Offloading Cellular Networks through ITS Content Download

Francesco Malandrino, Claudio Casetti, Carla-Fabiana Chiasserini
Dipartimento di Elettronica e Telecomunicazioni, Politecnico di Torino
lastname@tlc.polito.it

Marco Fiore
Université de Lyon, INRIA, INSA-Lyon
marco.fiore@insa-lyon.fr

***Abstract*—Content downloading by mobile users is expected to significantly increase the cellular network load. Vehicular users, in particular, are likely to engage in information retrieval on the move: in this context, Intelligent Transportation Systems (ITS) can play an important role in offloading the cellular infrastructure. We investigate the effectiveness of ITS in this task, considering that roadside units (RSUs) can exploit mobility prediction to decide which data they should fetch from the Internet and schedule transmissions to vehicles, either potential relays or downloaders. Rather than presenting a specific prediction scheme, we propose a model that allows us to express and account for any prediction technique in a simple, yet effective, manner. We then provide a probabilistic graph-based representation of the system that accounts for the prediction uncertainty. We use such a representation to study the network dynamics by efficiently solving a (non-integer) LP problem. Our results show that the above approach to content downloading through ITS can achieve an 80% offload of the cellular network. Also, we investigate the dependency of the system performance on the accuracy of the mobility prediction, and which prediction errors have the largest impact.**

I. Introduction

The growing communication speed promised by new cellular technologies, such as LTE-Advanced, as well as the spreading of tablets, smartphones and USB dongles, is luring consumers into the false conviction that every information content is always readily available for prompt downloading. In fact, there is a serious concern lurking behind commercial offers promising ever-increasing speed of the air interface: the bandwidth availability in the cellular backhaul is hard-pressed to keep the pace with what is being offered on the air interface and what users expect [1].

In light of the cellular network congestion, many advocate the development of alternative communication systems to support and, as it were, relieve the cellular network in areas where the demand by mobile users is expected to be the thickest. In particular, content downloading accounts for most of the traffic in access networks [2], and is thus a prime candidate to be offloaded [3]. Its requirements are very different from those of information dissemination [4], [5] and uploading of user-generated data [6]. This makes solutions designed to offload these kinds of traffic unfit for the downloading scenario we consider.

Within the context of Intelligent Transportation Systems (ITS), several proposals in the literature have suggested the deployment of roadside infrastructure units (RSU) that provide spotty radio connectivity to on-board units in passing vehicles. Communication occurs through the DSRC (Dedicated Short-Range Communications) technology, according to what is commonly referred to as an Infrastructure-to-Vehicle (I2V) paradigm, and also leveraging opportunistic vehicle-to-vehicle (V2V) connectivity. Previous work, e.g., [7], has established that, in order to efficiently support content downloading, (i) RSU deployment should target the areas expected to be the most crowded by vehicles and (ii) I2V content transfer should be complemented by V2V data relaying.

A part of the picture is still missing, though. Given the ability to delivery information to passing by vehicles through a carefully planned-out RSU deployment, what exactly should be delivered to them? A spotty coverage could meet expectations only on condition that the short time under coverage is fruitful: RSUs should prefetch the content so as to have it promptly available for passing-by vehicles requesting it. Matching between storage at RSUs and demands by vehicles is, however, easier said than done. One possibility is that RSUs have access to the content demand and to predictions of mobility patterns, and exploit them to take prefetching decisions, as in [8]. Additionally, to make V2V transfers more effective, RSUs can leverage a similar approach for I2V communication toward relay vehicles deemed to meet downloaders later on.

In order to relieve the cellular network from the content delivery task, our work is the first to jointly study the problems of content prefetching at RSUs, scheduling of I2V transmissions and management of V2V relay transfers, in presence of inaccurate mobility prediction.

To do so, we model the uncertainty affecting the mobility prediction through a *fog-of-war* probabilistic representation of the inter-node contacts (Sec. III). Our fog-of-war model is not a prediction technique itself, rather a convenient way to express the accuracy of the prediction, and to study its effect on the content downloading performance. It can thus provide an abstraction of any prediction technique (e.g., the ones we discuss in Sec. VII) and allows us to draw conclusions of general validity.

The output of the fog-of-war model is used to build a time-expanded graph with probabilistic weights, representing the evolution of the inter-node contacts (Sec. IV-A). We exploit the graph to formulate an optimization problem, to be solved at each RSU, that jointly addresses content prefetching and scheduling (Sec. IV-B). The data scheduled by RSUs toward relays are then delivered to downloaders, according to different schemes, namely, a greedy strategy exploiting opportunistic encounters and an RSU-driven scheduling of relay-to-downloader transmissions (Sec. V). In our performance evaluation, we compare the offloading efficiency of the system

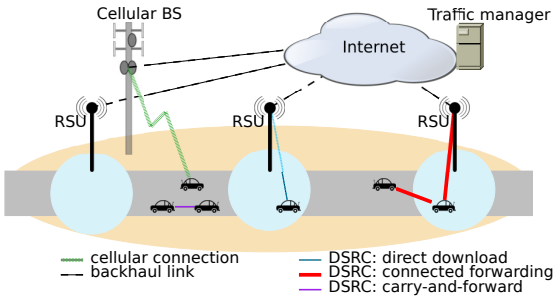


Fig. 1. Network system.

outlined above against benchmark solutions. Furthermore, we account for the existence of location-specific content and assess its benefits on the capability of ITS to relieve the cellular network (Sec. VI).

II. System Model

We consider a DSRC-based vehicular network composed of mobile users and fixed roadside units (RSUs), deployed over a road topology that is also covered by a cellular infrastructure. As depicted in Fig. 1, RSUs provide a spotty but high-throughput, inexpensive connectivity to vehicles, whereas the cellular network guarantees seamless coverage, which however comes at some connection cost. We assume that RSUs and vehicles have one DSRC interface only, and that I2V and V2V communications occur on different frequency channels.

Users of the vehicular network may become *downloaders*, i.e., they may wish to retrieve different types of data from the fixed network (e.g., the Internet). Assuming that vehicles have both a DSRC and a cellular radio interface, multiple transfer paradigms for content delivery are possible. More precisely, downloaders can exploit the ITS network to perform *direct* transfers from the RSUs, or to be assisted by other vehicles acting as *relays*. In the latter case, we consider connected forwarding, i.e., traffic relaying through a connected multi-hop path, as well as carry-and-forward, i.e., traffic relaying through vehicles that store and carry the data before delivering them to the target downloader. Alternatively, downloaders can resort to *cellular* transfers, in order to retrieve the desired content from the fixed network.

We model the downloaders' demand by considering *what* they request and *how* they get it, as follows. As far as the *what* is concerned, we address both the cases of location-independent and location-specific content demand. Regarding the *how*, downloaders try at first to obtain the data through inexpensive opportunistic exchanges with RSUs and relay vehicles. If the desired content cannot be fully retrieved within a timeout T, the downloaders will pay to fetch the remaining portion via a cellular transfer. Note that this model provides an incentive for users to offload the cellular network through ITS. Next, we detail the operations that the network and the users undertake during the content downloading process.

A user wishing to retrieve a content generates a request to an Internet-based query management system, via either an RSU or the cellular network [8]. Such a management system forwards the pending request to the RSUs in the area where the downloader is traveling. RSUs are then in charge of (i) fetching portions of the content from some server storing it, and (ii) delivering the data to the target downloader directly, or to a relay vehicle deemed to meet the downloader later on.

It is clear that, in order to efficiently use the network resources over the backbone and the airtime on the wireless medium, RSUs must take content prefetching and scheduling decisions by foreseeing future direct or relay transfer opportunities that involve downloader vehicles [9]. To that end, we assume that a forecast of the future I2V and V2V contacts is periodically issued by a traffic manager to the RSUs, as in emerging real-time traffic monitoring systems [10]. Such information also includes the identity and pending queries of downloaders that are in the network at the time of the issued forecast; we stress that the traffic manager is instead unaware of future content requests.

Based on such contact information and taking into account the rate B at which data can be retrieved from the server, the RSUs make locally optimal decisions on which data to prefetch and toward which vehicles (either relays or downloaders) they should be transmitted. If RSUs delegate portions of content to relays, and these are in range of, or subsequently meet, a downloader interested in such a content, V2V transfers occur. Multi-hop data transmissions, be they of the connected forwarding or carry-and-forward type, are limited to two hops from the RSU, since this already yields nearly optimal performance [7]. For the same reason, we do not compare our approach against complex multi-hop routing protocols for DTNs, whose complexity is unnecessary in a vehicular network with infrastructure.

We also remark that all vehicles are assumed to be available to relay traffic whenever they are not receiving data from an RSU. Given the storage capabilities of today's communication nodes, the memory capacity at RSUs and vehicles is not considered to be an issue.

III. The Traffic Manager Prediction

We assume that the traffic manager predicts the node mobility with time granularity δ; in the following, we refer to the interval of duration δ as time step. The prediction is updated every H steps, upon the reception of new information on vehicle positions. We also consider that H is the time horizon over which the prediction is made.

Based on the predicted positions of vehicles, the traffic manager considers that two nodes (either mobile users or RSUs) are neighbors if their distance is below or equal to their maximum radio range. Then, by defining a wireless link shared by a pair of neighboring nodes as a *contact*, it forecasts the contacts deemed to exist at each of the next H time steps, with a given probability. A contact may extend over multiple steps and its data rate can be related to the node distance and propagation conditions (see, e.g., the description in Sec. VI-A).

To model the limited accuracy in the prediction of the contacts and their characteristics, as compiled by the traffic manager, we adopt a prediction technique-independent approach. Rather than considering one specific prediction methodology

(e.g., among those cited in Sec. VII), we propose a *fog-of-war model*, which provides an accurate abstraction of virtually any prediction technique and accounts for different precision levels of the forecast (a more thorough discussion on this aspect can be found in [11]).

Specifically, let $\mathcal{P}(u, H)$ be a contact prediction generated by the traffic manager at step u for the next H steps. Given that the prediction accuracy may be affected by several sources of error, we assume the predicted V2V and I2V contacts occurring between the present time, u, and the prediction horizon, $u+H-1$, to be affected by a Gaussian-distributed noise with zero mean and variance σ^2. The Gaussian distribution is chosen to describe the effect of several additive prediction errors (e.g., uncertainty on the node positions, on the propagation conditions, on the link establishment procedure).

More formally, for each actual contact between a generic node pair starting at step $k \in [u, u+H)$, we extract a realization ν of the noise. If $|\nu| \leq 1$, we associate a probability $1-|\nu|$ to the contact, which expresses the likelihood with which the traffic manager expects the contact to take place. Otherwise, the contact is evicted and a new, *spurious* one is created and associated with a probability equal to $\min\{|\nu|-1, 1\}$. The nodes sharing the spurious contact are chosen randomly among the network nodes, and the spurious contact inherits the duration and data link rate of the true contact that it has replaced. This simple model allows us to capture the possibility that prediction techniques underestimate actual contact opportunities, when $0 < |\nu| \leq 1$, and wrongly forecast future contacts, when $|\nu| > 1$.

The variance σ^2 models the accuracy of the prediction, since the larger the zero-mean noise variance, the less precise the estimation of the connectivity. We express the variance as $\sigma^2 = \sigma_0^2(k-u)$ for V2V contacts and $\sigma^2 = \frac{\sigma_0^2}{2}(k-u)$ for I2V contacts. Indeed, due to the mobility of both link endpoints, we expect V2V contacts to be affected by a variance that is twice that of I2V contacts[1]. Also, we let σ^2 grow linearly[2] with $k-u$, which accounts for the fact that predicting contacts farther in time becomes increasingly harder. As a result, spurious contacts, appearing with the same frequency with which actual contacts are evicted, are more frequent if the prediction accuracy is low (i.e., high σ_0^2) and the estimation is pushed far ahead in time (i.e., large $k-u$).

Finally, we point out that, since our fog-of-war model is defined by the value of σ_0^2, by varying such a value, we can match the output of different prediction techniques. To verify that, we applied a Markovian prediction technique of the first and second order to the reference scenario that we use later in our performance evaluation (Sec. VI-A). We found a very good agreement when σ_0^2=1.68 for the first-order model, and when σ_0^2=1.22 for the second-order model [11].

[1]This is because in I2V contacts the position of one of the two nodes (the RSU) is known exactly.

[2]A linear growth model is simple and, as shown in [11], serves our purposes. The non-linear growth can be considered without additional complexity.

IV. Pre-fetching and scheduling at RSUs

Upon compiling the prediction $\mathcal{P}(u, H)$, the traffic manager forwards it to each RSU which, in turn, updates it with the contacts with passing vehicles it actually sees (whether they were predicted in advance or not). Such contacts are assigned a probability equal to 1, while wrongly predicted I2V contacts involving the RSU are assigned a zero probability. Thus, each RSU r_i has its own prediction $\mathcal{P}_i(u, H)$ and updates it as the time elapses. The prediction is used to generate a directed time-expanded graph with probabilistic weights (TEG-PW), on which the RSU formulates a linear programming (LP) problem that jointly optimizes prefetching and scheduling. Note that LP problems can be solved in polynomial time and, in particular, our formulation is suitable to be solved on line.

A. Building the TEG-PW

The prediction $\mathcal{P}_i(u, H)$ allows an RSU r_i to model the time evolution of the contacts between network nodes through a time-expanded graph. Since the prediction is based on discrete time steps of duration δ, the same granularity is used in the construction of the graph.

In the graph, each vehicle v_l appearing in the prediction $\mathcal{P}_i(u, H)$ at step $k \in [u, u+H)$ is associated to a vertex v_l^k, whereas each RSU r_i is mapped at each step k onto a vertex r_i^k. We denote by $\mathcal{V}^k$ and $\mathcal{R}^k$ the sets of vertices representing, respectively, the vehicles and the RSUs at step k. At every k, a directed edge connecting two vertices represents the predicted contact between the corresponding pair of nodes. Such edges are referred to as intra-step and correspond either to I2V links, i.e., of the type (r_i^k, v_l^k), or to V2V links, i.e., of the type (v_l^k, v_m^k). We denote the set of I2V edges during step k by $\mathcal{E}_r^k$, and that of V2V edges by $\mathcal{E}_v^k$. Every intra-step edge in $\mathcal{E}_r^k$ and $\mathcal{E}_v^k$ is associated to a finite weight, representing the predicted data rate of the corresponding link at step k. As previously outlined, at the generic $k \in [u, u+H)$, each contact in $\mathcal{P}_i(u, H)$ is characterized by a probability of occurrence and an estimated data rate. We thus include these two aspects in the weight of an intra-step edge. As an example, consider a V2V contact between vehicles v_l and v_m at step k. We associate to the edge (v_l^k, v_m^k) a weight $w(v_l^k, v_m^k) = p(v_l^k, v_m^k) \cdot b(v_l^k, v_m^k)$, where $p(v_l^k, v_m^k)$ is the estimated contact probability between the two vehicles at k and $b(v_l^k, v_m^k)$ is the estimated maximum amount of data that can flow over the link during that time step. An identical discussion applies to I2V contacts.

Also, directed edges, of the type (v_l^k, v_l^{k+1}) or (r_i^k, r_i^{k+1}), are drawn between vertices representing the same node at two consecutive steps. While the edges in $\mathcal{E}_v^k$ and $\mathcal{E}_r^k$ represent anticipated transmission opportunities, these edges, referred to as intra-nodal, model the same node over time. They thus represent the possibility that vehicles physically carry data during their movement. Since we assume that the vehicle memory capabilities are unlimited, all intra-nodal edges are associated to an infinite weight. Note that accounting for the contact duration, instead of considering them as atomic, allows to model critical aspects of the real-world communication, like channel contention and the presence of hidden nodes.

Finally, the server(s) (from which RSUs retrieve the data) are modeled as a vertex named α. The graph is completed with B-weight edges (α, r_i^k), from α to any vertex $r_i^k \in \mathcal{R}^k$.

B. Making optimal decisions

At each step k, RSU r_i needs to take its prefetching and scheduling decisions. Specifically, each RSU determines: (i) which data, among those not already stored, have to be prefetched, in order to be transmitted to the vehicles (accounting for the limited rate at which data can be retrieved from the server); (ii) which data already available[3] at the RSU have to be delivered via I2V contacts actually seen at step k, i.e., to downloaders through direct transfers as well to candidate relays deemed to meet downloaders later on.

RSUs take decisions with the aim to maximize the fraction of content that users retrieve through ITS, within a time T since their request. Thus, each RSU formulates an optimization problem based on its TEG-PW as detailed next.

Let $\mathcal{C}$ be the content set, $t_{l,c}$ the step at which the generic downloader v_l sends a request for content c, and $\phi_{l,c}^k$ the fraction of the content that a user downloads at step k through the ITS network. Then, each RSU maximizes the following objective function over all content c and downloaders v_l:

$$\sum_{l} \sum_{c \in \mathcal{C}} \sum_{k=t_{l,c}}^{t_{l,c}+T} \phi_{l,c}^k \,, \tag{1}$$

The quantity $\phi_{l,c}^k$ can be computed by evaluating the amount of data that can be transferred at step k (i.e., the flow) over the edges of the type (v_m^k, v_l^k) and (r_i^k, v_l^k), with v_m and r_i being, respectively, a relay and an RSU storing at step k part of, or all, content c requested by v_l. More specifically, for each k, we define the expected flow for content c that is carried over the link associated to a V2V (resp. I2V) contact as $f_c(v_m^k, v_l^k)$ (resp. $f_c(r_i^k, v_l^k)$). From our definitions in Sec. IV-A, we have

$$f_c(v_m^k, v_l^k) \le w(v_m^k, v_l^k), \quad f_c(r_i^k, v_l^k) \le w(r_i^k, v_l^k). \tag{2}$$

By leveraging the flow definition above, we can write:

$$\phi_{l,c}^k = \frac{1}{s_c} \left[\sum_{(v_m^k, v_l^k) \in \mathcal{E}_v^k} f_c(v_m^k, v_l^k) + \sum_{(r_i^i, v_l^k) \in \mathcal{E}_r^k} f_c(r_i^k, v_l^k) \right],$$

where s_c is the content size.

The evaluation of the expected flows must account for the channel contention among network nodes as well as among flows related to different content transfers. Thus, beside ensuring non-negative flows in the TEG-PW, we need to introduce the constraints listed below.

Flow conservation. In the case where each downloader wants to fetch a different content, we need to impose that the total flow for a content on outgoing edges, scaled by the probability that the corresponding contacts occur, is equal to the total incoming flow for the same content. E.g., in the case of a relay vertex, we have:

[3]Data cached at RSUs are modelled by the flow on intra-nodal edges.

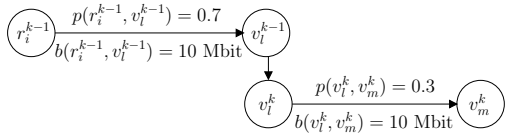


Fig. 2. Flow conservation: an example.

$$\sum_{(r_i^k, v_l^k) \in \mathcal{E}_r^k} f_c(r_i^k, v_l^k) = \sum_{(v_l^k, v_m^k) \in \mathcal{E}_v^k} \frac{f_c(v_l^k, v_m^k)}{p(v_l^k, v_m^k)} + f_c(v_l^k, v_l^{k+1}) \,. \tag{3}$$

As an example, consider the 2-step evolution in Fig. 2, where v_m is a downloader for content c. Note that the transmissions from r_i to v_l and from v_l to v_m take place at different steps, thus channel access has no effect here. Intuitively, we can try to transfer 10 Mbit from r_i to v_m, and we will succeed with probability $0.7 \cdot 0.3 = 0.21$. Then, the overall expected flow delivered to the downloader is $0.21 \cdot 10 = 2.1$ Mbit. However, if only the constraints in (2) were applied on each of the two intra-step edges, the expected flow should not exceed b times the edge probability. Hence, we could incorrectly conclude that the expected flow from r_i to v_m is $\min\{0.7 \cdot 10, 0.3 \cdot 10\} = 3$ Mbit. Instead, imposing (3) for vertices v_l^{k-1} and v_l^k, it correctly results that $f_c(r_i^{k-1}, v_l^{k-1}) = f_c(v_l^k, v_m^k)/p(v_l^k, v_m^k)$, i.e., $f_c(v_l^k, v_m^k) = 2.1$, which is consistent with our intuition.

Flow causality. In the case where more than one downloader requests the same piece of information, we replace the flow conservation constraint in (3) with the weaker constraint of *causality*. Indeed, while flow conservation implies causality, the vice versa does not hold.

In order for a node (be it a vehicle or an RSU) to transmit some data (of any content) at step k, such data must have been already downloaded from some other node at step $h \le k$. In other words, we need to introduce a *causality* constraint, imposing that, at each step k, the data downloaded by node v_m from node v_l until k (as opposed to "during step k alone") is no more than the data v_l obtained until k from other nodes. Thus, for any edge (v_l^k, v_m^k) and content c, we have that:

$$\sum_{h=1}^{k} \frac{f_c(v_l^k, v_m^k)}{p(v_l^k, v_m^k)} \le \sum_{h=1}^{k} \left[\sum_{v_n^h \in \mathcal{V}^h \setminus v_m^h} f_c(v_n^k, v_l^k) + \sum_{r_i^h \in \mathcal{R}^h} f_c(r_i^h, v_l^h) \right].$$

Channel access. We assume that the nodes access the channel using an IEEE 802.11-based scheme with RTS/CTS handshake. Thus, when v_l transmits to v_m, all neighbors of v_l and v_m must be silent. Also, recall that we assume V2V and I2V traffic not to interfere. Then, the channel access constraint for any v_l at step k is:

$$\sum_{\substack{(v_n^k, v_o^k) \in \mathcal{E}_v^k \\ c \in \mathcal{C}}} \mathbb{1}_{[v_n^k, v_l^k]} \frac{f_c(v_n^k, v_o^k)}{b(v_n^k, v_o^k)} + \sum_{\substack{(v_p^k, v_o^k) \in \mathcal{E}_v^k \\ c \in \mathcal{C}}} \mathbb{1}_{[v_o^k, v_l^k]} \left(1 - \mathbb{1}_{[v_p^k, v_l^k]}\right) \cdot \frac{f_c(v_p^k, v_o^k)}{b(v_p^k, v_o^k)} + \sum_{\substack{(r_i^k, v_o^k) \in \mathcal{E}_r^k \\ c \in \mathcal{C}}} \mathbb{1}_{[r_i^k, v_l^k]} \frac{f_c(r_i^k, v_o^k)}{b(r_i^k, v_o^k)} \le 1 \,,$$

where the indicator function is equal to 1 if the specified vertices either are neighbors or coincide, and it is 0 otherwise.

The three sums on the left hand side of the inequality account for the fact that the following events cannot take place at the same time: (i) v_l or a vehicle within range of v_l transmit, (ii) v_l or a vehicle within range of v_l receive, (iii) an RSU that is a neighbor of v_l transmits.

As far as RSUs are concerned, we still have to impose that the total duration of the transmissions by a generic RSU r_i cannot exceed one time step:

$$\sum_{(r_i^k, v_l^k) \in \mathcal{E}_r^k} \sum_{c \in \mathcal{C}} \frac{f_c(r_i^k, v_l^k)}{b(r_i^k, v_l^k)} \leq 1 \,.$$

In conclusion, at each time step, each RSU r_i formulates an optimization problem aimed at maximizing (1) under the above constraints. The solution of the problem yields the optimal prefetching and scheduling decisions, based on the prediction $\mathcal{P}_i(u, H)$. Since all constraints are linear expressions with respect to the control variables f_c's, the problem falls in the LP category, hence it can be efficiently solved in real-time.

V. Content delivery through V2V relaying

When the solution of the LP problem leads an RSU to schedule transmissions to relays, the latter are in charge of delivering the data to the downloaders. We envision two approaches to manage V2V data relaying, detailed next.

RSU-driven relaying. The solution to the optimization problem formulated by each RSU, as described in Sec. IV-B, implicitly schedules relay-to-downloader data transfers in addition to RSU-to-downloader and RSU-to-relay ones. Such a scheduling is optimal with respect to the contact prediction available at each RSU and the requests it is aware of, and it can be easily leveraged to drive V2V transfers. To that end, it is sufficient that, based on the contacts they foresee, RSUs provide the relay vehicles with the identity of the downloaders the data are intended for, as well as the expected contact times. Relays will then use this information to decide on when to establish a V2V connection with a given downloader.

Clearly, the performance of this approach highly depends on the prediction accuracy. Uncertainty in the contact estimation can lead either to failure in delivering the data, if a foreseen V2V link turns out not to be established, or to a waste of opportunities, if an exploitable V2V contact is not predicted. Also, the scheduling computed by different RSUs may result to be incompatible, since they are generated from different TEG-PWs: this leads to unexpected channel contention and consequent delays, or impossibility to deliver all data.

Greedy relaying. A dual approach to the RSU-driven relaying consists in letting V2V transfers take place in a greedy fashion, by exploiting any opportunity to make incomplete downloads progress. In this case, the LP problem is only employed to take prefetching and I2V transfer decisions at the RSUs, while relays and downloaders autonomously manage V2V transfers. The greedy relaying protocol we adopt involves three phases and is repeated periodically.

In the first phase, each downloader advertises the list of contents it is currently downloading, detailing, for each of them, the amount of data it needs to complete the transfer. As shown in Fig. 3, a generic downloader v_m will thus announce, at step k and for each incomplete content c, the quantity $M_{m,c}^k = s_c \cdot \left(1 - \sum_{i=t_{m,c}}^{k-1} \phi_{m,c}^i\right)$. The missing data information broadcast by downloaders is received by relays within range. This phase requires loose synchronization (with accuracy of the order of 1 ms) among nearby vehicles, which can be easily obtained through, e.g., GPS, and is already foreseen in the current standards for vehicular networks.

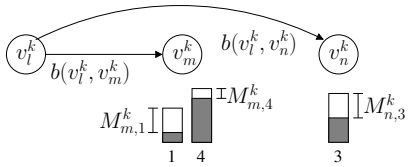


Fig. 3. Greedy relaying example. In phase 1, downloaders v_m and v_n have incomplete contents 1, 4 and 3, respectively, and announce the missing data. In phase 2, relay v_l, storing all missing data, allocates its airtime to satisfy the requests by v_m and v_n, adopting a water-filling approach.

In the second phase, each relay filters the missing data requests received from downloaders in its neighborhood, only retaining those for contents it actually stores. Then, based on the SNR computed on the received broadcast transmission, it estimates the link data rate b, hence the time needed to complete each of the retained transfers.

A relay then decides how to serve the requests, by formulating and solving a max-min fairness problem. The rationale behind such a choice is that a max-min fair allocation of the airtime allows downloads to progress evenly, not favoring large downloads over small ones or vice-versa, yet guaranteeing that the medium is fully exploited. Also notice that, consistently with our objective (1), we can fully exploit incomplete transfers. A water-filling approach is employed to efficiently solve the problem. Once the locally-optimal allocation is obtained, in the third phase relays start to transmit their data to target downloaders. If multiple relays are neighbors, or hidden terminals to each other, their allocations will have to share the medium according to the constraints on channel access defined in the previous section.

VI. Results

We now detail the mobility and communication scenario we take as a reference and present the impact of the parameters of the fog-of-war model on the contact prediction. The results on content downloading in the reference scenario follow.

A. Reference scenario

We consider a real-world road topology representing a 3×3 km^2 section of the urban area of Turin, Italy, portrayed in Fig. 4. We focus on 30 minutes of consistently fluid traffic conditions, such that, at any instant, the scenario includes about one thousand vehicles simultaneously traveling over the area and taking part in the ITS. The vehicular mobility has been synthetically generated using the SUMO simulator [12]. The time granularity of the resulting mobility trace is 1 s, hence we set the granularity of the traffic manager prediction and the periodicity of the execution of the V2V data relaying

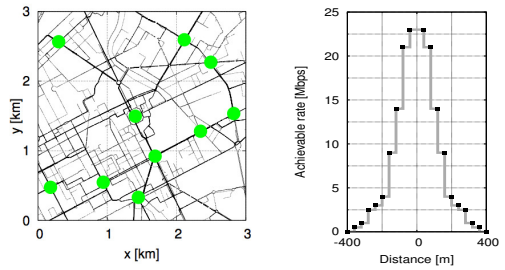


Fig. 4. Road topology (left) and achievable network-layer rate (right).

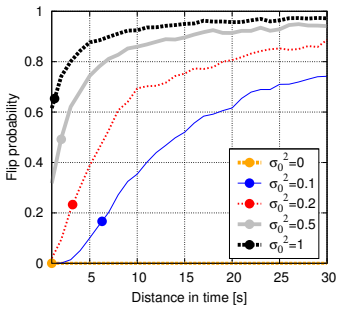


Fig. 5. Contact flip probability vs. the prediction time-span, for varying σ_0^2. Dots represent the average probability value.

protocol to 1 s. We remark that we preferred a synthetic trace over real-world ones, e.g., taxi or bus traces, as these only include a small portion of the car traffic and have a time granularity too coarse for the resulting TEG-PW representation to be effective.

Fig. 4 also depicts the default deployment that we assume for the ITS infrastructure, with 10 RSUs located at the most crowded intersections, represented by green dots. Based on the findings in [7], such a placement strategy allows ITS-based downloading to perform close to the optimum.

With regard to the communication technology, we assume that the nodes use an 802.11-based protocol, and that rate adaptation is employed. It follows that the value of the achievable network-layer rate between any two nodes is set according to their distance. In particular, we refer to the 802.11a experimental results in [13, Fig. 5] to derive the values shown in Fig. 4, and we use them as samples of the achievable network-layer rate. Also, we limit the maximum radio range of any node to 200 m, since, as stated in [13], this distance allows the establishment of a reliable communication in 80% of the cases.

As for the cellular network, we assume that full cellular coverage of the area is available. A user can always complete its download through the cellular infrastructure if it could not retrieve the whole content through the ITS within T seconds. Unless otherwise specified, we set T=120 s.

Users' content demand is modeled by assuming that $|\mathcal{C}|$=100 items are available and have the same size s_c=10 MBytes. The per-user request rate is Poisson distributed with rate $\lambda = 0.005$. When location-specific content is considered we identify the vehicular flows in the mobility trace, through the κ-means clustering algorithm. Then, identical demands characterize vehicles belonging to the same flow.

Finally, we assume that the traffic manager generates its predictions every 30 seconds, forecasting the next 30 seconds of contacts. Since δ=1 s, this implies H=30 in the following.

B. Behavior of the fog-of-war model

The impact of the fog-of-war model parameters, in the above reference scenario, is shown in Fig. 5.

The left plot presents the probability of contact flip, i.e., that an actual contact is removed and a spurious one is created, as a function of the time at which the contact should begin (i.e., $k-u$, u being the step at which the prediction is compiled). The curves are obtained for different values of σ_0^2, with $\sigma_0^2 = 0$ corresponding to a flawless prediction. As expected, the larger the σ_0^2, the higher the probability to predict spurious contacts. Also, the time span $k-u$ has a significant impact, as contacts established farther in the future become less predictable and are affected by a higher flip probability.

In the plot, the dots on the curves represent the flip probability computed over all actual contacts observed within the prediction horizon. Note that for $\sigma_0^2 \geq 0.5$ the majority of predicted contacts are spurious, while for $\sigma_0^2 = 0.1$ we have a quite reliable prediction (about 4 out of 5 actual contacts are correctly forecast). This is due to the fact that contacts already existing at step u are associated with a null distance in time, hence they are always correctly predicted.

C. Performance of content downloading

We evaluate the effectiveness of offloading content download from cellular to ITS networks, in the reference scenario previously described.

We first assume (i) a content demand process where each content is requested by vehicles with equal probability, (ii) unlimited time validity for contents, and (iii) $B = 100$ Mbit/s, i.e., high-bandwidth links connecting the RSUs with the content servers. Note that this essentially implies ideal ITS operation, as RSUs need to download contents only once, thanks to their unlimited cache size and the infinite content validity. We refer to this scenario as our *baseline* system configuration, and employ it to study the impact on the download performance of: forecast accuracy, V2V relaying strategy, timeout for the ITS data retrieval and ITS infrastructure dimensioning. The rationale is that the baseline scenario allows us to study the wireless portion of the system, while avoiding biases due to the demand distribution or to backbone limitations.

As a second step, we relax the assumptions on the RSU content retrieval operation, content demand and validity, and investigate: (i) a *constrained* system configuration, where the content validity is limited in time and the RSU backbone bandwidth is reduced; (ii) a *location-specific* system configuration, where the content requested by vehicles is influenced by the traffic flow they belong to. The latter configuration also allows us to compare the offloading performance obtained with a contact prediction with that achieved by a forecast-agnostic, push-based scheduling scheme based on content popularity.

Baseline scenario. The performance of the offloading process in the baseline scenario is presented in Fig. 6(a), which

portrays the average fraction of requested content that a vehicle can successfully download through the ITS before the expiration of the timeout T. The results have been obtained as σ_0^2 varies, under the greedy relaying scheme.

The offload fraction is broken down into content retrieved directly from RSUs and content obtained from relays through V2V communication, and it is compared against the ideal offload performance. The latter is derived by solving the optimization problem for $\sigma_0^2 = 0$, a very large prediction horizon (namely, $H = 300$) and assuming that future user requests are known a priori; this enables perfect I2V scheduling.

Firstly, we observe that ITS can relieve the cellular network of 70-80% of the cost associated to content download. Secondly, a great part of the merit goes to V2V relaying, bearing between 30 and 60% of the content transfer effort, which confirms that opportunistic transfers are highly beneficial in the offload process. Thirdly, the overall performance is not too far from the ideal one, which would allow a 90% offload.

The impact of the accuracy of the contact prediction is shown by varying σ_0^2. Quite surprisingly, very accurate predictions (low values on the x axis) result in a performance that is just slightly better than that scored by almost random contact estimations (high σ_0^2's). Inaccurate predictions lead however to a reduced contribution of V2V with respect to I2V transfers, as the former drops from more than 75% to less than 40% of the overall offloaded fraction.

The actual cost of an imprecise contact prediction is revealed by Fig. 6(b), which shows the offload efficiency, i.e., the ratio of the amount of data delivered to a downloader to that transmitted by the RSUs (to either downloaders or relays). A low efficiency implies a waste of wireless resources at the RSUs, while a high efficiency means that only useful ITS-based transfers are performed. The efficiency can be higher than one, since a relay can download some content (or part of it) and then provide it to multiple downloaders. The plot clearly shows that, in order to maintain high offload fractions, the less precise the information on future contacts, the larger the amount of data the RSUs have to transfer to relays.

Another interesting fact underscored by Fig. 6(b) is that RSU-driven relaying consistently performs worse than the greedy approach. The reason for such a behavior is that the amount of data transmitted by the RSUs is the same in the two cases, but the former is unable to exploit data transfers to future downloaders (of which RSUs are unaware). This is an important contribution to the performance, unlike the optimized RSU-driven scheduling that is beneficial only in the rare case of multiple, simultaneous relay-downloader transfers. As a consequence, the greedy approach is to be preferred and we will focus only on it in the following.

Fig. 6(c) further details the offload performance, showing the cumulative distribution function (CDF) of the fraction of content that each downloader can retrieve through the ITS. Results are shown for quite accurate ($\sigma_0^2 = 0.1$) and rather imprecise ($\sigma_0^2 = 1$) predictions, and benchmarked against the ideal case. The CDFs clearly identify two larger classes of downloaders: those that can get a very small percentage (possibly zero) of the data they request, and those (over 50% of the total) who can obtain almost all (80% or more) of the data through ITS. Interestingly, the latter category does not seem to be affected by σ_0^2, as the curves are very close for high values on the x axis. On the contrary, the percentage of downloaders unable to get any data is sensibly reduced as the contact estimation precision grows. We can thus conclude that an accurate prediction is most useful to offload downloads for hard-to-reach users, e.g., those traveling on secondary roads.

Finally, Fig. 6(d) portrays the CDF of the delay in the ITS-based content delivery. A large portion of the data, amounting to 70% of the content size, is typically obtained within a short timespan (approximately 20 s). The results are similar in presence of ideal and precise contact predictions, although in the ideal case the higher fraction of downloaded contents leads to an increased latency for users on unfavorable routes. An inaccurate contact prediction, instead, yields higher delays.

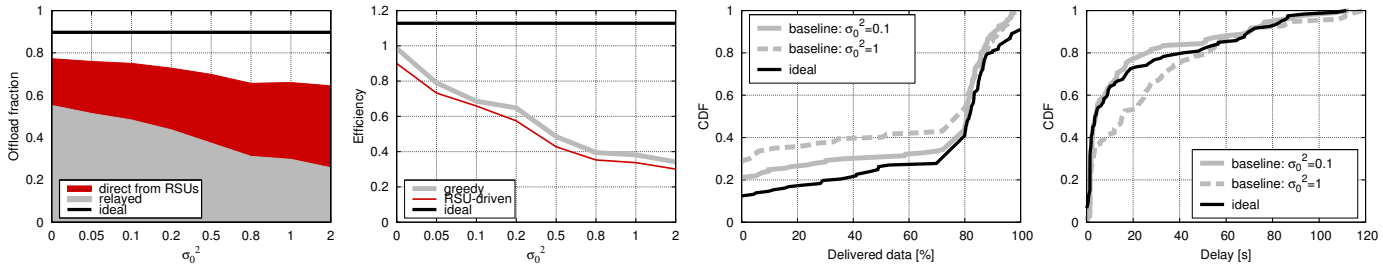
Tab. I shows the offload fraction for varying σ_0^2 and number of deployed RSUs. As expected, increasing the number

TABLE I
OFFLOAD FRACTION AS THE NUMBER OF RSUS AND σ_0^2 VARY

σ_0^2 \ No. RSUs	6	8	10	12	14	16
0.1	0.55	0.67	0.76	0.79	0.92	0.94
1	0.48	0.57	0.66	0.71	0.82	0.84

TABLE II
OFFLOAD FRACTION AS THE TIMEOUT T AND σ_0^2 VARY

σ_0^2 \ T [s]	60	120	180	240
0.1	0.69	0.75	0.78	0.80
1	0.58	0.66	0.71	0.72



(a) Offload fraction (b) Offload efficiency (c) Offload distribution (d) Delay

Fig. 6. Content download performance in presence of cellular network offloading via ITS-based communication, under the baseline system configuration.

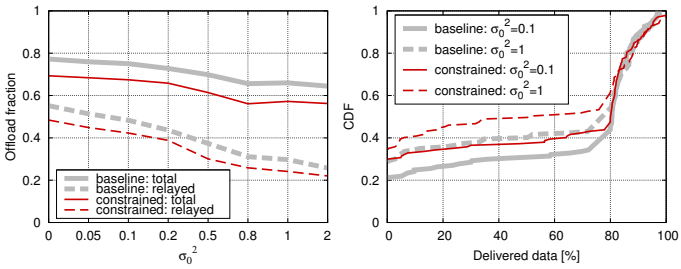


Fig. 7. ITS-based download performance in the constrained configuration.

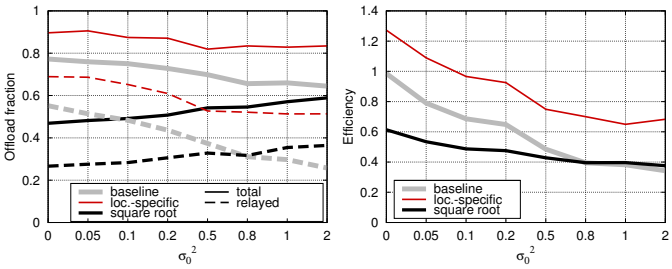


Fig. 8. Impact of location-specific contents on ITS-based download performance and comparison against a content popularity-based approach.

of RSUs favors the ITS-based offloading process. However, improving future contacts estimation can compensate for a less pervasive ITS coverage. Indeed, by cross-checking similar offload fractions over different columns, we note that an accurate prediction requires between 20 and 30% less RSUs, while maintaining similar performance.

The benefits of an accurate prediction are also shown in Tab. II, which reports the offload fraction for different values of σ_0^2 and T (the time after which users start retrieving data from the cellular network). Indeed, the higher the T, the larger the amount of data downloaded through the ITS, however improving the forecast reliability pays significantly more than delaying the use of the cellular network.

Summary: Our results show that ITS is a viable alternative, or complementary solution, to cellular networks for content downloading by mobile users. In particular, if a relatively reliable mobility prediction is available, the offload of the cellular infrastructure can be achieved by sparing wireless resources, better serving downloaders on secondary roads, reducing the download latency, and lowering the ITS deployment cost. Furthermore, an imprecise mobility prediction has a relatively small impact on the actual offload fraction, but it significantly impairs the system efficiency.

Constrained scenario. Here, we focus on the case of RSU backbone links with bandwidth limited to $B = 10$ Mbps and contents expiring after an exponentially distributed time with mean equal to 200 s. We remark that the latter condition forces, upon expiration of a content, both RSUs and downloaders to discard any portion of the content they previously obtained, and restart the download from scratch.

The offload fraction obtained in such a constrained configuration is presented, and compared to our baseline, in Fig. 7. More precisely, the left plot shows the average offloading fractions as σ_0^2 varies, while the right one details the per-downloader CDF of the offload fraction. The first plot clearly evidences that the introduction of the constraints leads to an unchanged trend with respect to the contact prediction accuracy, at the cost of a performance reduction. Interestingly, the performance drop mainly concerns the download via V2V relaying, since, upon expiration of the content, relays have to discard the data and cannot help in the delivery any longer. In the second plot, we can once more observe how less performing network operations affect downloaders on unfavorable routes (e.g., traveling on secondary roads).

Location-specific content scenario. We now evaluate the offload performance in presence of location-specific content. Users belonging to the same vehicular flow request contents according to the same Zipf's distribution with exponent 2, and the distributions corresponding to different vehicular flows are shifted by 20 contents with respect to each other. Vehicles not belonging to any vehicular flow request contents with equal probability.

Fig. 8 shows that, in presence of location-specific content, the amount of data the downloaders can retrieve through the ITS significantly increases, mostly due to V2V relaying. Indeed, thanks to the tighter correlation between the vehicles' routes and the content they request, it is likely that the desired information is obtained from nearby vehicles. This also explains the very high efficiency of relayed traffic in the right plot of Fig. 8.

The plots also portray the performance of an approach based on content popularity that exploits knowledge of the popularity distribution, instead of the mobility forecast. Specifically, it lets RSUs select the content to be pushed towards a relay, with a probability proportional to the square root of the content popularity [14]. For fair comparison, we force the amount of data sent by the RSUs to match what is observed in our prediction-based scheme with location-specific content.

The results show that predicting the contacts allows for significantly better performance than the knowledge on the content popularity. This is due to the high mobility of our scenario: either the content is delivered to the right vehicular flow, or retrieving the content from a vehicle carrying the data becomes very hard. Such an observation is confirmed by the curve referring to relayed traffic in the left plot, which is significantly lower for the square root approach than in the predicted-based scheme. As a last remark, the offload fraction in the square root case grows with the increase of σ_0^2, since, for fair comparison, the RSUs inject more data in the network to match the amount observed in the location-specific case. Nevertheless, such an increase in the delivered data does not make up for the higher radio resource consumption, thus leading to a lower efficiency.

VII. Related work

A few works have studied scenarios where opportunistic transfers and cellular technologies coexist, so as to offload the infrastructure through user-to-user communication. However, the problem they address is not that of content downloading, but the dissemination of some data item to all mobile

users. Thus, rather than scheduling the transfer of large-sized heterogeneous contents, the problem becomes that of determining how many copies of each item shall be injected in the network and which users are most suitable to receive them. Among these works, [4] considers vehicular users, while [5] deals with more generic smartphone-equipped cellular network customers. Clearly, solutions designed for dissemination offloading cannot be applied to our context.

Content downloading is instead the target of [8]. There, however, only I2V direct transfers are considered, and the focus is on the prefetching of contents at RSUs, which are assumed to have high-latency, low-bandwidth links to the Internet. The objective is then to optimize the usage of such links, by estimating the amount of traffic the vehicles will be able to download from each RSU. Moreover, in [8] the use of the cellular infrastructure is limited to signaling purposes.

Works such as [15] investigate content downloading through publicly available WiFi hotspots, and the link between acceptable delay and offloading fraction. Note that, unlike previous works, we study content downloading in ITS accounting for all communication methodologies, i.e., I2V, V2V, and cellular-based, at a time. This allows us to jointly investigate the problems of content prefetching at RSUs, scheduling of I2V transfers, and management of opportunistic V2V transfers.

The approach we adopt relates our work to the problem of transmission scheduling in wireless networks, which has been widely studied. However, most works address the case of connected multi-hop networks, e.g., [16], or social delay-tolerant networks, e.g., [17]. The vehicular environment mixes elements of both, thus solutions that assume full reachability or contacts periodicity [18] in the order of hours or days do not apply to our context. A scheduling and prefetching scheme for content downloading in vehicular networks is presented in [19]. This work, however, employs simplistic mobility models and does not consider the presence of a cellular infrastructure. As further additions to the literature on transmission scheduling to vehicles, we take into account, for the first time, the role that mobility-based communities have in the generation of content demand, and evaluate the impact of uncertainty in the estimation of future I2V and V2V contacts.

Concerning the latter aspect, there are several ongoing efforts on inferring future vehicular contacts, given the current position and past car trajectories [9]. The two main approaches consist in modelling the vehicle location through a Markovian process, and in studying the time and duration of contacts. Thanks to our fog-of-war model, our system can use any of these techniques, or future, more accurate ones, as an input.

Finally, the representation of a time-varying network as a time-expanded graph has also been employed in our previous work in [7]. Beside the different scope, the time-expanded graph we propose here significantly differs from the above representation as we introduce probabilistic edge weights, in order to model uncertainty in the prediction of inter-node contacts.

VIII. Conclusion

We addressed the problem of cellular network offloading through ITS content download. We studied content prefetching and data transmissions scheduling from roadside units in the realistic case of finite-horizon, inaccurate mobility prediction. We showed that, if the prediction error is not overwhelming, vehicles can be effectively served by the ITS, through either direct download from RSUs or relaying, thus relieving the cellular network from the download traffic. The offload efficiency was close to an ideal case and significantly better than that of a content popularity-based solution. Further benefits can be garnered in presence of location-specific content, requested by vehicles traveling on the same roads.

Acknowledgments

This work was supported by the European Union through the FIGARO project (FP7-ICT-258378) and by Regione Piemonte through the IoT_|_ToI project.

References

[1] U. Paul, A.P. Subramanian, M.M. Buddhikot, S.R. Das, "Understanding traffic dynamics in cellular data networks," *Infocom*, 2011.
[2] R. Pries, F. Wamser, D. Staehle, K. Heck, P. Tran-Gia, "Traffic measurement and analysis of a broadband wireless Internet access," *VTC09 Spring*, 2009.
[3] S. Dimatteo, P. Hui, B. Han, V.O.K. Li, "Cellular traffic offloading through WiFi networks," *IEEE MASS*, 2011.
[4] J. Whitbeck, Y. Lopez, J. Leguay, V. Conan, M. Dias de Amorim, "Relieving the wireless infrastructure: When opportunistic networks meet guaranteed delays," *WoWMoM*, 2011.
[5] B. Han, P. Hui, V.S.A. Kumar, M.V. Marathe, G. Pei, A. Srinivasan, "Cellular traffic offloading through opportunistic communications: a case study," *CHANTS*, 2010.
[6] I. Trestian, S. Ranjan, A. Kuzmanovic, A. Nucci, "Taming user-generated content in mobile networks via drop zones," *Infocom*, 2011.
[7] F. Malandrino, C. Casetti, C.-F. Chiasserini, M. Fiore, "Content downloading in vehicular networks: What really matters," *Infocom Mini-Conference*, 2011.
[8] S. Yoon, D. T. Ha, H. Q. Ngo, C. Qiao, "MoPADS: A mobility profile aided file downloading service in vehicular networks," *IEEE Trans. on Veh. Tech.*, vol. 58, no. 9, 2009.
[9] H. Zhu, S. Chang, M. Li, S. Naik, S. Shen, "Exploiting temporal dependency for opportunistic forwarding in urban vehicular networks," *Infocom*, 2011.
[10] TomTom, "How TomTom's HD TrafficTM and IQ RoutesTM data provides the very best routing," *White paper*, 2010.
[11] F. Malandrino, C. Casetti, C.-F. Chiasserini, M. Fiore, "Representing a Markovian prediction technique through a fog-of-war model," http://www.telematica.polito.it/malandrino/techrep-markov.pdf, 2011.
[12] C. Barberis, G. Malnati, "Epidemic information diffusion in realistic vehicular network mobility scenarios," *ICUMT,* 2009.
[13] D. Hadaller, S. Keshav, T. Brecht, S. Agarwal, "Vehicular opportunistic communication under the microscope," *MobySys,* 2007.
[14] E. Cohen, S. Shenker, "Replication strategies in unstructured peer-to-peer networks," *Sigcomm,* 2002.
[15] A. Balasubramanian, R. Mahajan, A. Venkataramani, "Augmenting mobile 3G using WiFi," *MobiSys*, 2010.
[16] R. L., Cruz, A. V. Santhanam, "Optimal routing, link scheduling and power control in multihop wireless networks," *Infocom*, 2003.
[17] W. Gao, G. Cao, "User-centric data dissemination in disruption tolerant networks," *Infocom*, 2011.
[18] U. G. Acer, P. Giaccone, D. Hay, G. Neglia, S. Tarapiah, "Timely data delivery in a realistic bus network," *Infocom Mini-Conference*, 2011.
[19] B. B. Chen, M. C. Chan, "MobTorrent: A framework for mobile Internet access from vehicles," *Infocom*, 2009.