Evaluating Device-to-Device Content Delivery Potential on a Mobile ISP's Dataset

Leonhard Nobach*, Yannick Le Louédec†, and David Hausheer*
*Peer-to-Peer Systems Engineering Lab, TU Darmstadt
Email: {lnobach,hausheer}@ps.tu-darmstadt.de
†Orange Labs, France Telecom
Email: yannick.lelouedec@orange.com

***Abstract*—Device-to-Device (D2D) content delivery is an emerging approach, where end-user devices exchange content with other end-user devices in communication range, instead of retrieving content from an operator's infrastructure. This way, the operator network can be offloaded from congestion caused by the transmission of popular content, and the content consumer's quality of experience may increase. However, D2D content delivery is only effective in situations where a device in proximity has the requested content available, which is more likely to happen with popular content in crowded areas. The availability of content in communication range of a consumer constitutes an upper bound of the success of a D2D content delivery mechanism, which is referred to as the *potential* of D2D delivery.**

This paper provides a quantitative answer to the question of this potential, and identifies the most important properties a D2D mechanism must provide. An evaluation model is proposed and developed, which can be applied to real-world mobile user traces to determine the quota of content requests that could be served via D2D content delivery. The model is applied on a dataset of a major European Internet service provider and the evaluation results are discussed. The paper concludes that there is potential to deliver up to 60% of requests for popular content via D2D, if a reliable mechanism to predict a user's content consumption is available.

I. Introduction

In today's telecommunication networks, the size of requested on-demand content highly increases. Especially regarding video content, this trend has at least begun with the demand for high-definition (HD), and continues with the increased resolution and frame rate of upcoming *Ultra HD* videos. Because of this traffic increase, the *delivery* of this high-resolution on-demand content to a consumer device more and more becomes a major challenge. Besides the classic client-server approach, Content Delivery Networks (CDNs) are currently the state of the art to deliver *on-demand* content. IP Multicast, which is used by many service providers to broadcast *live* video streams in their network, is often not useful when it comes to delivering on-demand content. Besides, IP Multicast delivery is currently not feasible for over-the-top (OTT) content providers, as multicast techniques were not widely implemented in the public Internet[1].

However, state-of-the-art approaches for content delivery like CDNs only insufficiently reduce load of the access network, still leaving a congestion problem at this network level. Peer-assisted content delivery making use of the infrastructure may even increase the load [1]. Congestion avoidance is especially crucial for mobile infrastructures, as the shared wireless medium may become a bottleneck in situations of increased content requests.

Device-to-Device (D2D) content delivery is an emerging approach, where (commonly mobile) consumer devices like smartphones, tablets or laptops transfer content between themselves, instead of receiving it over a mobile operator's access network. This way, it is targeted to offload infrastructure-based networks from traffic caused by delivering popular on-demand content, thus mitigating congestion and delay issues.

However, D2D delivery has limited effect in certain situations. To retrieve content via D2D communication, a high content popularity is required, as well as a sufficient device density in proximity. Only the presence of both criteria makes it likely that there is a device in proximity serving the content.

First, this paper quantifies the *potential* of D2D content delivery. The potential is defined as the upper bound of the actual delivery success, caused by the fact that there must be a content provider in range. The D2D potential is of high importance to justify and motivate further research and development related to D2D communication. Secondly, an answer to the importance of prediction and caching properties on the success of the D2D delivery mechanism is provided. Thirdly, the implications of how D2D interactions are distributed over time are investigated.

The paper starts with introducing and explaining the proposed evaluation model in Section II. The input data format, the method of obtaining data, the evaluation algorithm and the simulation setup are introduced in Section III. The results that were obtained by applying the proposed model on the given dataset are shown and discussed in Section IV. Before this work is concluded, Section V gives an overview over related studies in mobile networks and relevant work on D2D content delivery.

II. Evaluation Model

The evaluation method that is used in this work applies a model of D2D content delivery on a real-world dataset. The dataset was obtained from a large European mobile Internet service provider, comprised of anonymized cell association data of mobile devices. Because of privacy concerns, the user's

[1] An exception is the *MBone* overlay network

978-3-901882-77-7

content requests were not used, instead, a content consumption model was applied.

Given position traces of mobile device users, we evaluate the quota of successful D2D deliveries like if the users in the trace data had used D2D content delivery in reality, by applying a valid model of a D2D delivery mechanism. This section explains and discusses the model, after stating and justifying several assumptions. These assumptions are deduced from plans for further research on D2D delivery.

Assumption 1: We use a one-hop delivery model. This means a device can retrieve content only from devices in direct communication range which have this content already stored for own purposes. There are no selfless or artificially rewarded intermediate devices that relay content for others. However, our model allows selfless caching after content consumption to be able to provide content to direct neighbors afterwards.

Assumption 2: We assume a "*perfect*" prediction mechanism for content consumption, however, bounded by a maximum possible prediction duration t_p. The mechanism is perfect in the sense that for every content request at time t, the consumption can be predicted by the mechanism in the interval of $[t - t_p, t]$, and, for every point in time t where a content consumption is predicted, the content will be eventually requested between $[t, t + t_p]$. Informally, every content request which is at most t_p in the future is predicted, and there are no false predictions. Such a perfect prediction mechanism can only be designed for e.g. regular updates, but not for content consumption triggered by user behavior. However, the goal of this work is to evaluate the potential of D2D, while leaving the problem of content consumption prediction for further research. Prediction mechanisms used for prefetching [2] [3] may be candidates for usage in D2D content delivery mechanisms as well.

Assumption 3: We assume two devices are in D2D communication range to each other, if they are in the same cell. On the one hand, according to the Hidden Node Problem, the assumption does not always hold. However, it can be shown that for equally-distributed devices in an area, the effect of this problem is approximately compensated by the number of devices which are in communication range to each other, but in different cells. On the other hand, in the last sentence, we assumed approximately equal communication ranges of all physical links. But in reality, the range of a mobile base station may be larger compared to the communication range of our physical layer used for D2D content delivery, e.g. IEEE 802.11 (WiFi). However, upcoming technologies like LTE Direct may provide D2D communication ranges up to 500 meters [4], comparable to urban cell sizes [5] [6].

A. D2D Content Delivery Model

As mentioned, our model assumes a prediction mechanism which can tell if the user consumes a certain content item (video/audio file, application) in the future. In our model, a prediction can be made at most t_p before the time of the content consumption t_{req} (Figure 1 ①). This time interval is called the *prediction phase*. After t_{req}, the content remains

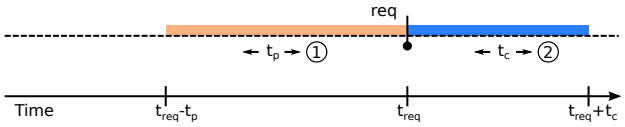

Fig. 1. Illustration of a request in our model, with a given prediction duration and cache lifetime. By default, a request can be predicted for t_p (orange/light bar), and the request is cached for t_c (blue/dark bar).

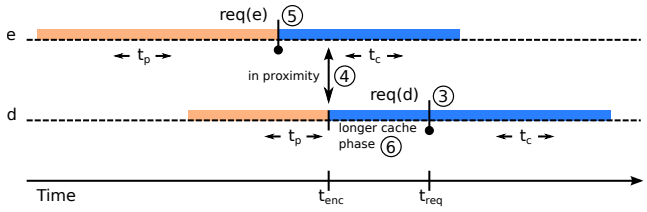

Fig. 2. Illustration of a successful D2D content delivery for Device d, based on our model.

in the device's cache for t_c until it becomes unavailable for others (the *caching phase* ②).

In our model, a request of a device for content is the point in time where the user wants to actually consume the content, i.e. when it is needed. Given a device d making a request $req(d)$ for a content item at time $t_{req(d)}$ (Figure 2 ③). If, at any time[2] t_{enc} in the prediction phase of $req(d)$, a device e was in communication range (proximity) of d ④ which requested the same content ⑤ and is in the caching phase of the request for this content at the time t_{enc}, then:

- A D2D content delivery for the request $req(d)$ is assumed to be successful.
- The caching phase for $req(d)$ starts earlier ⑥; it does not start at $t_{req(d)}$ but at t_{enc}. This is because in a successful D2D transfer, the content is assumed to be transferred at the first time the devices are in proximity, and remains in the cache until requested (and longer). If the above condition is fulfilled by multiple devices e or at multiple points in time t_{enc}, the caching phase starts at the earliest t_{enc} (the D2D mechanism exploits the first opportunity to obtain content).

Otherwise, the D2D transfer is assumed to be not successful, as there is no device in range having the content cached. The caching phase then still starts at the time of request, as in this case the content is assumed to be retrieved on-demand via infrastructure. In either case, the content is stored for t_c after the request.

Given this evaluation model, several cache times can be evaluated per successful D2D request, as illustrated in Figure 3. The **Provider Cache Time (PCT)** is the time the content has spent in the providing device's cache until the exchange happens. The **Altruistic PCT (APCT)** is the time from the providing device's own content request until the exchange.

[2] t_{enc} may be informally described as a point in time where d "encounters" (is in communication range of) another device, which may have or may not have the content

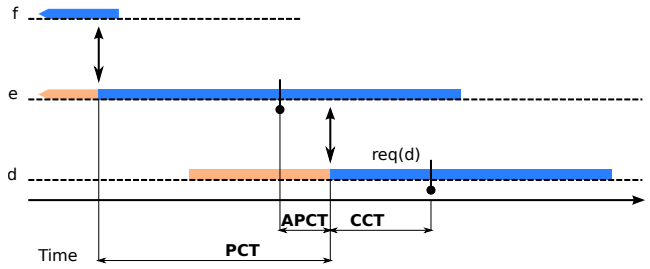

Fig. 3. Illustration of several time definitions of a content request $req(d)$ which could be successfully served via D2D.

If the exchange happens *before* the provider's own content request, the APCT is defined as 0. The **Consumer Cache Time (CCT)** ranges from the time of the D2D exchange to d's content request $req(d)$.

B. Proximity Model

As already mentioned, it is assumed that devices are in communication range, if they are in the same mobile cell. This is an approximation to real-world scenarios, as two users in the same cell may be in range of the cell base station, but not in range to each other. This assumption was justified in Section II. No exact positioning (e.g. GPS) is used, as it is too privacy-sensitive, hard to anonymize, and hard to obtain.

In the input data explained in Section III-B, no continuous join/leave information is given, instead the position dataset provides location samples upon several events. Because of this and due to evaluation complexity considerations, the proximity model is sampled into slots of $q = 10min$. As a consequence, in our model, we assume two devices are in communication range, if they are in the same cell during the same time slot.

III. Input Data and Evaluation Setup

The evaluation conducted in this paper on the one hand makes use of on *real-world position data*, which were obtained from tracing users of a large mobile network operator; on the other hand it is based on a *content consumption model* that is applied on these users. The evaluation component was initially planned and is capable of running on real-world content consumption data, however the collection of these would require deep-packet inspection and would arise serious privacy issues, not to mention necessary difficult anonymization efforts.

A. Required Position Data Format

The position samples are based on a user's cell associations. At frequent time intervals, the data set needs to contain information for every user being associated to a specific cell. In a database or CSV format, it is comprised of the following fields:

- The current time. Must be accurate, can be relative (e.g. 0 = start of experiment).
- A user ID. The ID should be anonymized by using e.g. any random unique number for a user. The user ID should not change during the entire experiment for the user.
- A cell ID. The ID uniquely identifies a mobile base station. The cell ID should also be anonymized like the user ID, so that no geographical information can be deduced.

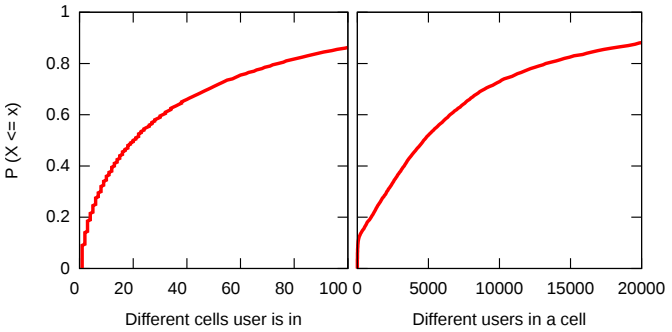

Fig. 4. CDF of the number of different cells a user is in, and the number of different users which are in a cell during the whole measurement.

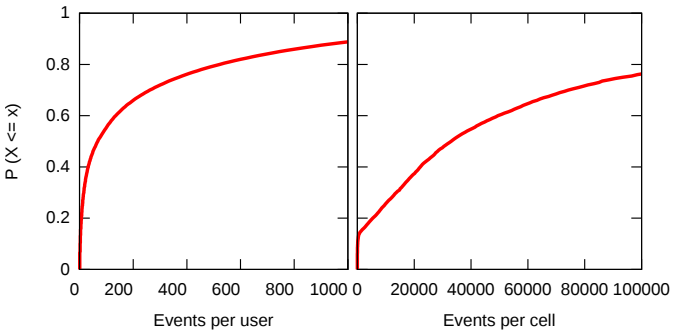

Fig. 5. CDF of the total number of events related to a user and cell during the whole experiment.

B. Position Data Used

The input position data of mobile devices were collected in one of the larger cities in the south of France. The data collection started on March 4th and ended on March 19th 2015 (0:00h), thus lasting for 15 days. The dataset was comprised of ca. 450 million anonymous cell UID samples of 1.2 million anonymized users.

Figure 4 summarizes the number of different cells a user has joined and left, as well as the number of different users every cell contains, both during the whole data collection period. On the one hand, it can be seen that half of the users were in 20 different cells, and that 20% of the users joined and left 100 different cells or more. On the other hand, half of the mobile cells served for more than 5000 different mobile users during the experiment.

For every device in this dataset, a sample was generated periodically every 3 hours, whenever there is a handover during a connection, as well as upon connecting to the network. Figure 5 depicts the number of events per user or cell. We can see that a large part of the users only generated a small set of samples, much less than even a periodic location update would produce during the measurement period, like device users which are

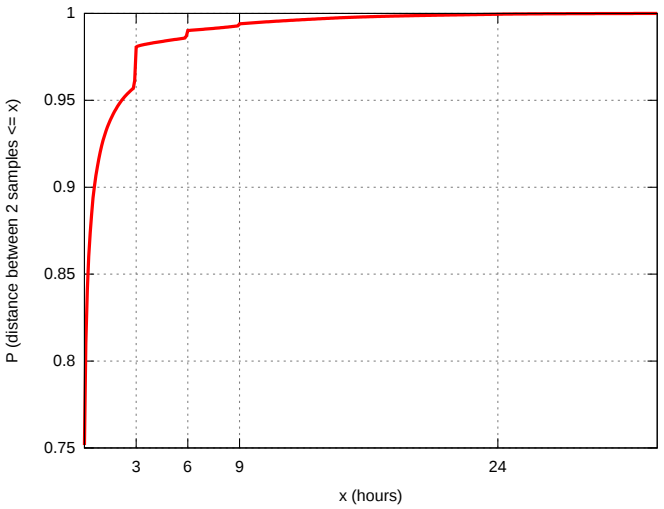

Fig. 6. CDF of the distances between successive samples related to the same user.

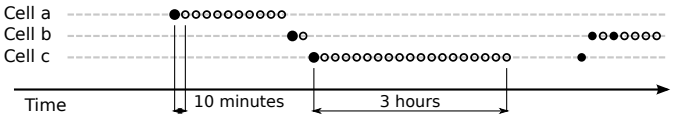

Fig. 7. Artificial samples (non-filled circles) are created after measured ones (filled circles). Example of a device changing between three cells.

visible only a short time. As the measurement only covers a small area of France, this may be caused by commuters entering or leaving, or transit traffic passing through this area (cars, trains).

In the cumulative distribution function in Figure 6, which depicts the the distances between two successive events of the same user, we can see the effect of the periodic location update. This effect is visible at 3h (6h and 9h, if an intermediate location event was missing, e.g. due to a disconnection). However, most events are shorter apart than the periodic location update (more than 95%).

When an idle device does not change cells, it would be assumed offline by the proximity model for up to $3h - q$, although it is just idle. To fix this issue, upon a sample, we added artifical samples in the same cell for 3 additional hours (equidistant, with a distance of $q = 10min$ between them), but not longer than a possible following original sample (Figure 7).

C. Evaluation Algorithm

As there are no interdependencies between distinct content items in the proposed model, Algorithm 1 is executed separately for every item. It is a simplified but essential version of the algorithm used by the software developed for this evaluation. It works on a queue Q of request objects (containing all requests of all devices for the particular content item). A request object $r = (t_r, t_b, d, p)$ consists of a parameter t_r which is the time of the request, and a variable parameter t_b, which resembles the time the content item starts to be in a device's cache. Initially, $t_b = t_r$. The queue Q is **always ordered** by the variable t_b (ascending), this means it may be reordered if it changes. d is a Boolean parameter which expresses if this request could be served via D2D. It is initially false, p is the device that does the request. The function $prox(t1, t2)$: returns a set of tuples (p, t) with all devices p in the same cell between time t_1, and t_2, according to our proximity model, and the corresponding time t of proximity to device d.

Result: Quota of requests that can be served via D2D
d2d_requests = 0;
total_requests = 0;
while *Q not empty* **do**
  Get first element $r = (t_r, t_b, d, p)$ from Q;
  for *all devices* (p_{enc}, t_{enc}) *in* $prox(t_b, t_r + t_c)$ **do**
    if *there is an element* $r_2 = (t_{r2}, t_{b2}, d_2, p_2)$ *in* Q, *where* $p_2 = p_{enc}$ *and* $t_{r2} - t_p < t_{enc} < t_{r2}$ **then**
      $d_2 = true$ (Mark request as D2D-served);
      $t_{b2} = t_{enc}$ (set t_b to time of proximity);
      (Reorder Q, if necessary);
    end
  end
  if $d = true$ **then**
    d2d_requests++;
  end
  total_requests++;
  Remove first element from Q;
end
quota = d2d_requests / total_requests;

Algorithm 1: Simplified evaluation algorithm, returning the quota of D2D requests that could be served via D2D content delivery

The result is the quota of possible requests that could be served via D2D communication.

D. Content Consumption Model

As mentioned, because of privacy concerns, the real content consumption behavior of the users was not evaluated; instead, two different content consumption models were applied, a *linear* one and a *Weibull*-shaped one (Figure 8). For every user in the traces, there is a probability p_{req} to access the content during the evaluation period *at all*. These users then consume content at randomly-distributed points in time:

According to the *linear* request model, the randomly selected devices request the content between March 5th and March 17th once. For each of these devices, the actual probability of the request is equally distributed inside the aforementioned time interval. The applied consumption model especially describes the delivery of e.g. a software update for a mobile platform like iOS or Android, but may be valid for other consumption behavior, as well. For example, Chowdhury et al. [7] find that 50-60% of video popularity peaks at the first

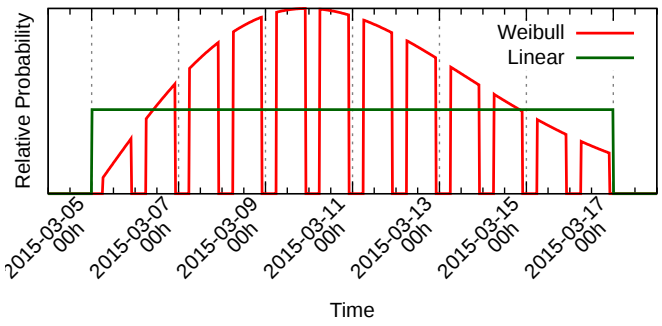

Fig. 8. The *Weibull* and *Linear* consumption model's probability density functions.

day. However, they also find that peak popularity may occur later, especially for certain video categories.

To better resemble a smooth increase and decay of content popularity as well, we also consider a Weibull-shaped request model. Its origin is on March 5th at 00:00h, and furthermore follows a Weibull distribution with $\lambda = 7$ days and a form parameter $k = 2$. These parameters were chosen to provide a moderately increasing popularity, which slowly decreases after its maximum, and is very small at the end of the experiment. As an exception, the Weibull request probability between 22:00h and 09:00h of each day is zero (which shall resemble the lack of requests at night) as well as the probability to request content after March 17th, 00:00h (as the position data ends one day later).

E. Simulation Implementation

The evaluation algorithm was implemented on the Java platform and language. Java Collections and the GNU Trove library were used to reduce complexity in map, list and queue operations, but especially to save memory. The simulation server was based on Ubuntu with 96GB of RAM. The evaluation software was run in the Oracle JRE. An evaluation run with 30 different sample sets of both content consumption models, as well as all parameter variations, required around 12 days of simulation time.

IV. Results

Figure 9, 10 and 11 show the quota of the requests which could be served via D2D content delivery (further called **D2D quota**)[3]. First of all, it can be seen that for all values of p_{req} used, a fundamental ratio of requests can be served via D2D in general (around 10%-60%). The prediction duration t_p is found as the parameter with greatest influence on D2D delivery success.

In Figure 9, the D2D quota is depicted in relation to the prediction duration. Various curves are shown for all request probabilities p_{req} and content consumption models used. It can be observed that the prediction duration (t_p) has a significant, positive influence on the D2D quota, as the quota grows

[3]In Figure 9, 10, 11, 13 and 14, 95% confidence intervals are depicted, although they are relatively small.

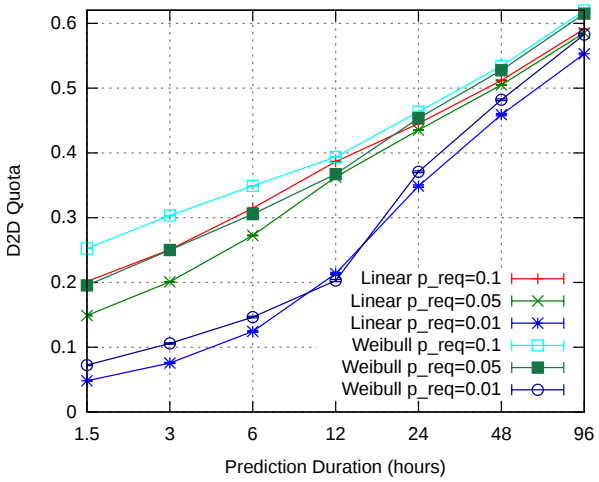

Fig. 9. The development of the *D2D quota* under influence of the *prediction duration* (t_p). Depicted are multiple curves for different p_{req}s and consumption models.

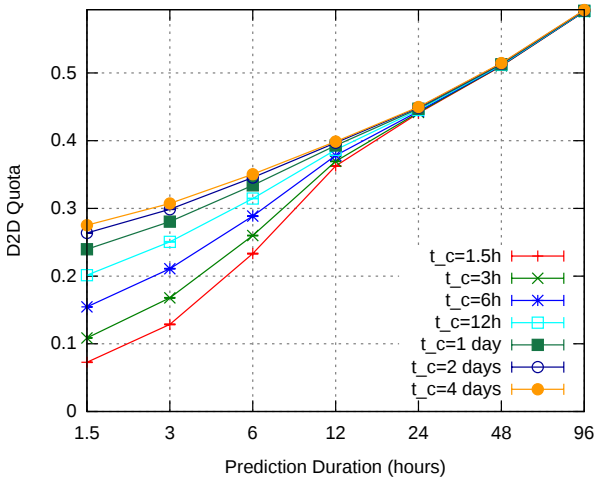

Fig. 10. The D2D quota for $p_{req} = 0.1$ and the linear content consumption model. For different cache lifetimes t_c the values converge with increasing prediction duration t_p.

logarithmically with t_p. However, there is an anomaly resulting in a significantly decreased D2D success during $t_p \leq 12$ compared to the rest of the measured time scale. This *12-hour anomaly* could be caused by daily commuting cycles, device users being absent from crowds over night, or users powering off the devices when sleeping. This hypothesis is examined in Section IV-A.

Figure 10 also depicts the D2D quota under different prediction duration, but with curves for different cache lifetime parameters (t_c). We observe that a high cache lifetime supports the D2D success if only a short prediction duration is present. However, with increasing prediction duration the cache lifetime loses importance. This relation between t_p and t_c is also shown in Figure 11, where the increase of D2D success with higher cache lifetime is only relevant with $t_p \leq 12h$.

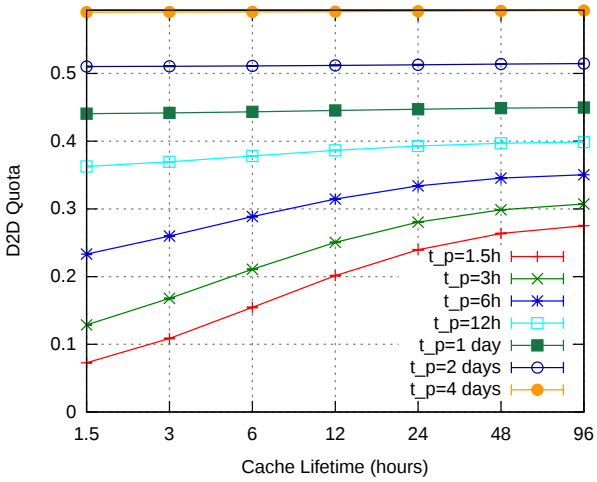

Fig. 11. The D2D quota development under different cache lifetime t_c. A relevant influence can only be observed for low prediction durations (t_p).

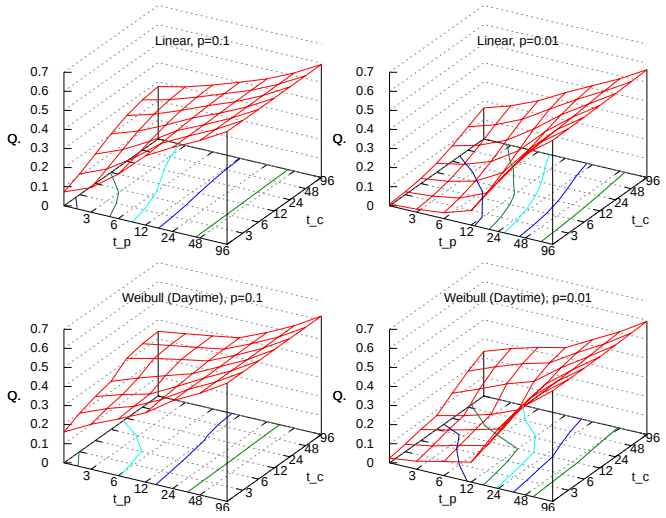

Fig. 12. The D2D quota under different values of *prediction duration* and *cache lifetime* (3-dimensional). For different p_{req} and consumption models.

For better understanding of the mentioned 12-hour anomaly, Figure 12 shows four three-dimensional plots for different request probabilities and distribution models. They depict the D2D quota under different parameters of both prediction duration and cache lifetime. Confidence intervals are not plotted for better visibility, but they appear to be equal or similar to the ones in the 2D graphs. It can be observed that either a prediction duration or a cache lifetime greater than 12 hours is required to bridge the "gap" introduced by the 12-hour anomaly. The cause for it will be discussed in Section IV-A.

A first finding is that if we have a mechanism available being able to predict at least the next 12 hours, we therefore may reduce the cache lifetime to a minimum or may even switch off further caching after content consumption. However, if such a prediction mechanism is not available, we should enable long (altruistic) caching instead (and for this, we require incentives for the user).

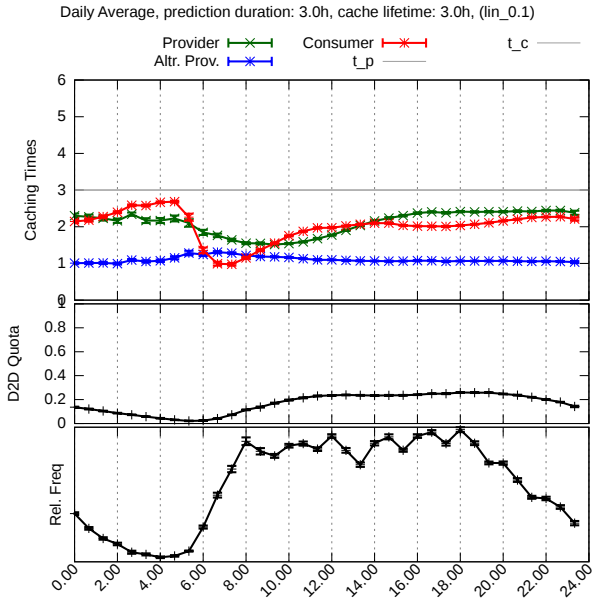

Fig. 13. Average values of cache times and D2D quota of requests made, as well as the relative number of D2D exchanges that occurred at specified time of the day. Linear request model, request probability $p = 0.1$, $t_c = t_p = 3h$

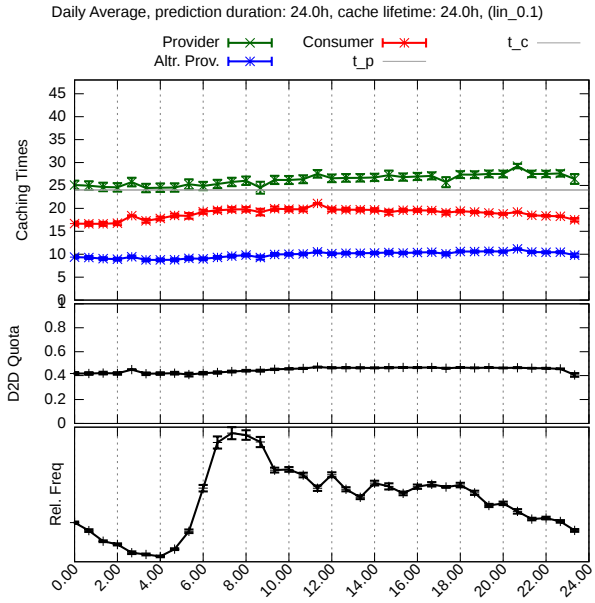

Fig. 14. Like in Figure 13, but with different parameters $t_c = t_p = 24h$

A. *Daytime-Dependent Results*

In contrast to the previous section which presented and discussed summary results, this one is about how the observed values evolve over the day. In Figure 13 and 14, various average values for every time of day are plotted. In both figures, the top graphs show the *provider cache time (PCT)*, the *altruistic provider cache time (APCT)* and the *consumer cache*

time (CCT). These *average* values are explained in Chapter II. and should not be confused with the *parameters* prediction duration and cache lifetime. The middle graph shows the daytime-dependent D2D quota, while the bottom graph depicts the relative frequency of D2D exchanges. The latter expresses the frequency or density of actual D2D transmissions taking place at this time.

In Figure 13 ($t_p = t_c = 3h$), we can see that the D2D quota significantly drops at night time. Here, neither the prediction can look more than 3 hours ahead, nor can the cache keep the content longer than 3 hours afterwards. Between 06:00 and 08:00, the D2D quota increases again. The average consumer cache time drops in the same time range, which means most of the content requested was exchanged only a short time ago, likely during an increased commuting in the morning. Because of the small prediction duration and cache lifetime, this content could not be exchanged the day before.

The overall D2D quota of Figure 13 is not as good as it is in Figure 14, where the prediction duration and cache lifetime was set to 24 hours instead. Here, the overall D2D quota more than doubles, and is especially better over night, caused by the possibility to fetch content the day before. Caching times do not significantly differ during the day. It can also be observed that with increased prediction and caching deadlines, more D2D exchanges (bottom graph) tend to happen in the morning. This is caused by a *greedy* content exchange behavior: Given that the consumption is predicted, every device tries to use the first chance to exchange content with another device having the content. This may lead to peaks at the beginning of crowded events, or at the start of phases in which a device user is more among crowds (like the start and end of the workday). Problems arising with that are investigated in Section IV-B.

B. Absolute, Time-Dependent Results

In Figure 15 and Figure 16, we can see an overview of measured parameters over a whole, single simulation run. The differences between the simulation runs are the selected prediction duration and cache lifetime parameters. While the prediction duration *and* the cache lifetime are both set to 3 hours in Figure 15, they are set to 48h in Figure 16. In the latter figure, the long-term behavior is important. On the one hand it can be seen that, especially on the first day in the morning after the content becomes popular, there is a peak of D2D content exchanges. Again, the long prediction time enables the devices to take the first chance to exchange content which is then consumed during the next two days. On the other hand, we can observe fewer exchanges during the last two days of the experiment. Here, many devices do not require subsequent exchanges, as the content was already fetched the days before.

A finding here is that a long prediction time can deteriorate the positive effect of smooth (i.e. approximately equally distributed) consumption models on spectrum usage, by shifting D2D exchanges at the start of content popularity periods. However, if the spectrum does not allow the required amount of D2D exchanges at this time and requests become denied, they may as well be conducted at a later point in time.

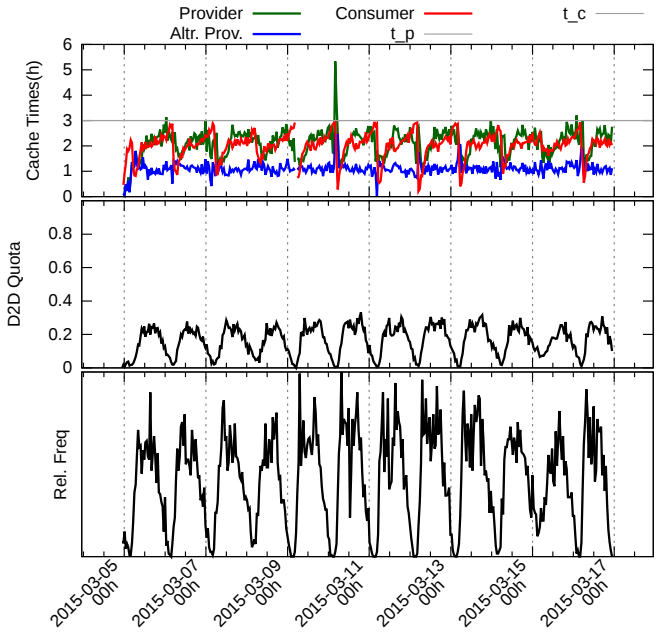

Fig. 15. Average cache times and D2D quota of requests made at the specified timeslot, as well as the relative number of D2D exchanges at this timeslot. Linear request model, request probability $p = 0.1$, $t_c = t_p = 3h$

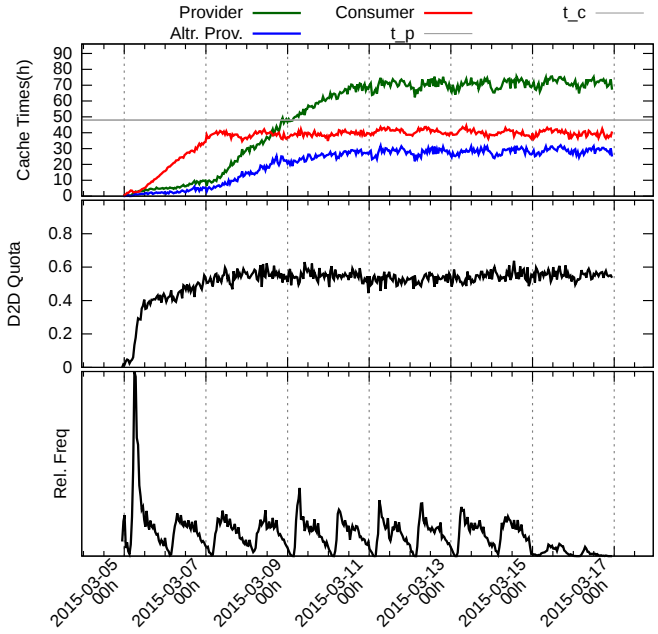

Fig. 16. Like in Figure 15, but with different parameters $t_c = t_p = 48h$

V. Related Work

The idea to offload mobile carrier networks via device-to-device content delivery was previously described [8], [9]. Here, decentralized discovery strategies are not used, instead, a central operator mediates between communication partners in range. The reason for a decision against a decentralized discovery has been to avoid periodic scanning for communication partners, consuming energy, and thus shortening battery lifetime of mobile devices [8].

Our previous work also proposed an approach towards content delivery based on a one-hop transmission [10]. The contribution of this work was the sketch of a decentralized discovery method, while avoiding energy and privacy issues through a slight modification of the 802.11 MAC layer. The question about D2D potential *in general* was asked, but left for future work. The present paper now provides a quantitative answer, while using the previously proposed mechanism as the basis for the simulation model described in Section II.

Another attempt to evaluate opportunistic communication with real-world measurements was conducted by Han et al. [11]. Here, content is delivered by the mobile base station to a small subset of mobile subscribers (the *target set*) in a first step, while these nodes propagate the information to the entire set of interested nodes afterwards. Their work focuses on the selection of an optimal aforementioned target set. Han et al. put an increased focus on evaluating the selection strategies using mobility traces. Unlike in our work, the traces were obtained from restricted groups, like the Haggle [12] and the Reality Mining project [13]. Besides the smaller user base of the traces obtained by both projects, a major part of the devices participating has likely belonged to students and university staff which took part in the study. The devices have actively been collecting mobility trace data. The authors also discuss available incentives to participate in the target set.

A data collection method very similar to the one used in this paper was used by Shafiq et al. [14]. The anonymized dataset contains usage data of 100 million customers of a large mobile ISP of the United States. The data was collected at the GGSN nodes, at the time of various highly crowded events in metropolitan areas. However, the work focused on evaluating radio resource allocation and opportunistic *connection* sharing rather than D2D content delivery, requiring a different set of metadata, such as performance-related information.

Jiang et al. [15] conduct a model-based evaluation on mobile data traces, showing how it can be used for the purpose of urban planning. Contrary to our work, information about the signal strength is included in the data set, and can be used to obtain a more fine-grained position of the devices. However, this method is more privacy-sensitive. Hei et al [16] conduct a measurement study of PPLive, an IPTV software popular in Asia. The measurement study is focused on evaluating on the one hand the user behavior, on the other hand the behavior of technical properties of the P2P network.

Prediction mechanisms for content consumption were investigated by Zhao et al. [2]. The mechanism's prediction is based on a user's information in social media streams. The mechanisms are used in conjunction with an approach that prefetches content when there is improved Internet connectivity over an infrastructure. D2D transmission techniques were not used. Based on this work, Do et al. [3] refined the approach and achieved a prediction accuracy of 72% in average.

VI. Conclusion and Future Work

Given the proposed model, there is a significant potential of D2D content delivery to offload a mobile operator's network. The potential ranges from 10% to 60% of traffic which can be offloaded to the D2D medium, depending on the parameters of the model. We also find that the most influential parameter of the D2D content delivery model is the *prediction duration* which can be offered. Especially a reliable mechanism that is able to predict a user's content consumption for more than 12 hours in advance will lead to high success of D2D content delivery.

The longer a prediction mechanism can forecast the content consumption and the more reliable it is, the less relevant an altruistic caching becomes after content consumption. However, if our prediction is only able to forecast the consumption for less than 12 hours, every device should cache the content for a longer period of time after consumption, to be able to provide the content to others.

Another finding here is that a long prediction time can deteriorate the positive effect of smooth, temporally well-distributed consumption behavior on spectrum usage, by shifting content exchanges to the start of content popularity periods, or daytimes. The resulting peaks of D2D exchanges may lead to congestion, as the spectrum used for D2D exchange only allows a limited amount of simultaneous D2D deliveries at a time. In this case, several requests become denied and have to be conducted at a later point in time. The proposed model does not include congestion behavior, but it nevertheless shows up the importance of congestion control and spectrum management in D2D content delivery research.

The actual delivery success likely remains below the potential, depending on the success of prediction mechanisms, caching strategies, and the absence of transmission errors. In future work, the proposed model may be refined. Different prediction mechanisms that exist or that are yet to be discovered may be integrated into the model. The evaluation did not yet make use of real content consumption data due to privacy concerns. However, the proposed model supports this type of data; a privacy-preserving method to make use of them would further support the evaluation of D2D potential. Finally, cells may be classified to return results for different localities (urban, rural).

Acknowledgments

This work has been supported in parts by the European Union (FP7/#317846, SmartenIT and FP7/#318398, eCousin). The authors would like to acknowledge valuable comments by their colleagues and project partners, especially from Ali Gouta, who helped the authors in early evaluation runs.

REFERENCES

[1] T. Karagiannis, P. Rodriguez, and K. Papagiannaki, "Should internet service providers fear peer-assisted content distribution?" in *ACM SIGCOMM IMC*, No. 5, 2005, pp. 6–6.

[2] Y. Zhao, N. Do, S.-T. Wang, C.-H. Hsu, and N. Venkatasubramanian, "O2SM: Enabling efficient offline access to online social media and social networks," in *Middleware 2013*. Springer, 2013, pp. 445–465.

[3] N. Do, Y. Zhao, S.-T. Wang, C.-H. Hsu, and N. Venkatasubramanian, "Optimizing offline access to social network content on mobile devices," in *IEEE INFOCOM*, 2014, pp. 1950–1958.

[4] Qualcomm, "Creating a digital 6th sense with LTE Direct - LTE Direct overview presentation," 2015. https://www.qualcomm.com/documents/creating-digital-6th-sense-lte-direct

[5] T. Rappaport, S. Sun, R. Mayzus, H. Zhao, Y. Azar, K. Wang, G. Wong, J. Schulz, M. Samimi, and F. Gutierrez, "Millimeter wave mobile communications for 5G cellular: It will work!" *IEEE Access*, Vol. 1, pp. 335–349, 2013.

[6] E. Dahlman, S. Parkvall, and J. Skold, *4G: LTE/LTE-advanced for mobile broadband*. Academic Press, 2013.

[7] S. A. Chowdhury and D. J. Makaroff, "Popularity growth patterns of youtube videos-a category-based study," in *WEBIST*, 2013, pp. 233–242.

[8] X. Bao, Y. Lin, U. Lee, I. Rimac, and R. R. Choudhury, "DataSpotting: Exploiting Naturally Clustered Mobile Devices to Offload Cellular Traffic," in *IEEE INFOCOM*, 2013, pp. 420–424.

[9] N. Golrezaei, A. Dimakis, A. Molisch, and G. Caire, "Wireless Video Content Delivery through Distributed Caching and Peer-to-Peer Gossiping," in *ASILOMAR*, 2011, pp. 1177–1180.

[10] L. Nobach and D. Hausheer, "Towards decentralized, energy-and privacy-aware device-to-device content delivery," in *Monitoring and Securing Virtualized Networks and Services*. Springer, 2014, pp. 128–132.

[11] B. Han, P. Hui, V. A. Kumar, M. V. Marathe, G. Pei, and A. Srinivasan, "Cellular traffic offloading through opportunistic communications: A case study," in *ACM MobiCom CHANTS*, 2010, pp. 31–38.

[12] A. Chaintreau, P. Hui, J. Crowcroft, C. Diot, R. Gass, and J. Scott, "Impact of human mobility on opportunistic forwarding algorithms," *IEEE Transactions on Mobile Computing (TMC)*, Vol. 6, No. 6, pp. 606–620, 2007.

[13] N. Eagle, A. S. Pentland, and D. Lazer, "Inferring friendship network structure by using mobile phone data," *Proceedings of the National Academy of Sciences*, Vol. 106, No. 36, pp. 15 274–15 278, 2009.

[14] M. Z. Shafiq, L. Ji, A. X. Liu, J. Pang, S. Venkataraman, and J. Wang, "A first look at cellular network performance during crowded events," in *ACM SIGMETRICS Performance Evaluation Review*, Vol. 41, No. 1, 2013, pp. 17–28.

[15] S. Jiang, G. A. Fiore, Y. Yang, J. Ferreira Jr, E. Frazzoli, and M. C. González, "A review of urban computing for mobile phone traces: Current methods, challenges and opportunities," in *ACM SIGKDD UrbComp*, No. 2, 2013, p. 2.

[16] X. Hei, C. Liang, J. Liang, Y. Liu, and K. W. Ross, "A measurement study of a large-scale P2P IPTV system," *IEEE Transactions on Multimedia*, Vol. 9, No. 8, pp. 1672–1687, 2007.