Enhancing Mobile Data Offloading with Mobility Prediction and Prefetching

Vasilios A. Siris — *vsiris@aueb.gr*
Dimitrios Kalyvas — *kalyvas.d@gmail.com*
Mobile Multimedia Laboratory, Department of Informatics
Athens University of Economics and Business, Greece

We present procedures that exploit mobility prediction and prefetching to enhance offloading of traffic from mobile networks to WiFi hotspots, for both delay tolerant and delay sensitive traffic. We evaluate the procedures in terms of the percentage of offloaded traffic, the data transfer delay, and the cache size used for prefetching. The evaluation considers empirical measurements and shows how various parameters influence the performance of the procedures, and their robustness to time and throughput estimation errors.

I. Introduction

Over the last few years, a major trend in mobile networks is the exponential increase of powerful yet affordable personal mobile devices, such as smartphones and tablets, with multiple heterogeneous wireless interfaces that include 3G/4G/LTE and WiFi. The proliferation of such devices has resulted in a skyrocketing growth of mobile traffic, which in 2011 grew 2.3-fold, more than doubling for the fourth year in a row, and is expected to grow 18 times from 2011 until 2016[1]. On the flipside, despite its increase, the mobile data revenue significantly lags behind the exponential growth of data traffic.

One solution to address the strain from the exponential growth of mobile data traffic is network expansion and deployment of 4G/LTE technology. However, such a solution is both costly and time consuming. A second solution is to expand the mobile access network using femtocells, which exploit the existing backhaul infrastructure (cable, xDSL). Two issues with femtocells are the interference between femtocells and an operator's metro cells and the cost of the customer premises devices.

Another alternative is to move a portion of the mobile data traffic to WiFi networks, exploiting the significantly lower cost of WiFi technology and existing backhaul infrastructure. WiFi offloading can reduce pressure on the most costly component of the mobile network, the RAN (Radio Access Network) which accounts for 70-80% of the total cost for a mobile network, with the rest of the costs going to the mobile backhaul/core. The industry has already verified the significance of WiFi offloading by initiatives of large mobile operators: Since May 2010, AT&T has been deploying WiFi access points in areas with consistently high mobile data traffic. Other major operators including Verizon and T-Mobile are also increasing WiFi usage to offload mobile data traffic.

The goal of this paper is to propose and evaluate procedures that exploit mobility prediction combined with WiFi and mobile throughput prediction, along with data prefetching to enhance offloading of mobile data to WiFi hotspots. Mobility prediction can provide information on a vehicle's route and when the vehicle will reach different locations along its route. Such mobility information can be combined with geolocation information regarding WiFi hotspot access and WiFi/mobile throughput, to predict the number of WiFi hotspots that the vehicle will encounter, the duration of access and estimated throughput for each hotspot, and the estimated mobile throughput in route segments where it will have only mobile access. In summary, our contributions are the following:

- We develop procedures that exploit mobility prediction for delay tolerant traffic to minimize the data transfer throughput over the mobile network, hence increase the amount of offloaded data traffic.
- We develop procedures for both delay tolerant and delay sensitive traffic that exploit data prefetching, i.e., proactive caching of data in WiFi hotspots that the vehicle will encounter along its route, in order to increase the amount of mobile traffic offloaded to WiFi and to reduce the data transfer delay.
- We evaluate the procedures considering empirical measurements and a wide range of parameter values, and show their robustness against time and throughput estimation errors.

[1] Source: Cisco Visual Networking Index: Global Mobile Data Traffic Forecast Update, 2011-2016, Feb. 12, 2012

MobiArch'12 http://doi.acm.org/10.1145/2348676.2348682

Prior work has shown the predictability of bandwidth for cellular networks [12] and for WiFi [9, 10]. The work of [13] investigates how bandwidth prediction can improve scheduling in vehicular multi-homed networks and [14] investigates improvements for mobile video rate adaptation. Bandwidth prediction, together with transparent roaming and handover, for improving video streaming is investigated in [3]. Bandwidth prediction for client-side pre-buffering to improve video streaming is investigated in [11]. The works [14, 3, 11] focus on mobile networks, whereas our work investigates mobile data offloading to WiFi. Moreover, unlike [11] which considers pre-buffering at the client device, we investigate prefetching data to local caches in WiFi hotspots.

Exploiting delay tolerance to increase mobile data offloading to WiFi is investigated in [1]. The work of [7] showed that delay tolerance of up to 100 seconds provides minimal offloading gains; however, this applies to human daily mobility, rather than vehicles. The work in [5] applies a user utility model to reduce the mobile throughput by offloading traffic to WiFi, focusing on a transport layer protocol design to integrate cellular and WiFi networks, and utilizing throughput prediction over a 1-second interval. Our work differs in that we consider both delay tolerant and delay sensitive traffic, and exploit data prefetching and prediction involving multiple WiFi hotpots along a vehicle's route.

The feasibility of using prediction together with prefetching is investigated in [2], which develops a prefetching protocol (based on HTTP range requests), but does not propose or evaluate specific prefetching algorithms. In this paper we propose algorithms for delay tolerant and delay sensitive traffic, and evaluate their performance and robustness against time and throughput estimation errors. Prefetching to improve the performance of video delivery is investigated in [4], which proposes a centralized model to prefetch data in cellular femtocell networks. Prefetching algorithms to reduce the peak load of mobile networks by offloading traffic to WiFi hotspots are investigated in [8]. Our work differs in that we consider prefetching for multiple WiFi hotpspots along a vehicle's route, and investigate client-side algorithms for prefetching in the case of both delay tolerant and delay sensitive traffic.

The rest of this paper is structured as follows: Section II discusses mobility and bandwidth prediction. Section III proposes procedures to exploit prediction and prefetching for delay tolerant and delay sensitive traffic. Section IV evaluates the procedures considering empirical measurements and investigates their robustness to time and throughput errors. Finally, Section V concludes the paper identifying future research directions.

II. Mobility and bandwidth prediction

Route selection in vehicles is nowadays commonplace. The proliferation of smartphones equipped with GPS sensors can enhance route selection by periodically sending timestamped geo-location updates to a server (crowd-sourcing). Such timed geo-location data can produce real-time travel information for route segments, which can be used for route selection based on the actual travel time that considers road traffic conditions. We have implemented and demonstrated such an application, called "OptiPath", on Android smartphones [6]. With the OptiPath application, a smartphone records the route that it covers together with real-time traffic information, such as speed and duration, and periodically sends this information to a server. The server can use this information to estimate the travel time for different alternate routes between two endpoints. Interestingly, the architecture of such an application is very similar to systems that construct bandwidth or connectivity maps for mobile and WiFi access [9, 13, 10].

The goal of this paper is not to develop a new system for mobility and bandwidth prediction, but to propose and evaluate procedures that exploit prediction information that is available by systems such as the ones mentioned above, in order to utilize prefetching and enhance mobile data offloading to WiFi.

III. Exploiting mobility prediction and prefetching

In this section we present procedures that exploit mobility prediction and prefetching to enhance data offloading from the mobile network to WiFi hotspots. Mobility prediction provides knowledge of how many WiFi hotspots a node (vehicle) will encounter, when they will be encountered, and for how long the node will be in each hotspot's range. In addition to the aforementioned mobility information, we assume that there is information on the estimated throughput that is available in the WiFi hotspots and the mobile network, at different positions along the vehicle's route; for the former, the information includes both the throughput for transferring data from a remote location, e.g., through an ADSL backhaul link, and the throughput for transferring data from a local cache (note that this throughput estimate is used only in the case of prefetching).

We present procedures for both delay tolerant and delay sensitive traffic. For delay tolerant traffic, the procedures try to minimize the mobile throughput in order to maximize the amount of data offloaded to WiFi, while ensuring that the data is transferred within a given delay constraint. For delay sensitive traffic the goal is to minimize the transfer delay.

For prefetching, mobility and throughput information is used to estimate the data to be cached in the next WiFi hotspot that will be encountered. Prefetching can be advantageous when the throughput of transferring data from a local cache in the WiFi hotspot is higher than the throughput for transferring data from its original remote server location. This occurs when the backhaul link connecting the hotspot to the Internet has low capacity (e.g., for an ADSL link) or when it is congested; this is likely to become more common as the use of the IEEE 802.11n standard increases.

III.A. Delay tolerant traffic

For delay tolerant traffic our objective is to maximize the amount of data offloaded to WiFi, while ensuring that the whole data object is transferred within a given delay threshold.

The pseudocode for the procedure to exploit mobility prediction and prefetching is shown in Algorithm 1. The procedure defines the computations and actions that a mobile node takes when it exits a WiFi hotspot, hence has only mobile access (Line 15), and when it enters a WiFi hotspot (Line 24). Initially, the procedure estimates the amount of data that can be transferred over WiFi (Line 16), and from this the amount of data that needs to be transferred over the mobile network (Line 18). Additionally, the procedure estimates the total time the node has WiFi access (Line 17) and, from this value and the delay threshold, it estimates the duration the node has only mobile access (Line 18). From the amount of data that needs to be transferred over the mobile network and the duration of mobile-only access, the minimum throughput for transferring data over the mobile network can be estimated (Line 19). To perform prefetching, whenever the node exits a WiFi hotspot the procedure estimates the amount of data to be prefetched (cached) in the next WiFi hotspot (Line 20) and the corresponding offset (Line 21); this offset depends on the amount of data that will be transferred over the mobile network until the node reaches the next WiFi hotspot.

Note that when the node enters a WiFi hotspot, it might be missing some portion of the data object up to the offset from which data has been cached in the hotspot; this can occur if, due to a time estimation error, the node reaches the WiFi hotspot earlier than the time it had initially estimated. In this case, the missing data needs to be transferred from the data object's original remote location (Line 25). Also, again due to a time estimation error, the amount of data cached in the WiFi hotspot can be smaller than the amount the node could have transferred within the time it is in the hotspot's range. In this case, the node uses its remaining time in the WiFi hotspot to transfer data from the data object's original location (Line 27).

Algorithm 1 Procedure to exploit mobility prediction and prefetching for delay tolerant traffic

1: **Variables:**
2: D: size of data object to be transferred
3: T_{thres}: maximum delay threshold for transferring data object
4: N_{WiFi}: remaining WiFi hotspots to be encountered until T_{thres}
5: $D^{\min}_{\text{WiFi}}$: estimated minimum amount of data to be transferred in all WiFi hotspots that will be encountered
6: D_{mobile}: amount of data to be transferred over mobile network
7: $T^{\min}_{\text{WiFi},i}, T^{\max}_{\text{WiFi},i}$: min, max duration node is connected to WiFi i
8: T_{mobile}: total duration that node is not in range of WiFi
9: $T_{\text{next WiFi}}$: average time until node enters range of next WiFi
10: $R^{\min}_{\text{WiFi},i}, R^{\max}_{\text{WiFi},i}$: min, max throughput of WiFi i
11: R_{mobile}: throughput to download data over the mobile network
12: $D^{\text{cache}}_{\text{WiFi},next}$: amount of data cached in next WiFi hotspot
13: *Offset*: estimated position in data object of data transferred until node enters next WiFi hotspot
14: **Algorithm:**
15: **if** node exits WiFi hotspot **then**
16: $\quad D^{\min}_{\text{WiFi}} \leftarrow \sum_{i \in N_{\text{WiFi}}} \left(R^{\min}_{\text{WiFi},i} \cdot T^{\min}_{\text{WiFi},i} \right)$
17: $\quad T^{\min}_{\text{WiFi}} \leftarrow \sum_{i \in N_{\text{WiFi}}} T^{\min}_{\text{WiFi},i}$
18: $\quad D_{\text{mobile}} \leftarrow D - D^{\min}_{\text{WiFi}}$ & $T_{\text{mobile}} \leftarrow T_{\text{thres}} - T^{\min}_{\text{WiFi}}$
19: $\quad R_{\text{mobile}} \leftarrow D_{\text{mobile}} / T_{\text{mobile}}$
20: $\quad D^{\text{cache}}_{\text{WiFi},next} \leftarrow R^{\max}_{\text{WiFi},next} \cdot T^{\max}_{\text{WiFi},next}$
21: $\quad$ *Offset* $\leftarrow R_{\text{mobile}} \cdot T_{\text{next WiFi}}$
22: $\quad$ Cache $D^{\text{cache}}_{\text{WiFi},next}$ data in next WiFi starting from *Offset*
23: $\quad$ Transfer data over mobile network with throughput R_{mobile}
24: **else if** node enters WiFi hotspot **then**
25: $\quad$ Transfer data that has not been received up to *Offset* from original object location
26: $\quad$ Transfer data from local cache
27: $\quad$ Use remaining time in WiFi hotspot to transfer data from original object location
28: **end if**

The pseudocode to exploit mobility prediction without prefetching is shown in Algorithm 2. The procedure only estimates the throughput for transferring traffic over the mobile network. When prefetching is not used, the amount of data transferred over WiFi can be smaller, if the throughput for transferring data from a remote location is smaller than the throughput for transferring data from a local cache. We assume that this is the case, hence Line 13 of Algorithm 2 considers the backhaul throughput $R^{\min}_{\text{bckhl},i}$, rather than the WiFi throughput as in Line 16 of Algorithm 1.

Algorithm 2 Procedure to exploit mobility prediction for delay tolerant traffic

Variables:
D: size of data object to be transferred
T_{thres}: maximum delay threshold for transferring data object
N_{WiFi}: remaining WiFi hotspots to be encountered until T_{thres}
$D^{\min}_{\text{WiFi}}$: estimated minimum amount of data to be transferred in all WiFi hotspots that will be encountered
D_{mobile}: amount of data to be transferred over mobile network
$T^{\min}_{\text{WiFi},i}$: min duration node is connected to WiFi i
T_{mobile}: total duration that node is not in range of WiFi
$R^{\min}_{\text{bckhl},i}$: min throughput of backhaul at hotspot i
R_{mobile}: throughput to download data over the mobile network
Algorithm:
if node exits WiFi hotspot **then**
  $D^{\min}_{\text{WiFi}} \leftarrow \sum_{i \in N_{\text{WiFi}}} \left(R^{\min}_{\text{bckhl},i} \cdot T^{\min}_{\text{WiFi},i} \right)$
  $T^{\min}_{\text{WiFi}} \leftarrow \sum_{i \in N_{\text{WiFi}}} T^{\min}_{\text{WiFi},i}$
  $D_{\text{mobile}} \leftarrow D - D^{\min}_{\text{WiFi}}$ & $T_{\text{mobile}} \leftarrow T_{\text{thres}} - T^{\min}_{\text{WiFi}}$
  $R_{\text{mobile}} \leftarrow D_{\text{mobile}} / T_{\text{mobile}}$
  Transfer data over mobile network with throughput R_{mobile}
else if node enters WiFi hotspot **then**
  Transfer data from original object location
end if

III.B. Delay sensitive traffic

The pseudocode for the procedure to exploit mobility prediction and prefetching in the case of delay sensitive traffic is shown in Algorithm 3. Unlike delay tolerant traffic, in order to minimize the transfer delay for delay sensitive traffic, we always use the maximum throughput that is available in the mobile network (Line 11). As in the case of delay tolerant traffic, when the mobile node exits a WiFi hotspot it estimates the offset and the amount of data that needs to be prefetched in the next WiFi hotspot that the node will encounter.

Algorithm 3 Procedure to exploit mobility prediction and prefetching for delay sensitive traffic

1: **Variables:**
2: $T^{\max}_{\text{WiFi},next}$: maximum duration node is connected to next WiFi
3: $R^{\max}_{\text{WiFi},next}$: maximum throughput of next WiFi hotspot
4: $T_{\text{next WiFi}}$: time until node enters range of next WiFi
5: $R^{\max}_{\text{mobile}}$: maximum throughput of mobile network
6: $D^{\text{cache}}_{\text{WiFi},next}$: amount of data cached in next WiFi hotspot
7: *Offset*: estimated position in data object of data transferred until node enters next WiFi hotspot
8: **Algorithm:**
9: **if** node exits WiFi hotspot **then**
10:   $D^{\text{cache}}_{\text{WiFi},next} \leftarrow R^{\max}_{\text{WiFi},next} \cdot T^{\max}_{\text{WiFi},next}$
11:   *Offset* $\leftarrow R^{\max}_{\text{mobile}} \cdot T_{\text{next WiFi}}$
12:   Cache $D^{\text{cache}}_{\text{WiFi},next}$ data in next WiFi starting from *Offset*
13:   Transfer data over mobile network with throughput $R^{\max}_{\text{mobile}}$
14: **else if** node enters WiFi hotspot **then**
15:   Transfer data that has not been received up to *Offset* from original object location
16:   Transfer data from local cache
17:   Use remaining time in WiFi hotspot to transfer data from original object location
18: **end if**

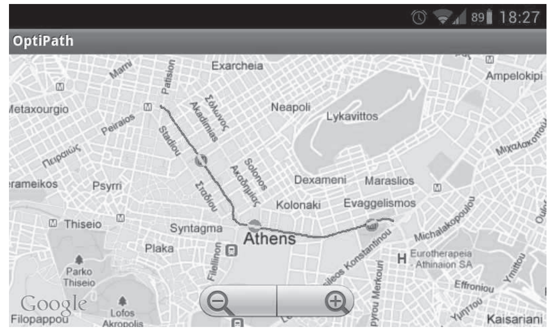


Figure 1: Route considered in the evaluation. The time and location information was obtained empirically. Three WiFi APs where enbedded along the route to conduct the evaluation. The total travel time for the route is 269 seconds.

Note that there is no procedure for exploiting only mobility prediction (without prefetching) for delay sensitive traffic. This is because for delay sensitive traffic the goal is to minimize the transfer delay, hence the maximum available mobile throughput is always used.

IV. Evaluation

The evaluation of the proposed procedures for mobile data offloading considers empirical measurements obtained by the OptiPath application mentioned in Section II. Specifically, we consider a journey between two locations in the center of Athens, Greece, for which the application provides a selected route and mobility information that involves the travel time for different route segments. Along the route, we embed three WiFi access points, each with range 60 meters. The route considered in our evaluation, with the three embedded WiFi networks, is shown in Figure 1 which contains a screenshot of the OptiPath application. Based on the aforementioned data, we can separate the full route into segments where the moving node has either mobile or WiFi connectivity, Table 1.

Our goal is to investigate the performance of the proposed algorithms for a wide range of parameter values, including errors in estimating the duration and the available throughput in each route segment, which can be caused by varying road traffic and wireless link conditions. The parameters considered are shown in Table 2. The time error determines how much the times at which the node changes access technology can differ from the empirical values in Table 1; for example, a 10% time error means that the time at which the first segment (where the node has mobile access) ends and the second segment (where it has WiFi access) begins is in the interval $[0.9 \cdot 12, 1.1 \cdot 12] =$

$[10.8, 13.2]$ seconds. Note that our empirical measurements show that under typical road traffic conditions, the timing for the various route segments can differ 10-20%.

The throughput error determines the throughput's deviation from its average in Table 2; for example, a 40% throughput error means that the mobile throughput is in the interval $[0.6, 1.4]$ Mbps. Note that our empirical measurements of the mobile throughput for the route in Figure 1 give values in the range $[1.2, 1.4]$ Mbps. In this paper we only consider the downlink direction, hence the backhaul throughput in Table 2 refers to the downstream. The improvements achieved in the uplink direction, by uploading data to a local cache, could be even higher, e.g., in the case of an ADSL backhaul.

The evaluation results presented in the remainder of this section are based on numerically computing the data transferred over the mobile and WiFi networks, for the parameters in Tables 1 and 2.

The behavior of the procedures of Section III is illustrated in Figure 2, which shows the cumulative data transferred as a function of time. Figure 2 was produced from numerical calculations using the parameters in Tables 1 and 2, assuming that the throughput in each route segment is constant; as a result, the data transferred in each segment is depicted as a straight line.

For delay tolerant traffic, the procedures seek to maximize the amount of traffic offloaded to WiFi, hence to minimize the amount of traffic transferred over the mobile network, while completing the data transfer until the moving node reaches the end of its journey. Hence, Figure 2(a) shows that for delay tolerant traffic all three WiFi hotspots are utilized. The procedure that exploits prediction and prefetching, Algorithm 1, differs from the procedure that exploits only prediction, Algorithm 2, in that more data is transferred over the WiFi network, which is possible since the transfer throughput from a local cache is higher than the throughput over the hotspot's backhaul link. Similarly, the slopes for the mobile segments are smaller for the procedure that exploits prediction and prefetching, since the amount of data transferred over the mobile network is smaller.

For delay sensitive traffic, Figure 2(b), the maximum available mobile throughput is always used, hence the slope of the mobile segments when prediction and prefetching is used (Algorithm 3) is the same as when prediction and prefetching is not used. As in the case of delay tolerant traffic, prefetching helps offload a larger amount of mobile traffic to WiFi, hence results in a smaller transfer delay compared to the case where no prefetching is used.

In the following subsections we discuss the evaluation results for delay tolerant and delay sensitive traffic. The graphs show averages and 95% confidence intervals from 20 runs of each scenario. In each run, the time instances the access technology changes and the throughput of WiFi, the hotspot backhaul, and the mobile network is randomly selected based on the average values and error percentages in Tables 1 and 2.

Table 1: Connectivity along route. The third column shows the time the node enters the corresponding segment. The times were obtained from empirical measurements and after embedding three WiFi APs along the route of Figure 1.

Segment	Access	Time (sec)
1	mobile	0
2	WiFi	12
3	mobile	32
4	WiFi	149
5	mobile	157
6	WiFi	227
7	mobile	233

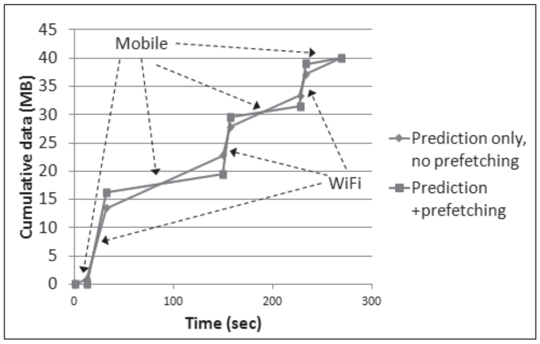
Table 2: Parameter values.

Parameter	Values
Data object size	10, 20 (default for delay sensitive), 30 (default for delay tolerant), 40, 50 MB
Mobile throughput	1 Mbps (average)
WiFi throughput	10 Mbps (average)
Backhaul throughput	2.5, 5 (default), 7.5 Mbps
Time error	10% (default), 20%, 30%, 40%
Throughput error	20% (default), 40%, 60%

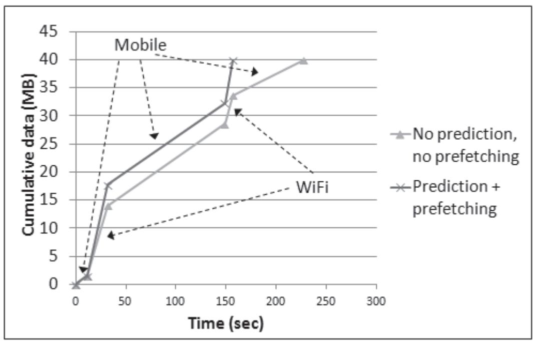
IV.A. Delay tolerant traffic

In this subsection we discuss results for delay tolerant traffic, where a data object needs to be transferred until the end of the vehicle's route in Figure 1. We compare the following three cases: the procedure in Algorithm 1 that exploits mobility prediction and prefetching, the procedure in Algorithm 2 that exploits only mobility prediction without prefetching, and the case when prediction is not utilized and the maximum available mobile throughput is always used. The metrics we consider are the percentage of traffic that is offloaded to the mobile network and the cache size when prefetching is used.

Data object size: Figure 3(a) shows the percentage of offloaded traffic for different data object sizes. For all data sizes the performance of the prediction + pre-

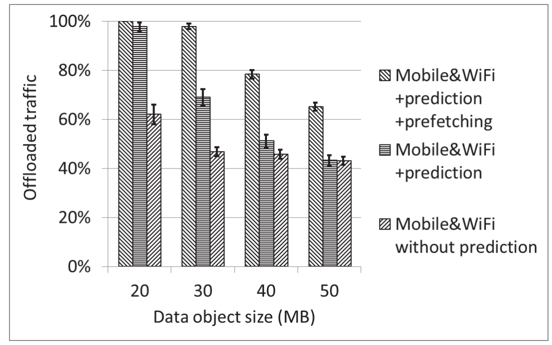


(a) Delay tolerant

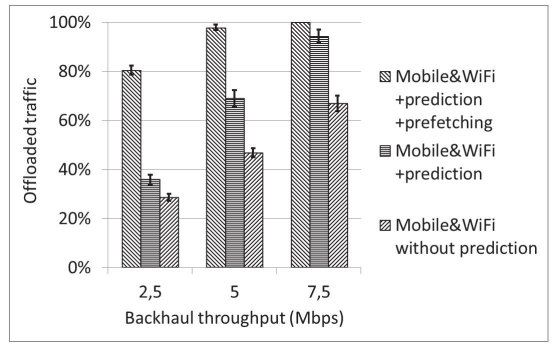


(b) Delay sensitive

Figure 2: Cumulative data as a function of time for delay tolerant and delay sensitive traffic. Data object size: 40 MB.



(a) Data object size



(b) Backhaul throughput

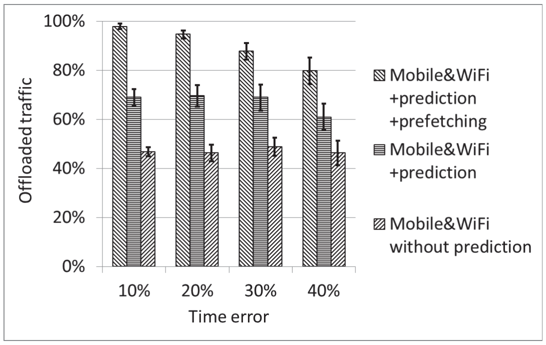
Figure 3: Percentage of offloaded traffic as a function of data object size and backhaul throughput. Delay tolerant traffic.

fetching scheme is more than 50% higher compared to the case where prediction and prefetching is not used. For small data object sizes, the performance of both the prediction + prefetching and the prediction schemes is close. The occurs because when the object size is small, then the hotspot backhaul throughput is enough to offload a high percentage of the mobile traffic, hence the improvement achieved by prefetching and downloading data from a local cache is not significant. On the other hand, for large data sizes, the performance of the prediction scheme is close to the performance when prediction is not used; this occurs because for large object sizes the mobile network is used close to its maximum throughput, hence prediction is not beneficial.

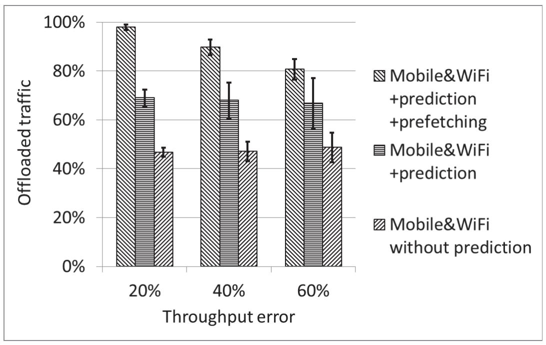
Backhaul throughput: Figure 3(b) shows the percentage of offloaded traffic for different hotspot backhaul throughputs. Observe that prefetching can offload approximately 50% (for 7.5 Mbps) and 180% (for 2.5 Mbps) more mobile traffic compared to when prediction and prefetching is not used, i.e., when the maximum available mobile throughput is used. Moreover, when the backhaul throughput is low, the performance when only prediction is used is close to the performance when prediction is not used; this happens because when the backhaul throughput is low, then the mobile network needs to be used more, hence the mobile throughput is close to its maximum. On the other hand, when the backhaul throughput is high, then the performance of prediction and prefetching is close to the performance when only prediction is used; this occurs because when the backhaul throughput is high and close to the WiFi throughput, there are smaller gains from downloading data from a local cache.

Time error: Figure 4(a) shows how the percentage of offloaded traffic is affected by the time error. Observe that the performance when prediction and prefetching are used decreases as the time error increases; this occurs because the time error reduces the effectiveness of prefetching. Also observe that the performance when only prediction is used is affected only for 40% time error. Nevertheless, the performance when prediction and prefetching are used is more than 70% higher than when prediction and prefetching are not used, and more than 30% higher than when only prediction is used.

Throughput error: Figure 4(b) shows that the throughput error affects the performance of the prediction



(a) Time error



(b) Throughput error

Figure 4: Percentage of offloaded traffic as a function of time and throughput error. Delay tolerant traffic.

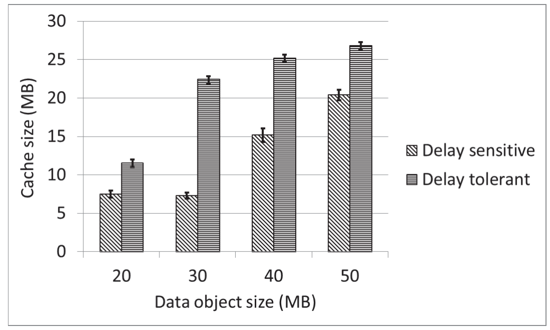


Figure 5: Cache requirements for prefetching

and prefetching scheme most. Nevertheless, its performance remains more than 20% higher than the prediction-only scheme and more than 65% higher than the case where prediction and prefetching are not used.

Cache requirements for prefetching: Figure 5 shows that, as expected, the cache requirements for delay tolerant traffic are higher than for delay sensitive traffic, since the former's goal is to maximize the amount of mobile data offloaded to WiFi. Moreover, the cache requirements increase with the data size for delay tolerant traffic. On the other hand, for delay sensitive traffic the cache requirements for 20 and 30 MB data objects remains the same, since for both these sizes only the first WiFi hotspot is used, and prefetching is utilized to transfer the same amount of traffic in both cases.

IV.B. Delay sensitive traffic

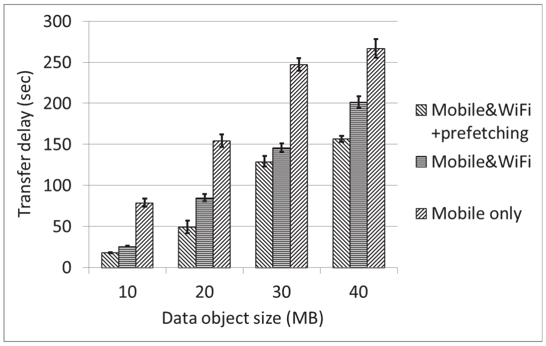
In this subsection we discuss the results for delay sensitive traffic. A key difference compared to delay tolerant traffic is that now the maximum throughput available in the mobile network is always used. We compare three cases: the procedure that exploits both mobility prediction and prefetching, Algorithm 3, the case where prediction and prefetching are not used, and the case where only the mobile network is used. The performance metric is the delay for transferring a data object.

Data object size: Figure 6(a) shows the transfer delay as a function of data object size. Prediction and prefetching achieve a transfer delay that is lower by more than 40% compared to the case where only the mobile network is used. The procedure that exploits prediction and prefetching achieves transfer delay that is 10% to more than 40% lower than the transfer delay when prediction and prefetching are not used; this improvement depends on the number of WiFi hotspots that are utilized, which in turn depends both on the location of the WiFi hotspots and the data object size. Specifically, only the first WiFi hotspot is utilized for data objects with size 20 and 30 MB, and a smaller percentage of the traffic is offloaded for a 30 MB data object; due to this, for a 30 MB object size, the performance when prediction and prefetching is used is closer to the performance when prediction and prefetching are not used.

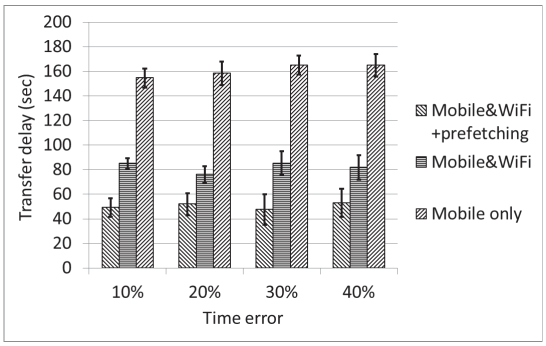
Time error: Figure 6(b) shows the influence of the time error on the transfer delay. Observe that as the time error increases, the variability of the transfer delay increases (the 95% confidence interval is larger), but the average transfer delay for all schemes remains the same. (We only present present performance results for delay sensitive traffic in the case of time errors; results for throughput estimation errors will be presented in an extended version of this paper.)

V. Conclusions and Future Work

We have presented and evaluated procedures that exploit mobility prediction and prefetching to enhance mobile data offloading, for both delay tolerant and delay sensitive traffic. Our evaluation shows how the performance depends on various parameter values, and the robustness of the proposed procedures



(a) Data object size



(b) Time error

Figure 6: Transfer delay for different data object sizes and time errors. Delay sensitive traffic.

to time and throughput errors. Future work includes evaluating the energy gains and extending the procedures to allow different tradeoffs between the delay, the amount of offloaded traffic, and the energy efficiency. We are also planning to develop a prototype to demonstrate the gains of the proposed offloading procedures.

References

[1] A. Balasubramanian, R. Mahajan, and A. Venkataramani. Augmenting mobile 3G using WiFi. In *Proc. of ACM MobiSys*, 2010.

[2] P. Deshpande, A. Kashyap, C. Sung, and S. Das. Predictive Methods for Improved Vehicular WiFi Access. In *Proc. of ACM MobiSys*, 2009.

[3] K. Evensen, A. Petlund, H. Riiser, P. Vigmostad, D. Kaspar, C. Griwodz, and P. Halvorsen. Mobile Video Streaming Using Location-Based Network Prediction and Transparent Handover. In *Proc. of ACM NOSDAV*, 2011.

[4] N. Golrezaei, K. Shanmugam, A. G. Dimakis, A. F. Molisch, and G. Caire. FemtoCaching: Wireless Video Content Delivery through Distributed Caching Helpers. In *Proc. of IEEE Infocom*, 2012.

[5] X. Hou, P. Deshpande, and S. R. Das. Moving Bits from 3G to Metro-Scale WiFi for Vehicular Network Access: An Integrated Transport Layer Solution. In *Proc. of IEEE ICNP*, 2011.

[6] C. Kalampokis, D. Kalyvas, I. Latifis, and V. A. Siris. OptiPath: Optimal Route Selection Based on Location Data Collected from Smartphones. In *Proc. of Joint ERCIM eMobility & MobiSense Workshop, co-located with WWIC*, 2012.

[7] K. Lee, I. Rhee, J. Lee, S. Chong, and Y. Yi. Mobile Data Offloading: How Much Can WiFi Deliver? In *Proc. of ACM CoNEXT*, 2010.

[8] F. Malandrino, M. Kurant, A. Markopoulou, C. Westphal, and U. C. Kozat. Proactive Seeding for Information Cascades in Cellular Networks. In *Proc. of IEEE Infocom*, 2012.

[9] A. J. Nicholson and B. D. Noble. BreadCrumbs: Forecasting Mobile Connectivity. In *Proc. of ACM Mobicom*, 2008.

[10] J. Pang, B. Greenstein, M. Kaminsky, D. McCoy, and S. Seshan. Wifi-Reports: Improving Wireless Network Selection with Collaboration. In *Proc. of ACM MobiSys*, 2009.

[11] V. Singh, J. Ott, and I. Curcio. Predictive Buffering for Streaming Video in 3G Networks. In *Proc. of IEEE WoWMoM*, 2012.

[12] J. Yao, S. S. Kahnere, and M. Hassan. An Empirical Study of Bandwidth Predictability in Mobile Computing. In *Proc. of ACM WinTech*, 2008.

[13] J. Yao, S. S. Kahnere, and M. Hassan. Geo-intelligent Traffic Scheduling for Multi-Homed On-Board Networks. In *Proc. of ACM MobiArch*, 2009.

[14] J. Yao, S. S. Kahnere, and M. Hassan. Quality Improvement of Mobile Video Using Geo-intelligent Rate Adaptation. In *Proc. of IEEE WCNC*, 2010.