Towards D2D-based Opportunistic Data Relay Service in Partial Not-spots

G. Chandrasekaran[†], N. Wang[†], J. Jun[*], M. Xu[*] and R. Tafazolli[†]
[†]University of Surrey, United Kingdom
{g.chandrasekaran, n.wang, r.tafazolli}@surrey.ac.uk
[*]Tsinghua University, China
{ji-j14, xumw}@tsinghua.edu.cn

Abstract— With the recent development of Device-to-Device (D2D) communication technologies, mobile devices will no longer be treated as pure "terminals", but they could become an integral part of the network in specific application scenarios. In this paper, we introduce a novel scheme of using D2D communications for enabling data relay services in partial Not-Spots, where a client without local network access may require data relay by other devices. Depending on specific social application scenarios that can leverage on the D2D technology, we consider tailored algorithms in order to achieve optimised data relay service performance on top of our proposed network-coordinated communication framework. The approach is to exploit the network's knowledge on its local user mobility patterns in order to identify best helper devices participating in data relay operations. This framework also comes with our proposed helper selection optimization algorithm based on reactive predictability of individual user. According to our simulation analysis based on both theoretical mobility models and real human mobility data traces, the proposed scheme is able to flexibly support different service requirements in specific social application scenarios.

I. INTRODUCTION

In the recent years, mobile infrastructure provisioning, in particular concerning radio network coverage in specific geographical areas, has become a major concern for mobile operators. In this context, one specific issue to be tackled is *partial not-spots*, which means a location or area that is not covered by all mobile networks. This means mobile users under one network do not have service coverage in a given area whereas customers from another network may get the service at the same location. According to [1], over 21% of the UK areas are affected by partial not-spots, meaning that even customers with major mobile operators may find themselves out of service coverage in some areas, to note that mobile coverage in UK is already one among the best in Europe. However, the good news is that a partial not-spot is covered by at least one mobile operator, in which case it can be envisaged that a device without local network coverage can leverage another device in its proximity as a data relay node, which has network access in order to get network connections.

Recent technology development in D2D communications has enabled direct communication between mobile devices [1] without necessarily involving cellular links in data transmissions. Emerging mobile devices such as smart phones are featured with high computational power, larger storage capacity, long battery life, accurate GPS location functions, as well as heterogeneous wireless interfaces such as cellular, WiFi and Bluetooth. Such advanced features allow mobile devices to play a more active role in mobile networks rather than simply being a dummy terminal for consuming mobile data. In the context of 3GPP/ETSI, investigations on D2D communications have been mainly focusing on emergency calls, disaster recovery and public safety services [2]. In this paper we propose a new paradigm by which devices (User Equipment (UE)) are encouraged to provide Connectivity as a service (CaaS)/relay service for other nodes/devices in need. We mainly address how D2D can become an enabling technology for providing network coverage services for affected mobile end users in partial not-spots/disaster affected area/IoT environment, especially with regard to various applications requirements.

While it is attractive to conceive the idea of having D2D to provide CaaS to other mobile clients, its realisation requires sophisticated signalling protocol design, necessary knowledge about user mobility/locations, efficient D2D coordination algorithms and also practicality considerations. Towards this end, in this paper we start from the overall architecture design based on the emerging Mobile Edge Computing (MEC) paradigm [3], which also includes the specification of signalling protocol for enabling the CaaS paradigm along with network coordination. From here on we refer a user device which requires connectivity as *client*, and a device that is able to provide CaaS to client as helper. It is well-known that D2D communications are by nature opportunistic according to human mobility. That is, the opportunities and the duration of getting connected through D2D based data relay can be very uncertain for clients and might involve multiple helpers to satisfy a single client's relay. In this case, it is essential to understand how opportunistic communications based on different user mobility patterns can best cater for specific application requirements. Based on this, the next question is, if the network is able to capture and predict the mobility patterns of its own customers, how to identify optimised *helpers* based on such knowledge in order to best serve an incoming data relay request from a client.

On the other hand, we are particularly interested in optimised D2D coordination according to different application requirements and human mobility patterns. In this context, we mainly focus on the following two scenarios in which a client would request network connection with distinct requirements:

- Maximum Continuity (MC): Client in need of relatively *stable connection without frequent or long-duration loss* *of connectivity* would require the MC algorithm for their helper selection. Typical applications in this case may include Whatsapp-like text chatting or following live update of events such as football matches.

- Early Access (EA): In comparison, EA requires the availability of immediate Internet access, but not necessarily seamless connectivity. A typical scenario is that a user would like to send out (or receive) messages as soon as possible upon arriving at a place without own network connectivity.

As can be seen, depending on specific application requirements, clients may have very different user experience expectations on their network connection conditions. In this article we strive to shed lights on these technical issues by proposing a framework for supporting network-coordinated D2D based data relay services for clients. We first illustrate the proposed MC and EA algorithms against realistic mobility trace with 100% accuracy in prediction algorithm. Since human mobility has uncertainty in its predictability i.e. not always 100% predictable, this framework also comes with our proposed helper selection optimisation algorithms based on reactive predictability of individual user mobility/locations. The proposed scheme has been critically analysed under various level of prediction algorithm's predictability (or user's randominity), and comprehensively evaluated against four specific human mobility models and one real datasets in existence. We also discuss how our proposed reactive predictability algorithm takes reactive decision to cater individual helper's randominity, based on which we present quantified benefits over normal mobility prediction algorithms. Towards the end of the article we will provide our systematic analysis on how these mobility patterns may affect the D2D data relay service based on the nature of opportunistic communications.

II. Overall Framework Design

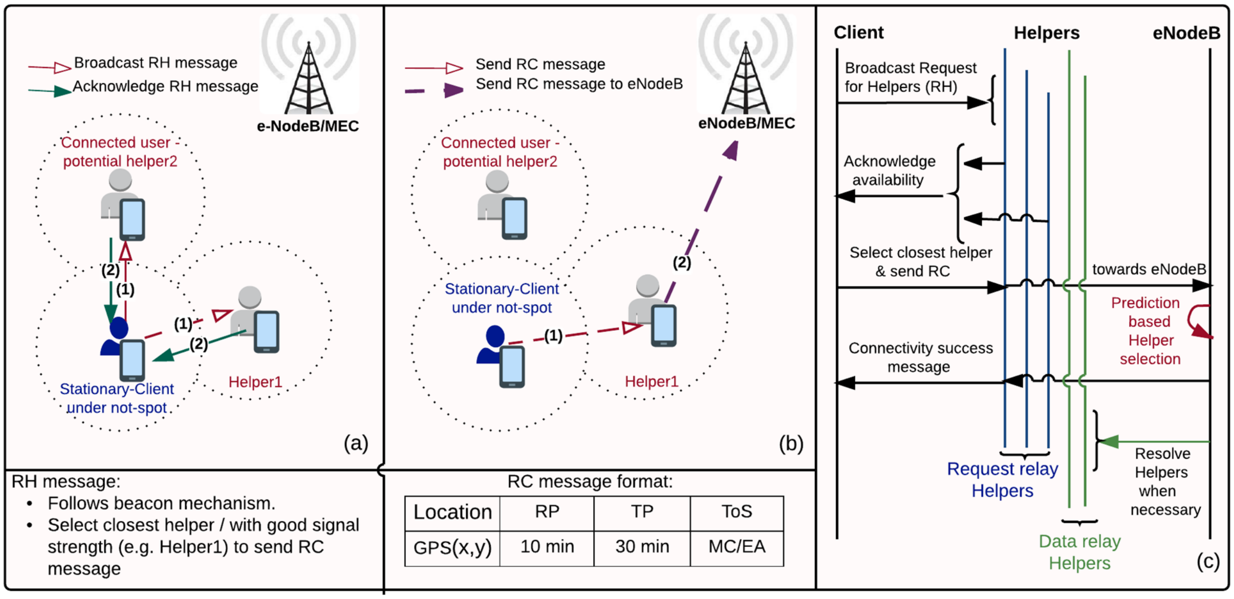
A client at non-coverage area obtains data connectivity by using helpers as a relay, the control plane functions and intelligence of decision making are mainly located at the cellular network edge, typically at the eNodeB (eNB) level, which also needs to maintain mobility profiles of local helper candidates. Specific client-helper relay pairing with the help of eNB is achieved in three stages such as the following.

A. Connectivity request delivery

Whenever a client wants to obtain data relay service in a partial not-spot area, it broadcasts a Request for Helper message (RH) in proximity. Available helpers receiving this broadcast will acknowledge by conveying its availability, this process is depicted in Fig.1a. The client then will send Request for Connectivity message (RC) to eNB via the closest (good D2D signal strength) acknowledged local helper, as in Fig.1b. RC message contains four fields, "*Current location*", "*Request period (RP)*", "*Tolerable period (TP)*" and "*Type of service (ToS)*". Current GPS coordinates of client is specified in current location field. The RP can specify the time duration for which client needs data connectivity and TP indicates the tolerable time limit within which RP duration has to be satisfied. For instance, RC message with RP=10minutes and TP=30 minutes, means client needs 10 minutes of connection within 30 minutes from the time when the request is issued. The last field in RC message is ToS which can indicate the desirable requested connection pattern, such as *"Maximum Connectivity (MC)"* or "*Early access (EA)*". These are the two types of requirement scenarios for which we will propose specific helper selection algorithms in the next section. Such an RC message received at eNB will be acknowledged towards the client via (by default) the same helper based on the decision algorithm outcome (either with a "*Connection success message"* or "*No Connection message")*. The whole signalling is depicted in Fig.1 (c), in general a helper does two different functions, to relay requests and to relay data.

B. Decision-Making Logic on Helper Selections

Each eNB needs to maintain necessary profile on GPS-assisted mobility information of helper candidates in order to predict the future spatiotemporal mobility pattern.


Broadcast RH message
Acknowledge RH message
e-NodeB/MEC
Connected user - potential helper2
(2)
(1)
(1)
(2)
Helper1
Stationary-Client under not-spot
(a)
Send RC message
Send RC message to eNodeB
eNodeB/MEC
Connected user - potential helper2
Stationary-Client under not-spot
(1)
(2)
Helper1
(b)
Client
Helpers
eNodeB
Broadcast Request for Helpers (RH)
Acknowledge availability
Select closest helper & send RC
towards eNodeB
Prediction based Helper selection
Connectivity success message
Resolve Helpers when necessary
Request relay Helpers
Data relay Helpers
(c)

RH message:
- Follows beacon mechanism.
- Select closest helper / with good signal strength (e.g. Helper1) to send RC message

RC message format:

Location	RP	TP	ToS
GPS(x,y)	10 min	30 min	MC/EA

Figure 1, RH and RC message delivery from client to helpers in partial not-spots

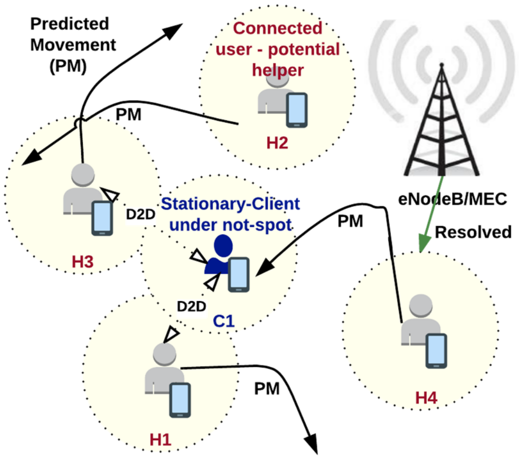

Predicted Movement (PM)
Connected user - potential helper
H2
PM
H3
D2D
Stationary-Client under not-spot
PM
eNodeB/MEC
Resolved
D2D
C1
H4
PM
H1

Figure 2, Data connectivity service delivery to client via helpers based on prediction at eNB

Upon an incoming RC message, the eNB takes the helper resolver role by looking up predictive mobility pattern profiles of the local helper candidates for possible data relay. Specific strategies for helper selection are based on received RC message's ToS field, detailed algorithm on such selection will be discussed in next section. Since there is only opportunistic D2D connection between helpers and client, chances are that multiple helpers may be required to satisfy a single request from a client. Towards this end, the eNB (resolver) should have the knowledge about the inferred location of the client (through location information embedded in RC message received), and helpers' predictive movements, typically this can be made available by various predictive movement models from literature [4-7]. In order to successfully resolve every incoming request, the eNB needs to maintain a pending request table (PRT) which records the states for those RC messages that have been directed towards local helpers. Upon the receipt of a successful acknowledgement for completion of RC, the corresponding request is removed from the PRT.

C. Data relay operations via helpers

The resolved helpers can either be already in proximity with the client by the time the request is made, or they might be currently outside the communication range of the client, but is predicted to be in proximity with client within the TP duration. If already in proximity, the eNB can immediately resolve the request to target helper. In later case, the eNB will continuously monitor for changes in target helper's mobility pattern against its profile. In case an identified helper candidate follows the predictive pattern and actually moves within the proximity to the client as predicted, and only then the eNB will actually resolve the request to helper. On any change in the predicted mobility of the target helpers the eNB has to act quickly and find alternative helpers, or eNB has to send *no connection message* to client via any next available helper in proximity (and remove respective entry from PRT as a failed case). By this way all extra overheads can be saved in case the prediction algorithm doesn't meet the standard. Once the eNB resolves a request to a particular helper, the helper is then expected to get ready to act as a data relay for the requested client. On completion of relay period the eNB takes care of billing and rewarding according to the data usage, and this feature is outside the scope of this article.

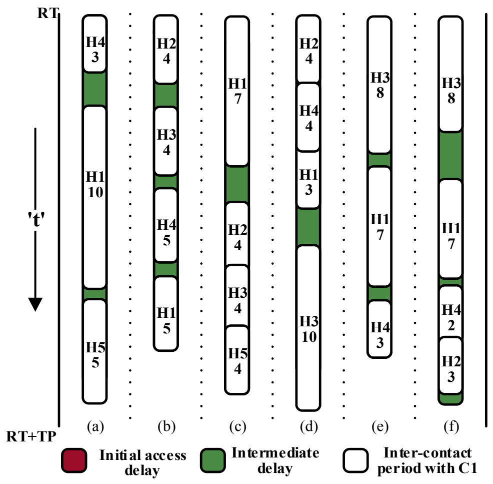
As shown in Fig. 1c, after receiving the acknowledgement for the RH message, a client sends a RC message via the closest helper to the eNB. Immediately an entry is created in its PRT for client's RC according to the ToS field. One such example is depicted in Fig.2, where the eNB predicts helper H4 as a potentially viable candidate to relay data service for client C1 within its TP duration. This is because helpers H1, H2 and H3 are predicted to move away from the client, while H4 is predicted to move toward C1 and stay for RP duration. So in this case, the eNodeB will wait for H4 to arrive in proximity with C1 as predicted, and once this happens eNB will immediately resolve the request to H4, after which H4 will act as a data relay helper for client C1. Upon the RP completion, C1 acknowledges the eNB via the last data helper, and the respective entry is removed from the PRT. To note that here in the example, helper1 (H1) acts as the *request* relay helper and H4 will act as *data* relay helper.

III. Prediction Based Helper Selection Algorithms

Mobile users with highly predictable mobility patterns are ideal for becoming helper candidates. In fact, predicting mobility patterns of users has been a widely analysed topic in the literature, C. Song et al's work on prediction indicates that there is a potential between 80% and 93% average predictability in user mobility [4]. By leveraging on the existing prediction algorithms [5-9], eNB can predict movement of helpers for TP duration, and then select suitable candidates which can relay connectivity to client based on algorithm specified in ToS field of RC message.

In this article we propose two representative types of helper selection algorithm according to client's application requirements, namely "*Maximum Connectivity (MC)*", and "*Early access (EA)*". Detailed working of MC and EA algorithms are shown in Fig.3b. MC algorithm aims to achieve maximum continuous connectivity once getting connected, specifically referring to minimum intermediate delay. As an example, in Fig.3a, we illustrate with a number of possible scenarios in which each column represents the predicted inter-contact time duration of helpers with C1 in minutes, from the requesting time (RT, time the helper relays RC to eNB, hence no initial access delay shown) until TP completion, already sorted according to their start time. Assuming RP is 10 minutes, in Fig.3a (case (a) and (b)), we have two and three possible candidate options to satisfy C1 such as {H4,H1}, {H1} for case (a) and {H2,H3,H4}, {H3,H4,H1}, {H4,H1} for case (b), with each option being able to satisfy the 10-miniute request. The MC algorithm then selects the list of helpers which has minimum intermediate delay, not necessarily concerning the initial access delay and number of helpers in the list. In this case that would be {H1} and {H4,H1} for case (*a*) and (*b*) respectively.

In some special cases the MC and EA algorithm might have to deal with tie-breaks, i.e. there might be multiple possible combinations of helper lists with the same intermediate delay D and initial access delay I respectively. Solution for such tie-breaker scenario is explained in algorithm section shown in Fig.3b.



MC	H1	H4,H1	H2,H3, H5	H3	H3,H1	H1,H4, H2
EA	H4,H1	H2,H3, H4	H1,H2	H2,H4, H1	H3,H1	H3,H1

(a) Examples

Maximum Continuity (MC)

```
eNodeB predicts time and inter-contact duration of all helpers to be in proximity with client.
   Define: D as accumulated intermediate delay.
Identify all the possible combinations lists along with D that can satisfy RP.
Select the Helpers combination list with minimum intermediate delay 'D'.
if (multiple combinations of list with same D) then
    Select list with minimum number of helpers
    if (multiple combinations of list with equal number of helpers) then
        Select list with minimum initial access delay
    end if
end if
Resolve all helpers from selected list when necessary.
```

Early Access (EA)

```
eNodeB predicts time and inter-contact duration of all helpers to be in proximity with client.
Identify all the possible combinations lists that can satisfy RP.
Select list with minimum initial access delay.
if (multiple combinations of list with same initial access delay) then
    Select the Helpers combination list with minimum intermediate delay 'D'.
    if (multiple combinations of list with same D) then
        Select list with minimum number of helpers
    end if
end if
Resolve all helpers from selected list when necessary.
```

(b) Pseudocode

Figure 3, Algorithm illustrations for MC and EA

It can be inferred that signalling overhead is actually the function of the number of times the data relaying helpers are replaced due to their mobility. Take case (d) in Fig. 3a as an example, there exist two possible candidate lists: {H2,H4,H1} (with overlapping duration of in-proximity between consecutive helpers) and {H3}, with the tie of zero intermediate delay in both scenarios. In this case the MC algorithm chooses list {H3} which has minimum number of helpers involved. Finally, EA algorithm aims to find helpers that can start to satisfy client's RC as early as possible. In Fig.3a, case (a) and (b), the EA algorithm selects sequentially the early possible list until the requested period is satisfied, i.e. {H1} for (a) and {H2,H3,H4} for (b) respectively. In cases when multiple helpers have tie-breaker on the same initial-access delay, then the set with minimum intermediate delay D is selected, for this case EA would still require the helpers' prediction outcome.

IV. Reactive predictability algorithm

As mentioned earlier, the accuracy of a prediction algorithm depends on individual users, where some users are only 80% predictable while others are up to 93% predictable [4]. In this section we propose the Reactive predictability algorithm (RPA), which characterises predictability on individual bases and hence facilitates helper selection based on different degrees of randominity, typically in terms of a user's association with a number of eNBs in a given region. When a user's randominity is higher, the corresponding predictability using a prediction algorithm is lower. To realise RPA, we first present a simple discrete-time Markov Chain (DTMC) based model to predict users' mobility [10-12]. State transition in DTMC is represented as user transfer from one region to another (in practise this could be a HO from one eNB to another). For the sake of simplicity, this eNB/ small cell coverage area is considered to be in the order of D2D coverage area. In the matrix below we represent DTMC with $t_{i,j}$ representing the probability of a direct transition (i.e., HO) from eNB_i to eNB_j. State transition can be considered to form a learning transition matrix that can be updated dynamically along with time period. Such transition matrix can be formed from users' previous mobility patterns captured by the network.

$$T_p = \begin{pmatrix} t_{1,1} & \cdots & t_{1,n} \\ \vdots & \ddots & \vdots \\ t_{n,1} & \cdots & t_{n,n} \end{pmatrix}$$

With the above time-dependent transition matrix T_p, where p is the period for which it belongs, based on which a simple algorithm can be used to predict the probability of user's location along with time dependency. More details on transition matrix initialisation, properties and learning process are described in [10]. Considering there are N number of eNB in the network with total number of users "U". Assuming that we have a client to be relayed located at eNB_j, with RP Δr and TP Δt. Based on maintained transition matrix T of DTMC, the network could predict a suitable helper set for client in two simple steps. First we find the maximum probability of all users in the network to reach eNB_j in n transitions within Δt period. Secondly we look for the probability of users to stay at eNB_j on arrival.

Prediction algorithm for suitable helpers selection

INPUT:

- CP_i: computed probability of user 'i' to be a suitable helper
- P_i: probability of user 'i' to reach eNB_j in n transitions.
- $T_{j,j}$: defines the probability of user to spend more time in $\text{eNB}_{j,}$ period of transition matric will be for $\boldsymbol{\Delta r}$ duration and more.

On request from a client located at eNB_j along with $\boldsymbol{\Delta t}$ and $\boldsymbol{\Delta r}$

```
DO:
for user "i" from 1 to U
    $P_i$ = Max ($T_{x,j}$ in n transition)
    $CP_i = P_i * T_{j,j}$
End for
```

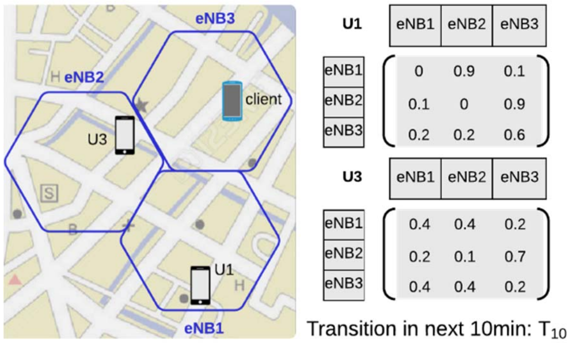
Complete implementation of such algorithm is presented above. The above algorithm is a simple way of computing the probability of various users to reach and spent more time in D2D proximity to a client. Period p in transition matrix selection can be based on Δt, Δr and user density, this decision is to be made by mobile operator. To note our intension here is not to propose an efficient prediction algorithm, but only to take leverage of prediction algorithm and learn user's most recent randominity patterns. For this purpose, any prediction algorithm [5-9] in the literature can be used to provide us with Computed Probability (CP) for user *i* to become a helper (represented by CP_i) along with the n transition list that user is to make. In order to find a suitable helper to relay a client in next Δt period, the algorithm has to consider the most recent prediction failure (recent randominity) in addition to having overall predictability of user. We propose reactive predictability for each user, having already provided with CP_i from existing prediction algorithms as,

$$RP_i = \propto CP_i^c + \beta \pi_i + \gamma \left[Avg. \sum_{x=0}^{T-2} \pi_i^x \right] \qquad (1)$$

$$(\alpha + \beta + \gamma = 1)$$

Where c in CP_i^c represents the current state prediction probability for user *i*, and π_i is represented as $|CP_i^{c-1}|$, which is the success/ failure state information from immediate previous prediction c-1, this can either be 0 or 1. $\propto, \beta$ & γ are three coefficients that is used to provide weights to all three metrics such as CP_i, immediate previous prediction correctness (π_i) and average of all previous correctness of prediction algorithm. On a failure in helper's prediction characteristics, rather than waiting for helper to move to client eventually, RPA will compute and filter out helpers who might not be fit to relay clients by dynamically selecting ones with higher probability of RP_i. Quantitative benefits of RPA will be discussed as part of performance evaluation section. A realistic example is depicted in below Fig. 4, where T for next 10 min period is shown for users 1 and 2. As mentioned earlier SA or eNB range is assumed to be equivalent to D2D coverage range. Assuming client's RC request with Δr 5minutes and Δt 10minutes reaches eNB. From T_{10} in above figure, we can find the value of CP for both users U1 and U2. It can be seen that sum of each row in the matrix will add to one, also for U1, probability of going to eNB3 in two transitions (eNB1 to eNB2 to eNB3) is greater than one transition (eNB1 to eNB3). Hence,

$$CP_1 = (t_{1,2} \text{ x } t_{2,3}) \text{ x } t_{3,3} = 0.9 \text{ x } 0.9 \text{ x } t_{3,3} = 0.81$$
$$CP_2 = (t_{2,3}) \text{ x } t_{3,3} = 0.7 \text{ x } t_{3,3} = 0.7$$



U1	eNB1	eNB2	eNB3
eNB1	0	0.9	0.1
eNB2	0.1	0	0.9
eNB3	0.2	0.2	0.6

U3	eNB1	eNB2	eNB3
eNB1	0.4	0.4	0.2
eNB2	0.2	0.1	0.7
eNB3	0.4	0.4	0.2

Transition in next 10min: T_{10}

Figure 4. Users predictive transition matrix for next 10minutes

Whereas, for CP_2 the probability with one transition is 0.7 while with two transition is 0.2*0.4 = 0.08, hence lesser than one transition. Assuming $t_{3,3}$ is one for both users, based on the above discussed simple prediction algorithm, U1 is a potential helper with predicted route from eNB1 to eNB2 and then to eNB3. As mentioned earlier, here period p for transition matrix selection can be based on Δt, Δr and user density. CP for U1 and U2 along with route, can also be obtained from various other prediction algorithms in literature [5-9]. Based on our proposed RPA, we can compute RP_i for every transition change. When U1 moves from eNB1 to eNB2 as predicted then,

$$RP_1 = 0.4 \text{ x } 0.81 + 0.3 \text{ x } 1 + 0.3 \text{ x } 1 = 0.924$$
$$RP_2 = 0.4 \text{ x } 0.7 + 0.3 \text{ x } 1 + 0.3 \text{ x } 1 = 0.88$$

Hence, U1 is a potential helper, so our proposed RPA's decision for helper selection matches with normal prediction algorithm. In case when U1 doesn't follow the prediction and when U1 moves to a different BS, π_1 will be zero. So RP_1 will be,

$$RP_1 = 0.4 \text{ x } 0.81 + 0.3 \text{ x } 0 + 0.3 \text{ x } 1 = 0.624$$

Now, U2 has higher RP values, so U2 will be potential helper candidate. In this case RPA was more reactive and acted dynamically by considering more recent randominity of user. In performance evaluation section, we present the quantitative benefits of RPA over just using prediction algorithm for helper selection.

V. Practicality Considerations

The realisation of the proposed scheme can effectively leverage on the emerging Mobile Edge Computing (MEC) paradigm [3]. MEC provides IT, storage and computing capabilities within the Radio Access Network in close proximity to users. The MEC server in our case can be deployed at the eNB side. MEC's local resources include location awareness, network measurement based tracking of active (GPS independent and network determined) helpers (as local mobile users), and access to real-time network context data (e.g. dynamic link quality, spectrum availability for D2D communications). Network measurement based tracking/ GPS-based location awareness could be a direct input for prediction algorithms, and other resources in MEC can be used to develop a more intelligent scheme with network condition awareness. In general, D2D communication between a helper and a client can be supported by one of the following three ways: direct WiFi communication, in-band spectrum communication or out-band spectrum communication. In in-band spectrum communication, communication between two devices utilises the same frequency as the current eNB cell operates, whereas in the out-band option, communication between two devices takes place in the frequency assigned explicitly for D2D communications. Details on the usage of the spectrum are outside the scope of this article but will certainly remain as part of our future work. Once receiving the directed resolve message from the eNB/MEC, the helper should be ready to act as relay. All the necessary D2D radio link establishment arrangements are according to 3GPP Proximity Services (ProSe) [2].

On the device energy consumption side, there is no doubt that additional energy needs to be consumed for opportunistically serving data relay on the helper side. Incentives such as higher priority in energy harvesting services for device battery sustainability on the helper side could potentially incentivise mobile uses to participate in serving as helpers. In fact, some existing D2D-based services such as "Karma Go [13]" could potentially address energy issues and incentive mechanisms. In addition, it can be also envisaged that mobile users belonging to different operators may also trade their data allowance based on the D2D partial not-spot coverage, depending on their roles as either clients or helpers at different geographical locations.

Last but not least, security is another concern for direct D2D communication. MEC's embedded security computing features can be used to resolve pre-selected, authorized and authenticated helpers for serving clients. As our approach is network-assisted, other social peer discovery, key exchange and D2D security techniques from the literature can be directly used in our scenario [14]. From data security point of view, client data can be end-to-end encrypted while being relayed by D2D helpers. While D2D communication is foreseen to be an integral part of cellular network [2], implementing such framework will involve very moderate additional deployment costs for network operators, mainly on the installation of software –defined network functions at the eNB side.

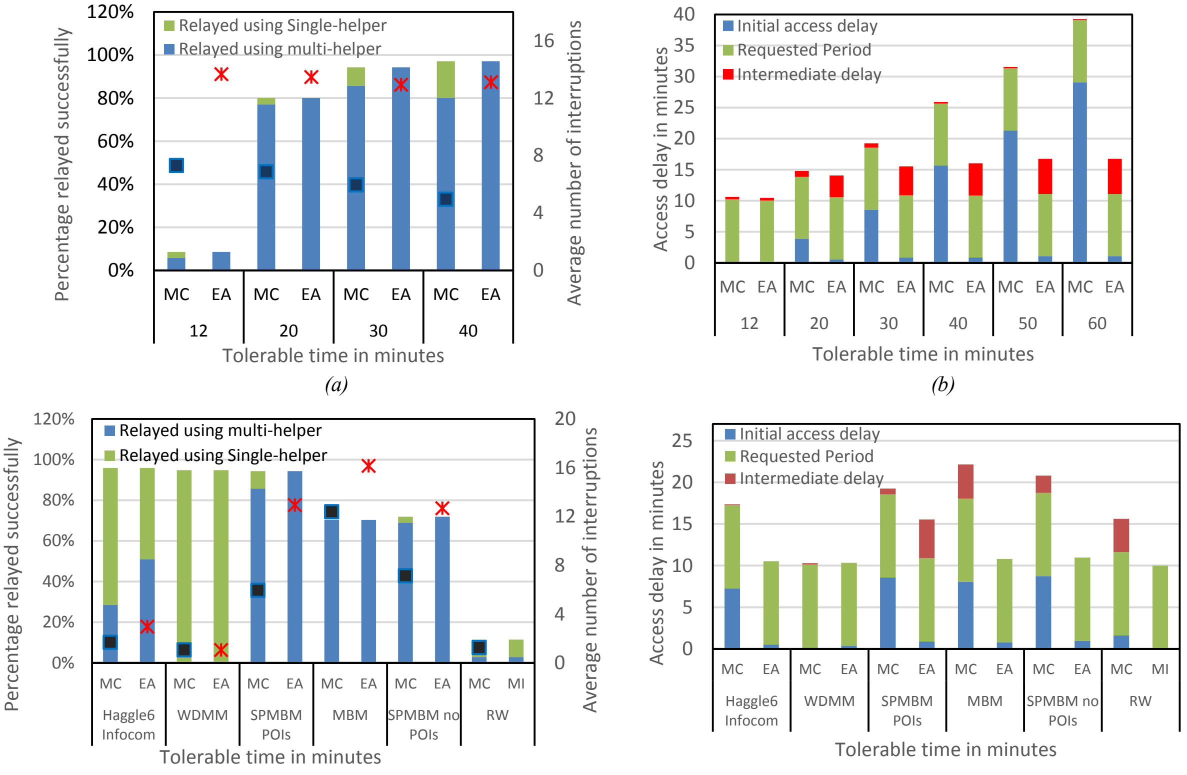
VI. Performance Evaluations

Concerning simulation based experiments, we implemented all the necessary functions in the Opportunistic Network Environment (ONE) simulator [15]. We simulated with 25 mobile users involved in potential D2D relay operations (either as a client or as a helper), and the area of the map was 1000 x 1000 m^2. We used the London Heathrow Airport road map for our performance evaluation. The ratio between clients and local helpers were 1:4 in all default cases. We set the RP value to be 10 minutes and TP value to be 30 minutes in all default cases unless specifically mentioned otherwise. Device communicates with another device using WiFi interface with radio coverage of 30 meters. Our performance metrics used include, percentage of requests relayed successfully using single and/or multiple helpers, initial and intermediate delay, and also average signalling overhead for every request.

Our successful relay statistics metric only captures if RP duration for clients was satisfied by helpers within TP. In the simulation we consider the ideal case where eNB's helper prediction accuracy is assumed to be 100% unless otherwise mentioned, in which case the resulting performance is treated as theoretical upper bound. While we use multiple mobility models and also real human mobility data traces in our experiments, we mainly focus on the shortest path map based model (SPMBM) [16]. Users in SPMBM select a destination from list of point of interest (POIs) on the map and take shortest route to that point. These POIs may be any popular real-world destinations such as restaurants or tourist attractions or simply assembly points, clients are located stationary at these POIs when making request, unless otherwise mentioned.

In Fig.5 (a), we plot the percentage of requests relayed successfully and their corresponding overhead over various TP and ToS. It is interesting to observe that even when TP of 12 minutes, we were able to successfully relay 33% of the requests with RP being 10 minutes. Also the MC algorithm's percentage relay using a single helper increases with TP. This is because the algorithms always aim to identify helpers that incur minimum intermediate delay and helper switchover respectively, this is also reflected by reduced overhead over TP. EA algorithm only concerns about reducing initial access delay regardless the service duration each helper can offer, and it is observed that no request is relayed using a single helper. As depicted in Fig.5 (b), each algorithm leads to distinct service performance characteristics: MC's intermediate delay reduces over increase in TP, and EA algorithm's initial access delay is much less compared to others. User (helper) density is another important metric that influences the service availability and performance. So, five clients were simulated under various helper density and we found that more than 60% of requests were relayed at very low user density (with only 10 helpers-1:2 ratio) in the given area, provided that those clients located at POIs and requests with sufficient TP. In practice, a client can intentionally move close to a PoI (e.g. assembly point at airport) in order to get better opportunity of being served.

We also noticed that mobility pattern of helpers influenced the percentage of successful service received by clients, hence in Fig.5 (c), we plotted over various mobility models. In addition, we also simulated against real data trace obtained from Cambridge/haggle infocom6 project [17], which contains traces of Bluetooth sightings by groups of users carrying small devices (iMotes) for a number of days. For our simulation we extracted 8 hours inter-contact data for 40 mobile devices (used as helpers), along with 10 client devices. Working Day Movement Model (WDMM), which emulates human mobility behaviours on working day basis, is proved to have characteristics such as, inter-contact time and contact time distributions similar to that of realistic real-time mobility traces like, Reality Project of MIT and Dartmouth traces [16]. We also used other models such as SPMBM with POIs and Map Based Movement (MBM) (clients not on POIs, but randomly selected destinations and path) [16]. As depicted in Fig.5 (c), with the most realistic trace such as haggle, percentage of single helper relay is more compared to multiple helper relay, with an overall of 96% successful relay, along with very small overheads, this is because in real-time inter-contact time between users follow power-law distribution [16]. For realistic helper mobility haggle trace and models such as WDMM, the algorithms are able to completely satisfy the client's social application's requests. However, such high percentage of successful relay rate is only to be considered as a theoretical upper bound. In Fig.5 (d), we plot the delay characteristics across various mobility models, this is similar to what was discussed in Fig.5 (c). Hence more realistic the mobility trace higher the relay performance obtained.



(a)

(b)

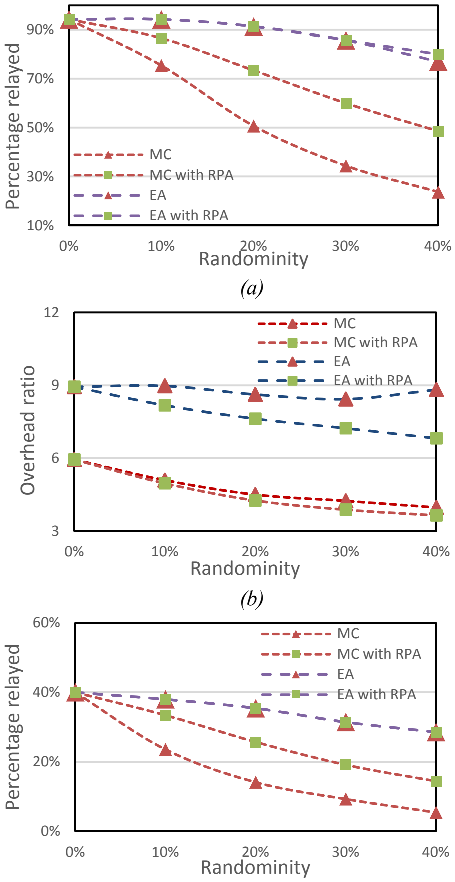
(c)

(d)

Figure 5, Performance, Overheads and Delays introduced by MC and EA over various TP, realistic mobility models and

Until now the accuracy of prediction algorithm was assumed to be 100%, but in practise according to [4] a user can only be 93% predictable. So in Fig.6 (a & b), we plot performance results with various degree of randominity in user prediction. It can be seen that performance of normal prediction algorithm reduces drastically with increase in randominity. This is because the most recent randominity of individual user was not given preference in normal prediction algorithm. Whereas with RPA more than 23% improvement was seen over normal prediction scheme in MC algorithm. EA always had better performance gain compared to MC, but also had very high overhead, this is because EA is opportunistic, and it depends on prediction outcome only during the tie-breaker scenarios. Hence in high randominity case, only a small percentage of request are affected, but still achieving almost comparable success relay percentage over 0% randominity. As can be seen in Fig.6 (b), our RPA tends to reduce the overhead rate (also initial access delay) by selecting right helper set in the beginning, rather than making a selection which is not dynamic enough to satisfy a client in next few minutes, such characteristics of RPA also reduced access delays.

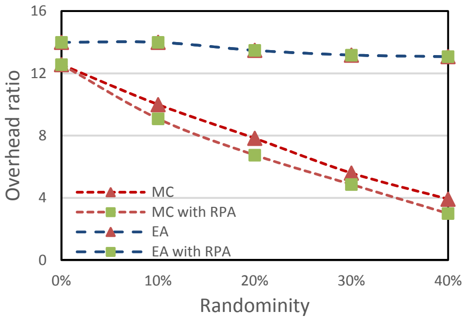
In cases when client in not-spot wants connectivity service while being mobile, the respective eNB could find suitable helper candidate based on two potential methods. First method is opportunistic based, i.e. location of client is updated to eNB based on opportunistic encounter between various helpers in its way. In which case MC cannot play any role, since eNB has no information on client's mobility or location, the relay percentage is completely opportunistic.



(a)

(b)

(c)



(d)

Figure 6, Performance and Overheads introduced over various degree of randominity in prediction algorithm with and without RPA implementation in static (a & b) and mobile (c & d) clients' scenario.

Secondly, the user could input/provide an estimate of its predicted route for the next few minutes of connectivity request. Based on such knowledge the respective eNB could find suitable helpers for various social requirement such as MC or EA. In Fig.6 (c & d) we investigate the performance evaluation with mobile clients under various degree of randominity. Above discussed opportunistic encounter based relaying method will provide the same relay percentage as in EA algorithm. Even when mobile clients' wanting connectivity, it can be seen that the percentage of request relayed reduced by almost 40% from what was obtained with stationary clients. Even with 20% randominity in prediction, RPA was able to successfully relay up to 30% of mobile clients' requests. We noticed that this relay percentage increased with density of users in the network.

VII. Conclusion

To facilitate the efficient use of D2D communication technologies, we have proposed a novel network-assisted D2D-based opportunistic data relay service for users affected in partial not-spots. Exploiting the network's knowledge on its local user mobility patterns, we mainly focused on two distinct algorithms to optimise the service performance with the consideration of specific social application requirements. Through simulations we have shown that, for realistic data trace, up to 96% of requests can be relayed successfully. Furthermore, we also showed the quantified benefits on percentage of relay and overhead ratio of RPA implementation over existing prediction algorithm used in helper selection. Deployment of our scheme will mainly involve software-defined functions and it is fully compatible with services provided from Mobile Edge Computing.

Acknowledgements

This work was supported in part by the EPSRC KCN Project (EP/L026120/1). We also would like to acknowledge the support of the University of Surrey 5GIC (http://www.surrey.ac.uk/5gic) members for this work.

Reference

[1] M. N. Tehrani, M. Uysal and H. Yanikomeroglu. Device-to-device communication in 5G cellular networks: Challenges, solutions, and future directions. Communications Magazine, IEEE 52(5), pp. 86-92. 2014.

[2] 3GPP Release-12, "Feasibility study for Proximity Services (ProSe)", 3rd Generation Partnership Project (3GPP), TR 22.803, 28-06-2013.

[3] 3GPP, "Potential collaboration on mobile-edge computing," TDoc ID: S3-151009, Source: ETSI ISG MEC, 2015.

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

[5] M. C. Gonzalez, C. A. Hidalgo and A. Barabasi. Understanding individual human mobility patterns. Nature 453(7196), pp. 779-782. 2008.

[6] L. Ghouti, T. R. Sheltami and K. S. Alutaibi. Mobility prediction in mobile ad hoc networks using extreme learning machines. Procedia Computer Science 19pp. 305. 2013.

[7] J. A. Torkestani. Mobility prediction in mobile wireless networks. Journal of Network and Computer Applications 35(5), pp. 1633. 2012.

[8] W. Wang and I. F. A. Yildiz. On the estimation of user mobility pattern for location tracking in wireless networks. Presented at Global Telecommunications Conference, 2002. GLOBECOM '02. IEEE. 2002.

[9] J. Chan and A. Seneviratne. A practical user mobility prediction algorithm for supporting adaptive QoS in wireless networks. Presented at Networks, 1999. (ICON '99) Proceedings. IEEE International Conference On. 1999.

[10] A. Mohamed, O. Onireti, S. A. Hoseinitabatabaei, M. Imran, A. Imran and R. Tafazolli. Mobility prediction for handover management in cellular networks with control/data separation. Presented at Communications (ICC), 2015 IEEE International Conference On. 2015.

[11] D. Katsaros and Y. Manolopoulos. Prediction in wireless networks by markov chains. IEEE Wireless Communications 16(2), pp. 56-64. 2009.

[12] N. A. Amirrudin, et al. User's mobility history-based mobility prediction in LTE femtocells network. Presented at RF and Microwave Conference (RFM), IEEE 2013.

[13] Karma Go, "Take WiFi with you on the go,".

[14] Y. Li, et al. Social-aware D2D communications: Qualitative insights and quantitative analysis. Communications Magazine, IEEE 52(6), pp. 150-158. 2014.

[15] A. Keränen, J. Ott and T. Kärkkäinen. The ONE simulator for DTN protocol evaluation. Presented at SIMUTools '09: Proceedings of the 2nd International Conference on Simulation Tools and Techniques. 2009.

[16] F. Ekman, et al. Working day movement model. Presented at MobilityModels '08: Proceeding of the 1st ACM SIGMOBILE Workshop on Mobility Models. 2008.

[17] J. Scott, R. Gass, J. Crowcroft, P. Hui, C. Diot and A. Chaintreau. CRAWDAD dataset cambridge/haggle, 2009.