by θ = 50 min*

a Refreshing ratio
b Refreshing overhead

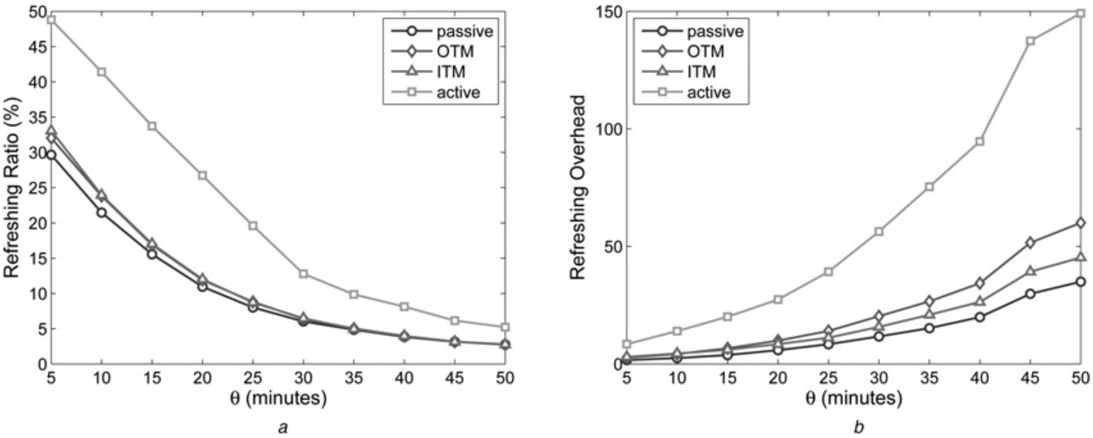

Fig. 8 *Performance comparison of maintaining cache freshness given different CCD threshold values using four schemes for Infocom trace. Cached data lifetime is given by L = 7 h*

a Refreshing ratio
b Refreshing overhead

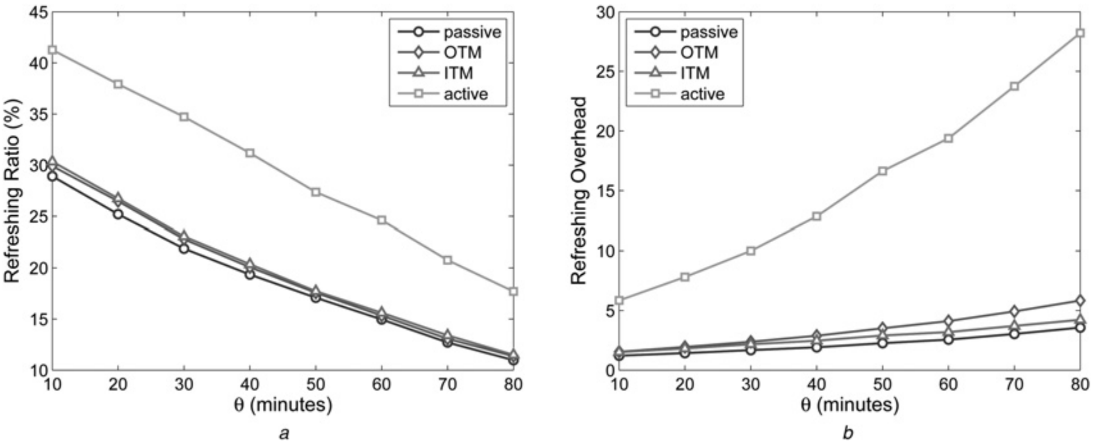

Fig. 9 *Performance comparison of maintaining cache freshness given different CCD threshold values using four schemes for Cambridge trace. Cached data lifetime is given by L = 26 h*

a Refreshing ratio
b Refreshing overhead

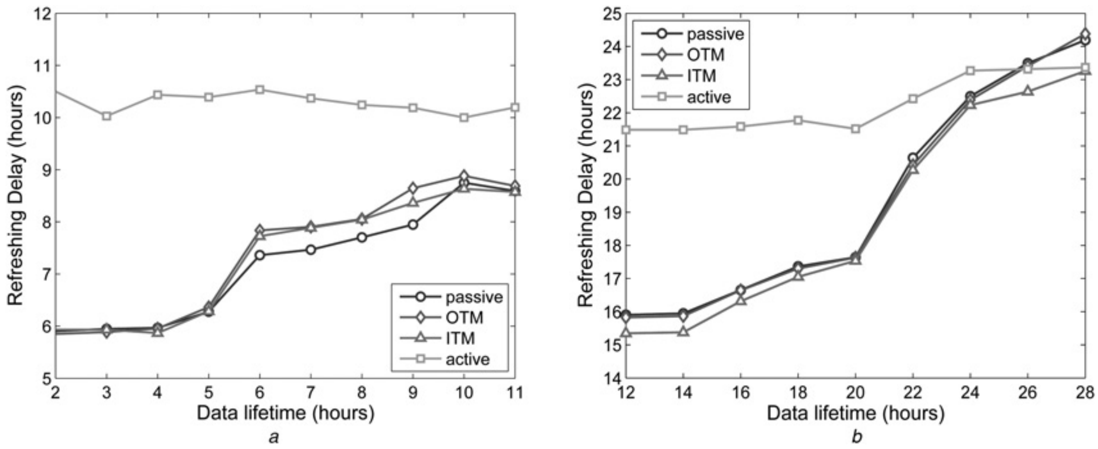

Fig. 10 *Comparison of the refreshing delay performance of the four scheme given different data lifetimes*

a Infocom ($\theta = 20$ min)
b Cambridge ($\theta = 50$ min)

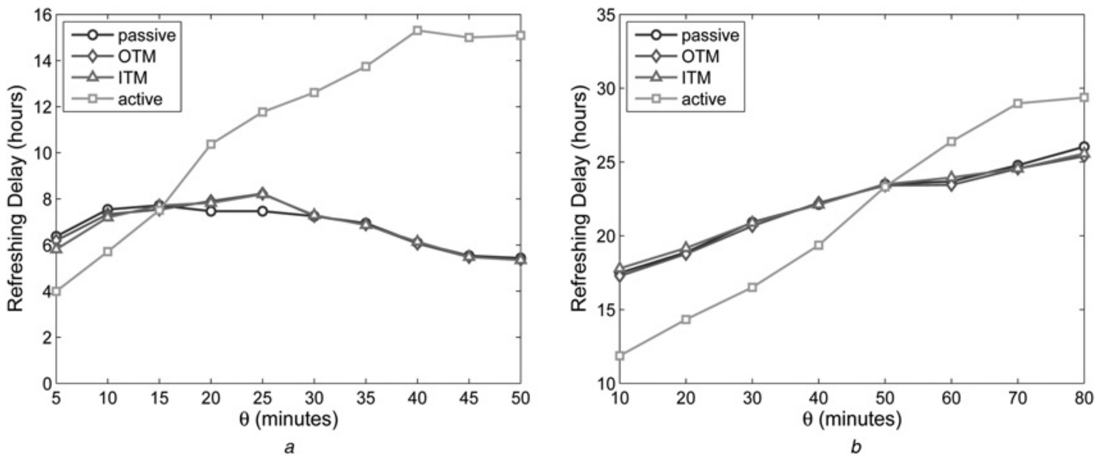

Fig. 11 *Comparison of the refreshing delay performance of the four schemes given different CCD threshold values*

a Infocom ($L = 7$ h)
b Cambridge ($L = 26$ h)

this complex delay performance is linked to the dynamic hierarchical DAT structure, which is spatially and temporally varying. The coupling effects of this dynamic DAT structure with the mobility statistics are highly complicated, and this is the main reason that the delay performance of these three schemes depends on the spatial dimension of the network and the mobility statistics. It is worth recalling that Infocom trace is for a conference site with an average inter-contact interval of 5.7 h and an average contact duration of 8.2 min, while Cambridge trace is for a larger urban area with average inter-contact interval of 17 h and average contact duration of 29 min.

6 Conclusions

We have proposed a contact duration aware cache refreshing scheme for mobile opportunistic networks. Specifically, we have established a mathematical model to describe the data delivery, where a complete data item can only be transmitted from one node to another via one or several intermittent contacts provided that the cumulative contact duration is sufficiently long. Based on this proposed ITM, we have investigated a hierarchical cache refreshing scheme, where nodes are organised as a dynamic tree structure that is evolving in both space and time, owing to the nature of mobile opportunistic networks, and updating files are disseminated along the tree. Moreover, we have discussed how to select relay nodes outside the tree so that updating files can be replicated to the selected relays to further enhance the probability of successful data delivery. The proposed contact-duration aware cache refreshing scheme with opportunistic relay selection is a distributed algorithm, and at any specific time, the cache refreshing structure is implicit to any node connected to it, while each node is only required to keep very limited relevant local information. The performance of our proposed scheme is evaluated by the trace-driven simulation on two realistic mobile traces using some existing cache refreshing schemes as benchmarks, and the results obtained have demonstrated the effectiveness of our contact duration aware cache refreshing scheme.

7 References

1 Li, Y., Hui, P., Jin, D., *et al*.: 'Evaluating the impact of social selfishness on the epidemic routing in delay tolerant networks', *IEEE Commun. Lett.*, 2010, **14**, (11), pp. 1026–1028
2 Gao, W., Cao, G.: 'User-centric data dissemination in disruption tolerant networks'. Proc. 2011 IEEE INFOCOM, Shanghai, China, April 2011, pp. 3119–3127
3 Li, Y., Jiang, Y., Jin, D., *et al*.: 'Energy-efficient optimal opportunistic forwarding for delay-tolerant networks', *IEEE Trans. Veh. Technol.*, 2010, **59**, (9), pp. 4500–4512
4 Yoneki, E., Hui, P., Chan, S., *et al*.: 'A socio-aware overlay for publish/subscribe communication in delay tolerant networks'. Proc. 10th MSWIM, Chania, Crete Island, Greece, October 2007, pp. 225–234

5 Li, F., Wu, J.: 'MOPS: Providing content-based service in disruption-tolerant networks'. Proc. 29th IEEE ICDCS, Montreal, Quebec, Canada, June 2009, pp. 526–533
6 Li, Y., Su, G., Wu, D., *et al*.: 'The impact of node selfishness on multicasting in delay tolerant networks', *IEEE Trans. Veh. Technol.*, 2011, **60**, (5), pp. 2224–2238
7 Gao, W., Cao, G., Srivatsa, M., *et al*.: 'Distributed maintenance of cache freshness in opportunistic mobile networks'. Proc. IEEE 32nd ICDCS, Macau, China, June 2012, pp. 132–141
8 Ioannidis, S., Massoulie, L., Chaintreau, A.: 'Distributed caching over heterogeneous mobile networks'. Proc. ACM SIGMETRICS 2010, New York, June 2010, pp. 311–322
9 Gao, W., Cao, G., Iyengar, A., *et al*.: 'Supporting cooperative caching in disruption tolerant networks'. Proc. 31st ICDCS, Minneapolis, MN, June 2011, pp. 151–161
10 Cao, P., Liu, C.: 'Maintaining strong cache consistency in the world wide web', *IEEE Trans. Comput.*, 1998, **47**, (4), pp. 445–457
11 Duvvuri, V., Shenoy, P., Tewari, R.: 'Adaptive leases: a strong consistency mechanism for the world wide web', *IEEE Trans. Knowl. Data Eng.*, 2003, **15**, (5), pp. 1266–1276
12 Cao, G.: 'A scalable low-latency cache invalidation strategy for mobile environments', *IEEE Trans. Knowl. Data Eng.*, 2003, **15**, (5), pp. 1251–1265
13 Cao, J., Zhang, Y., Cao, G., *et al*.: 'Data consistency for cooperative caching in mobile environments', *Computer*, 2003, **40**, (4), pp. 60–66
14 Huang, Y., Cao, J., Jin, B., *et al*.: 'Flexible cache consistency maintenance over wireless *ad hoc* networks', *IEEE Trans. Parallel and Distrib. Syst.*, 2010, **21**, (8), pp. 1150–1161
15 Li, Y., Wang, Z., Jin, D., *et al*.: 'Collaborative vehicular content dissemination with directional antennas', *IEEE Trans. Wirel. Commun.*, 2012, **11**, (4), pp. 1301–1306
16 Chaintreau, A., Hui, P., Crowcroft, J., *et al*.: 'Pocket switched networks: real-world mobility and its consequences for opportunistic forwarding'. Tech. Rep. UCAM-CL-TR-617, Computer Lab, Cambridge University, UK, Feb 2005
17 Wang, W., Srinivasan, V., Motani, M.: 'Adaptive contact probing mechanisms for delay tolerant applications'. Proc. MobiCom'07, Montreal, Canada, September 2007, pp. 230–241
18 Li, Y., Jin, D., Wang, Z., *et al*.: 'Exponential and power law distribution of contact duration in urban vehicular *ad hoc* networks', *IEEE Signal Process. Lett.*, 2013, **20**, (1), pp. 110–113
19 Li, Y., Jin, D., Hui, P., *et al*.: 'Revealing contact interval patterns in large scale urban vehicular *ad hoc* networks', *ACM SIGCOMM Comput. Commun. Rev.*, 2012, **42**, (4), pp. 299–300
20 Whitbeck, J., Conan, V., Dias de Amorim, M.: 'Performance of opportunistic epidemic routing on edge-Markovian dynamic graphs', *IEEE Trans. Commun.*, 2011, **59**, (5), pp. 1259–1263
21 Li, Y., Qiu, L., Jin, D., *et al*.: 'Contact duration aware evaluation for content dissemination delay in mobile social network', *Wirel. Commun. Mob. Comput.*, 2015, **15**, (3), pp. 527–537
22 Zhuo, X., Li, Q., Gao, W., *et al*.: 'Contact duration aware data replication in delay tolerant networks'. Proc. 19th IEEE ICNP, Vancouver, Canada, October 2011, pp. 236–245
23 Zhuo, X., Li, Q., Cao, G., *et al*.: 'Social-based cooperative caching in DTNs: A contact duration aware approach'. Proc. Eigth Int. Conf. Mobile Ad-hoc and Sensor Systems, Valencia, Spain, October 2011, pp. 92–101
24 Mogul, J., Douglis, F., Feldmann, A., *et al*.: 'Potential benefits of delta encoding and data compression for http', *SIGCOMM Comput. Commun. Rev.*, 1997, **27**, (4), pp. 181–194
25 Lenders, V., Wagner, J., Heimlicher, S., *et al*.: 'An empirical study of the impact of mobility on link failures in an 802.11 *ad hoc* network', *IEEE Wirel. Commun.*, 2008, **15**, (6), pp. 16–21
26 Zhu, H., Li, M., Fu, L., *et al*.: 'Impact of traffic influxes: revealing exponential inter-contact time in urban VANETs', *IEEE Trans. Parallel Distrib. Syst.*, 2011, **22**, (8), pp. 1258–1266
27 Wang, W., Srinivasan, V., Motani, M.: 'Adaptive contact probing mechanisms for delay tolerant applications'. Proc. 13th ACM Int. Conf. Mobile Computing and Networking (MOBICOM), Quebec, Canada, September 2007, pp. 230–241
28 Takács, L.: 'On certain sojourn time problems in the theory of stochastic processes', *Acta Math. Acad. Sci. Hung.*, 1957, **8**, (1-2), pp. 169–191
29 Hui, P., Crowcroft, J.: 'Human mobility models and opportunistic communications system design', *Philos. Trans. R. Soc. A, Math. Phys. Eng. Sci.*, 2008, **366**, (1872), pp. 2005–2016
30 Vahdat, A., Becker, D.: 'Epidemic routing for partially-connected ad hoc networks'. Tech. Rep., Duke University, 2000
31 Leguay, J., Lindgren, A., Scott, J., *et al*.: 'Opportunistic content distribution in an urban setting'. Proc. 2006 SIGCOMM Workshop Challenged Networks, Pisa, Italy, September 2006, pp. 205–212
32 http://www.haggleproject.org, accessed January 2014
33 Breslau, L., Cao, P., Fan, L., *et al*.: 'Web caching and zipf-like distributions: evidence and implications'. Proc. IEEE INFOCOM'99, New York, March 1999, pp. 126–134