Mobility Pattern Prediction to Support Opportunistic Networking in Smart Cities

Alessandro Morelli, Cesare Stefanelli, Mauro Tortonesi
University of Ferrara
Ferrara, Italy
{alessandro.morelli, cesare.stefanelli, mauro.tortonesi}@unife.it

Niranjan Suri
Florida Institute for Human and Machine Cognition
Pensacola, FL USA
US Army Research Laboratory
Adelphi, MD
nsuri@ihmc.us

Abstract—The ever increasing number of mobile devices in Smart Cities and their heavy use, not only for personal communication but also as a distributed network of sensors, generate a data deluge that stresses the traditional wireless communication infrastructure. The opportunistic networking paradigm seems particularly well suited to the Smart City scenario because it exploits resources that temporarily fall into the connection range of mobile devices as communication proxies, thereby providing cheaper and more energy efficient alternatives to the use of the cellular city network and actively contributing to its offloading. However, its efficacy highly depends on the effectiveness of discovering and using those resources. To improve the effectiveness of opportunistic networking in Smart Cities, we propose a solution which exploits a prediction model tailored for the urban environment that, by detecting complex recurring patterns in nodes' contacts, can forecast the future availability of strategic communication resources. Experimental results obtained in a simulated environment show that our solution can improve the dissemination process and ease the access to the wired network infrastructure.

Keywords—Smart City; opportunistic networking; mobile data offloading; prediction model; communication middleware

I. INTRODUCTION

The number of people living in the cities worldwide has been in the rising trend since way before the Internet era, and studies state that the urban population will almost double by the middle of the 21st century [1]. This incessant growth places new challenges to the city management under many points of view. For instance, new plans and strategies are required to assist both the public and the private transportation to meet the new requirements, new power grid infrastructures are necessary to distribute and control power resources in a smarter and more adaptable manner, and the public safety and the public health services need more efficient strategies and techniques to deliver information and data in real-time.

The concept of Smart City has emerged to address these challenges, describing a modern urban environment where the Information and Communication Technology (ICT) plays an essential role as the provider of means and techniques to effectively access and exploit the other assets of a city, such as its social and economic capitals. Many actors are making a pervasive and intensive use of ICT techniques to realize effective and sustainable solutions that aim to improve the quality of life of the smart citizens in many different areas [2] [3] [4]. The efficient gathering, processing, and dissemination of data are essential to implement all these new services in dynamic and heterogeneous environments such as the Smart City.

The ever-growing density of smartphones and tablets in the modern cities, their pervasiveness among the population, and their availability on the urban territory at no cost for the public administration make them extremely valuable resources. In fact, they have the potential to serve as sensors and collect many different types of environmental data, as well as the computational and the connectivity capabilities to implement non-trivial, effective dispatching strategies. Nevertheless, the mobility characteristic of these devices, which are carried around by the citizens as they move, makes it impossible to rely on the wired network infrastructure to reach the required levels of connectivity. However, due to the enormous growth in the mobile data traffic expected for the next years, the cellular network will be unable to satisfy, by itself, the future demand. For these reasons, the study of effective solutions to achieve *mobile data offloading* is becoming more and more important in the scientific community [5] [6].

Opportunistic Networking techniques have recently emerged to face many of the challenges that occur in Smart Cities. The Opportunistic Networking paradigm comes from the networking concepts that naturally emerged in the Mobile Ad-Hoc Networks and Delay Tolerant Networks research fields and evolved into a more complex and effective set of networking strategies and protocols. Differently from other solutions, this paradigm takes into account the information coming from both the application and the environment contexts. By analyzing the aspects in human interactions [7] or in mobility patterns [8], or both the context and the content of exchanged messages [9], applications that rely on Opportunistic Networking can maximize the effectiveness in reaching their goals, under the constraints provided by the current network status and by the communication means.

Given the peculiarities of the urban environment and the challenging requirements of smart applications, it seems natural to try to take the maximum advantage out of the Opportunistic Networking paradigm by exploiting the periodicity typical of the mobility patterns of some nodes (e.g., buses, trains, and metros, but also commuters, who might drive every day the same path to work) in order to support the data forwarding from the mobile sensors and devices to the wired network. In particular, this paper presents a sophisticated approach to *discover a wide spectrum of complex periodic patterns* that recur in the contacts with mobile nodes. In fact, in order to satisfy the needs of the citizens, the routes of public means of transportation might change periodically, for instance to adapt to congestion, to connect important areas of the city more frequently, or to serve different neighborhoods. Being able to detect these types of periodic patterns allows the *prediction of future contact opportunities* with nodes that represent potential communication resources, thus enabling a smarter usage of the scarce resources of the network.

We implemented our pattern detection algorithm at the application middleware level, to facilitate reuse and maintainability of the forecasting system and to decouple the application logic from all the strategies and functionalities that support the opportunistic discovery and management of the available resources (routing, forwarding, message caching, prediction models, etc.). Experimental results run in a simulated environment show that our algorithm can effectively detect contact patterns with mobile nodes. This allows the design of advanced information dissemination strategies that favor the usage of alternative, cheaper communication solutions, such as Wi-Fi or Bluetooth, available in the majority of the modern mobile devices. Results of performed simulations proved that these dissemination strategies can ease the access to the wired network infrastructure by means of connectivity interfaces which do not include the cellular network, thereby significantly contributing to its offload.

II. OPPORTUNISTIC NETWORKING IN SMART CITIES

Many recent studies focused on the importance of Opportunistic Networking as the communication paradigm to effectively address the challenges typical of heterogeneous, dynamic, and resource constrained scenarios [10] [11].

In the Smart City, several nodes in the network need to exchange high quantities of heterogeneous traffic. However, those nodes might not be able to directly access the wired network layer, or they could not be in range of any Wi-Fi or WiMax AP, causing the network to be partitioned. At the same time, the cellular network might not be an option, as the generated traffic could be excessive, the network might be congested, or sending data over it might be too expensive. Therefore, to be able to fulfill their goals to the best of their possibilities, applications running in Smart Cities have to consider other connectivity resources.

The mobility and the variety of nodes, such as user mobile devices, sensors, and vehicles, which can all be equipped with (at least) a network interface, and the widespread availability of free Wi-Fi APs are typical characteristics of Smart Cities. They identify a very heterogeneous network, where connections between nodes are mostly unstable. Such environment challenges applications, which need a solution that enable them to discover and exploit new connectivity resources quickly and effectively. Relying on the Opportunistic Networking paradigm would allow applications to maximize the benefits they obtain from nodes that temporarily fall under the connection range.

In Opportunistic Networks, applications need to adaptively apply the most appropriate dissemination strategies, choosing between many forwarding and replication protocols, communication semantics, and available connection technologies. To this end, applications have to take into account both their goals and the constraints placed by the environment.

Fig. 1 below shows a Smart City scenario. Surveillance applications are installed on sensor nodes equipped with a camera to take high resolution pictures of the current traffic conditions in some critical areas of the city. To get the most out of this kind of applications, the captured images need to be gathered and stored in a data center, usually located in the cloud, where enough computational and memory resources are available to process them and derive useful information. We could imagine the cameras connected directly to the data center via powerline or 3G communications. However, powerline communications might be difficult to deploy, and 3G communications could suffer the problems we already described for cellular networks.

An interesting option to improve nodes' connectivity is to opportunistically take advantage of mobile nodes that come into proximity and that could operate as message ferries between the cameras and one or more "sink" nodes connected to the data center that manages the Smart City information layer. A possible solution could be, for instance, to provide the public transportation vehicles with Wi-Fi or Bluetooth devices, so that the camera nodes can use either one of the two technologies to send the images to buses and trains passing by. Those vehicles will then carry the images to one or more sink nodes connected to the Smart City data center. Notice that also bus passengers could exploit the proximity to a sink for uploading heavy data contents, like social network activities, videos, or high quality images, thereby avoiding to connect to the more expensive cellular network and contributing themselves to its offload.

The issues described in the previous paragraphs call for an *adaptive communication middleware designed for Opportunistic Networks*, capable of analyzing the current network conditions and of exploring all the surrounding connection opportunities, to support the overlying applications. This middleware will tailor the dissemination strategy based on the discovered connection opportunities, under the constraints which characterize each device, and will provide applications with a set of mechanisms and tools to define policies to match their goals.

An adaptive communication middleware needs to have a *complete and accurate representation of the network status and its resources* in order to be able to satisfy the application requirements. However, often the knowledge about the current state of the system is not enough. For example, it is possible that a node, which is not currently reachable, will soon fall under the Wi-Fi range of another device. This would open new connection possibilities in the near future, although currently unknown. Therefore, to provide applications with all the information to design effective policies, the communication

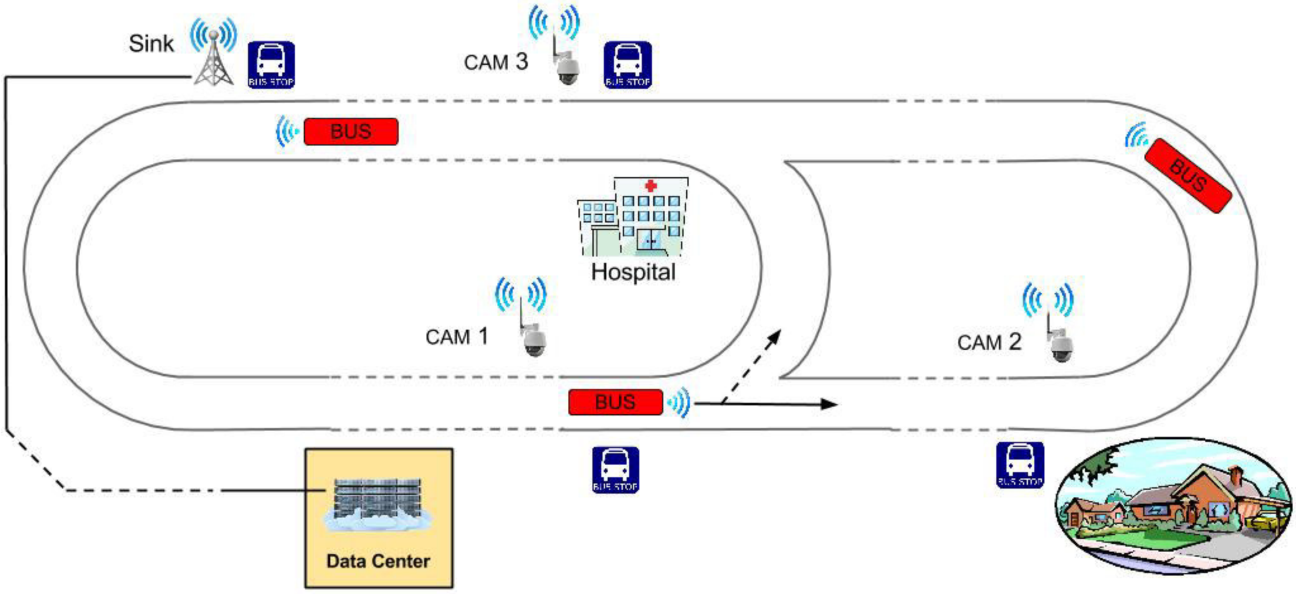

Fig. 1. A Smart City Opportunistic Networking Scenario

middleware should implement techniques capable of *predicting the presence of future resources*, whenever possible.

Having the knowledge of future contacts with other nodes at their disposal, applications can implement disruption tolerant policies. Also, the availability of predictions enables the design of policies that foster a fairer usage of the available resources. For example, prioritizing short-medium range communication technologies, like Wi-Fi or Bluetooth, against other types of solution, such as 3G communications, would assure a higher utilization of cheap wireless connectivity solutions, thus offloading the cellular network and leading to an improvement of the global performances of the Smart City network [6]. To the best of our knowledge, there is no communication middleware designed for Opportunistic Networks that features a prediction model to forecast the future contact times with other nodes of the network.

III. Prediction of Future Contacts for Effective Information Dissemination

In order to make the best decisions when it comes to opportunistic routing, applications require a knowledge of the environment in which they are submerged that has to be as complete as possible. The type of knowledge required comprises the set of nodes available within communication range, their characteristics, the connectivity resources capable of reaching them, and the network status, as well as any requirement that the applications might have in terms of bandwidth allocation, maximum latency, transmission reliability, set of destinations, etc.

Although this information is necessary for applications to select the best routing strategies, considering only the present conditions of the network might limit the output of the dissemination algorithm to a local optimum. In fact, in highly dynamic environments and under certain conditions, delaying the delivery of messages might open the door to better communication solutions. However, systematically delaying the dispatch of all messages to try to take advantage of better communication possibilities would place the risk of extremely increasing the latency of the information dissemination; this is unacceptable in some application domains and, anyway, never a desirable property.

To delay the messages delivery only when convenient, the communication middleware has to provide applications with the knowledge about future contact times with potential communication resources. In order to do so, the middleware can exploit the history of past contacts with the communication resources to build a forecast model which is able to infer the next contact times.

The computation of forecasts of future contacts with other nodes based on the history of past contacts is computationally expensive and, also, a challenging task. In fact, nodes can exhibit very complex periodic behaviors and so the process of discovering the patterns that underlie them might be very expensive. Also, forecast models need to be continuously reevaluated to keep their accuracy within a certain level. Finally, there is the need to provide applications with an evaluation of the reliability of the forecasts [12], so they can autonomously decide if and when to rely on the offered predictions.

For example, let us consider the bus route depicted in Fig. 1. The itinerary might have been conceived to prioritize the connections with certain areas of the city against others, situation which is not uncommon in modern urban realities. As a consequence, the bus might follow an itinerary which is not always the same, but varies accordingly to a predefined schedule. In the figure, the different paths are represented with two arrows: the dashed arrow represents the shortest path, whereas the normal arrow represents the longest one. To connect more frequently the most important served areas of the city (to the left in the figure) to the hospital, the route could be designed in such a way that the bus will take the shortest path twice in a row, before taking the longest path once, and then it will start over, repeating the same pattern.

While an unsophisticated forecast model would need less computing resources, it would also fail to recognize many common patterns in nodes mobility, or it would reach lower, possibly inadequate, levels of accuracy. For instance, a model which assumes that the nodes will follow a constant itinerary between consecutive contacts would fail to capture the behavior which characterizes the bus node in the scenario described above.

Nonetheless, a completely different approach, based on accessing the Internet to download the timetables of bus lines which pass by the camera, would present other problems. In fact, Smart Cities might have smart bus systems available that can use the cellular network to provide all the interested nodes (traffic cameras, bus stops, traffic lights, etc.) with the information about the next arrival time of one or more buses. However, all the traffic generated to periodically distribute and update this information to all the nodes would place a great burden on the cellular network and contribute to its congestion. An interesting possibility to reduce the traffic could be to limit the number of update messages to only one message, which notifies when a bus leaves the closest bus stop. With this information, a camera node nearby would simply have to learn the amount of time required for the bus to reach it. These values can then be crossed with the times at which the update messages were sent, to capture fluctuations in travel times due to varied congestion levels at different moments of the day/week. Finally, note that data gathered for distinct buses which travel the same paths to reach a camera could be merged to reduce memory usage and to increase the accuracy of the predictions.

For the reasons expressed above, there is the need for advanced prediction models that can recognize complex recurring patterns in the nodes' mobility, leading to solutions which can perform well under many circumstances. However, the limited memory and computational resources available on sensors and mobile nodes require a trade-off between the accuracy, the refinement, and the complexity of the forecast model. The chosen trade-off can vary based on the characteristics of the device. Alternatively, the middleware might provide applications with a set of multiple models, each with different complexities and characteristics. In turn, the applications will be responsible for choosing the model which best satisfy their requirements.

In this work we propose a *general, middleware-based solution* that, paying the cost of a more complex elaboration than the one necessary for simple approaches such as those described above, implements a model which can detect a broad spectrum of *periodically recurring patterns in nodes' mobility*. We believe the patterns that our solution can detect are realistic representations of those which characterize the intrinsic periodic behavior of many subjects of the Smart City, such as the public means of transportation. The proposed solution exploits a mathematical approach which allows to search for and discover periodic patterns while keeping the computational complexity of the model relatively low.

IV. An Efficient Mobility Prediction Model for the Urban Environment

In a modern city, the intrinsic periodic behavior of public transportations allows us to approach the problem of detecting periodically recurring mobility patterns of nodes from a simpler perspective. In fact, public means of transport equipped with a medium-range network device such as a Wi-Fi card, or with a small-range, low-power Bluetooth interface, can become mobile nodes with a very predictable behavior.

Most of means of transportation either have a fixed schedule throughout the day (that is, the inter-arrival time at the same destinations stays constant), or they move according to a certain constant pattern that repeats itself with some periodicity (several times a day, daily, weekly, etc.). These observations reduce the complexity of the problem of finding predictable patterns in the nodes' behavior, as we can assume the existence of periodically recurring patterns that underlie the intercontact times between two nodes. In addition, we can consider that discovered patterns will not change in the short period, since bus and train schedules and routes tend to remain unvaried for a long time, usually months or years. In this paper we address the latter case, whereas the former is just a special, simpler case.

In a Smart City, there are several categories of nodes which could take advantage of predictions about future contacts with other nodes. For example, the surveillance application described in section 2 could use predictions to implement a smart information dissemination policy, which aims to increase the ratio of messages sent using cheap, short-medium range communication links, like Wi-Fi or Bluetooth, instead of more expensive ones such as 3G. In fact, the knowledge derived from the prediction model enables informed decisions on whether to send the images via one communication interface or the other, according to both the estimated likelihood that a bus will approach the camera in the near future and with the urgency of the data.

Our middleware implements a prediction model to analyze the sequences of intercontact times collected for each node. It features a mathematical approach which can effectively compute the autocorrelation of a time series, thereby allowing the discovery of periodic patterns in the data. Moreover, the relatively low complexity of our solution makes it appropriate to be employed on devices with low computational resources, such as sensors or smartphones.

The ability to keep track of the nodes' contact history is a key feature of our communication middleware. This way, it gathers the necessary data to feed its prediction model, thereby enabling the forecast of the next contact times with the nodes and the computation of the predictions' reliability. The model can be configured with parameters which specify the maximum tolerance and the minimum accuracy and reliability allowed, so that applications are able to control the quality of the forecasts and change their dissemination strategy accordingly (readers can refer to [12] for a more detailed discussion on these parameters). This permits to design applications which can implement adaptive and sophisticated dissemination strategies, based on the current state of the network and on the information about future contacts with strategic nodes.

We further extended our communication middleware by adding the feature to collect statistics concerning the link duration within the nodes' contact history. Combining this knowledge with the prediction of the next contact time, the middleware can assess the amount of data which will be able to exchange with another node during the next contact window. This feature further increases the adaptability of the middleware, which puts at the overlying applications' disposal an evaluation of the bandwidth available during the next contact with a node, allowing applications to design more robust and refined policies. The investigation of the impact that this functionality might have on the dissemination process is, however, out of the scope of this paper.

In the next three sections we are going to introduce two possible approaches to detect periodically recurring patterns in the nodes' mobility and the algorithm to predict the next contact time with other nodes which is implemented in our middleware.

A. A Straightforward Approach for Detecting Recurring Mobility Patterns

Both the approaches we are going to present in this and in the following section consider the sequence of intercontact times with a certain node η as the finite time-series x_n, where $0 \leq n < N$, and N is the number of intercontact times observed so far. A straightforward way to extrapolate the periodicity of x_n is to calculate its autocorrelation function for some set of predefined lags (with the largest lag that cannot be greater than N/2).

Nodes that identify public means of transportation, as discussed in the previous chapters, show a regular, recurring behavior, which repeats itself with some periodicity. These characteristic gave us reasons to assume that the time series composed of the intercontact times between a static node and a node which identifies a public means of transportation can be described by a wide-sense stationary processes. In case of such a stochastic process, the autocorrelation function is defined as follows:

$$r_{xx}(\tau) = E[x[n] \cdot x^*[n - \tau]] \qquad (1)$$

where τ is the lag at which the expected value E[•] is computed and x^* is the complex conjugate of x. Since x_n is finite for each value in the range [0, N - 1], the autocorrelation function $r_{xx}()$ also exists and is finite. Once the autocorrelations for all the predefined values of τ have been calculated, the value of τ which corresponds to the highest output of the autocorrelation function will be the periodicity we were looking for.

B. Leveraging the Wiener–Khinchin Theorem to Discover Recurring Patterns in Nodes' Mobility

The problem of the solution described above lies in its *complexity*. In fact, if k is the number of lags we want to include in our search, the complexity of computing the autocorrelation is $O(n^2 \cdot k)$. Even if it is reasonable to assume $k \ll n^2$, the complexity is still quadratic in the length of the input.

To improve the efficiency of the search for periodic patterns in the time-series we present a different approach, based on the Wiener–Khinchin theorem and characterized by a smaller complexity. In the discrete-time case of wide-sense stationary processes for which the autocorrelation function, defined as in (1), exists and is finite, the theorem states that the spectral density $S(f)$ of x_n can be computed from the autocorrelation, as follows:

$$S(f) = \sum_{\tau=-\infty}^{+\infty} r_{xx}[\tau] e^{-i(2\pi f)k} \qquad (2)$$

From (2), it is possible to obtain the autocorrelation function $r_{xx}()$ by computing the inverse Fourier transformation on S(f). Compared to the solution which directly computes the autocorrelation values, the complexity of this second approach mainly depends on the complexity of performing the direct and inverse Fourier Transformations.

C. An Algorithm for the Prediction of the Next Contact Time

Starting from the result of the Wiener–Khinchin theorem, in this section we are going to present two algorithms implemented in our communication middleware: the one for the discovery of periodically recurring patterns in time series and the one for the forecast of the next contact time with a node. It is important to note that the output of the former algorithm is part of the input of the latter. Let X be the vector containing all the intercontact times observed so far for the node η; we can define an algorithm to discover the pattern recurring in the samples in X with the following steps:

1) *Compute the Fast Fourier Transform (FFT) of X:*

$$Y = \mathrm{FFT}(X)$$

2) *Compute the spectral density* $S(Y)$*:*

$$S(Y) = Y \cdot Y^*$$

3) *Obtain the autocorrelation vector* R_{XX} *applying the Inverse Fast Fourier Transform (IFFT):*

$$R_{XX} = \mathrm{IFFT}(S(Y))$$

4) *Find the index p, with p > 0, for which the value of* R_{XX} *is the greatest. p is the output of the algorithm.*

The complexity of the second solution is dominated by the FFT and IFFT functions which, as explained in the previous section, can both be computed with a complexity of $O(n \cdot \log(n))$.

The result of step 3 is a vector containing the values of the autocorrelation function computed over the input vector X for the lags in either the range [0, (N-1)/2] or the range [-(N-1)/2, (N-1)/2], depending on the implementation of the algorithms for computing the FFT and its inverse. In either case, given the symmetry of the autocorrelation function, the information contained in the output vector R_{XX} is the same.

If the number of samples in the vector X is large enough, the value of *p* returned by the algorithm in step 4 is the periodicity of the time series. Note that the described algorithm cannot discover periods greater than (N-1)/2. While this means that the algorithm needs the samples from at least two complete cycles to discover the periodicity in a time series, our experience with the problem suggests that the samples from three complete cycles are enough for it to produce accurate results.

Once the periodicity in the data has been discovered, two more steps are necessary to predict the next contact time with the node η. The first one involves the assessment of the next intercontact time. In order to do this, we used a technique based on the Exponentially Weighted Moving Average (EWMA), as we described in [12]. Considering *p* the periodicity of the input vector X (*p* is the output of step 4 of the algorithm described above), *t* the highest index in the nodes' contact history with respect to the node η (with the first entry having index 0), *ewma_s* the value returned by each invocation to the EWMA function, and α the smoothing parameter, the pseudocode of the algorithm that predicts the value of the next intercontact time can be written as follows:

```
time find_next_intercontact_interval
        (period p, contacts_vector X) {
    i = (t + 1) % p;
    ewma_s = X[i + 1].start - X[i].end;
    i += p;
    for (; i < t; i += p) {
       ewma_s = EWMA (α, X[i + 1].start -
                         X[i].end, ewma_s);
    }
    return ewma_s;
}
```

The first three lines serve to initialize the *ewma_s* variable with a valid value before it is used as a parameter for the call to *EWMA()*. The forecast of the next contact time can be computed by retrieving the end time of the last contact with the node η from the nodes' contact history and by adding it to the evaluation of the next intercontact time, obtained calling the *find_next_intercontact_interval()* function.

V. Experimental Results

We tested our solution using a simulated environment to reproduce the scenario depicted in Fig. 1. More specifically, we used the Network Simulator 3 (*http://www.nsnam.org*), version 3.16, for all the results presented in this paper.

To enable the message dissemination and replication in the simulated environment we took advantage of DisService, an information dissemination middleware purposely designed for extremely dynamic communication environments [13]. DisService supports the overlying applications by enabling the smart management of multiple links and by providing several message forwarding, caching, and replication strategies. These features characterize DisService as a general, effective solution for enabling opportunistic networking in challenging environments.

To support applications' adaptivity, we extended the DisService middleware with additional features to keep track of the nodes' contact history and to collect statistics on the duration of the links with those nodes. Based on this information, applications can build the strategy which best fits both their requirements and the current status of the network, as inferred from the statistics. More importantly, the new features allowed us to further enhance DisService, by implementing the prediction model which can forecast the future contact times with other nodes.

DisService pushes and receives information in the context of messages belonging to a subscription. Multiple instances of an application running on different nodes can share the same subscription and exchange data within it. DisService allows the nodes to receive, store, and carry those messages throughout the network, obeying to any configured policy, in a best effort manner.

For the computation of the Fast Fourier Transform (FFT) and of the Inverse FFT we relied on the high-performance FFTW library (*http://www.fftw.org*), which includes fast routines optimized for several CPU architectures.

A. Scenario

In the simulation scenario there are 5 different NS3 nodes that model three cameras, one bus, and one sink, as depicted in Fig. 1. Every node has a standard 802.11b wireless interface installed, with a maximum available bandwidth set to 11 Mbps. DisService is installed on each node, to handle both the reception and the dispatching of messages. In addition to Wi-Fi, the camera nodes also have a 3G-enabled interface which allows them to connect directly to the sink node. For the purposes of the simulation, we used an NS3 point-to-point radio link with a bandwidth of 1 Mbps to model the 3G connection between the cameras and the sink.

A surveillance application is running on each camera. They generate messages containing highly detailed pictures of the monitored area which need to be delivered to the data center managing the Smart City. DisService takes care of storing the messages in the local cache and of delivering them to one or more sink, which have direct access to the internet and to the data center in the cloud.

The application running on the cameras implements the following policy for managing the cached images. If the cache is full and the camera takes a new picture, the oldest image is replaced. This behavior is consistent with the purposes of the application, because we can assume that a surveillance software is more interested in delivering the most recent information. Nevertheless, to provide more flexibility and more control over the lifetime of the generated messages to the overlying applications, DisService allows the association of different priority levels to messages. This way, only older, lower-priority messages can be replaced with new ones. The application we used for the experiments generates messages with 3 different priority levels: low (normal images), medium (images took at fixed intervals, to provide periodic updates about the status of the street traffic throughout the Smart City), and high (images generated in correspondence of some events detected in the monitored area, or requested directly by the data center). Only messages belonging to the two higher priority levels can be delivered via 3G, if no other path to the data center is available. This restriction is necessary to avoid overloading the cellular network with low priority traffic, which should be reserved for the delivery of urgent data. In our tests, the average size of a picture is 5 MB.

Each camera is more than one kilometer from the others and from the sink, installed in a strategic area of the Smart City, like a large crossroad, or a traffic light which regulates the vehicle flows in streets likely to be subject to congestion. Given the importance of those areas, it is reasonable to think that there might be at least one bus stop in their proximity. For this reason, and to increase the interval during which the camera nodes are under communication range with the buses passing by, in our simulation scenario we assumed the presence of a bus stop close to each camera and to the sink. The distance between each node prevents any direct Wi-Fi communications, so the only way to use Wi-Fi is to exploit the temporarily available connections with a mobile node, like the bus in our scenario, which will function as a ferry and carry the messages from the cameras to the sink.

DisService periodically broadcasts packets (called HELLO messages) to signal the presence of a node to its neighbors. The instances of DisService running on camera nodes will use the information derived from the reception of the HELLO messages to fill in the vector containing all the intercontact times with the bus node (referred to as X in section IV.C).

We modeled the bus movements with a fixed waypoint mobility model. In accordance with the behavior of the bus, the node which identifies it in the network does not follow a constant route, but it changes periodically. As shown in Fig. 1, there are two possible paths that the node can take whenever it reaches the fork near CAM #1. In our experiment, the bus will take the shortest route twice in a row (identified by the dashed arrow in the figure), and then drive the longest path the third time (identified by the normal arrow). These choices will then be repeated until the end of the simulation, identifying a pattern which recurs with a periodicity of 3.

Two bus stops and the sink belong to the shorter path, while the bus encounters all the stops and the sink when traveling the longer path. 32 and 40 segments describe the shorter and the longer path, respectively. The bus travels over these segments each time with a different speed, randomly chosen from a uniform distribution which ranges from 46.8 km/h to 57.6 km/h. The bus also remains at each stop a random amount of time, uniformly distributed between 30 and 40 seconds, before resuming its ride. The choices above introduce a certain degree of variability and allowed us to simulate the effects of small changes in the current traffic conditions and of other elements which affect the bus' behavior, as well as to evaluate the robustness of our solution.

We performed 8 different simulations, to cover all the possible configurations of the prediction algorithm (enabled and disabled) with 4 different values for the cache size of the camera nodes (5, 10, 15, and 20 messages). Each simulation ran for 6 hours of simulated time. During the first two hours no messages were generated: this allowed the DisService instances running on the camera nodes to collect enough information about the mobility pattern of the bus node to feed the prediction model. Also, this made possible a fairer comparison between the solution with predictions and the one without them. The chosen amount of time was adequate to generate enough messages for each class in order to collect significant statistics.

B. Results

During the experiments we collected data representing the status of the simulations to elaborate statistics to describe the evolution of the tests. Fig. 2 shows the *Wi-Fi delivery ratio*, that is the percentage of messages delivered to the sink node via Wi-Fi, i.e., that reached the sink node via the bus, against the cache size, for the cases with predictions enabled and predictions disabled.

Independently from the specific cache size, the performances in terms of Wi-Fi delivery ratio are significantly higher when camera nodes can leverage predictions about the future presence of a ferry node. With predictions disabled, the Wi-Fi delivery ratio goes from about 67% to 81% when the cache goes from 5 to 20 messages, while, enabling predictions, those percentages range from about 86% to almost 94%. Labels in the figure show how many messages were delivered using Wi-Fi against the total. The difference between the two series of data ranges from 217 messages, with a cache size of 5, to 143 messages, with the cache capable of storing up to 20 messages. Considering the average message size of 5MB, enabling predictions redirected about 700MB-1GB of traffic from the cellular network to the opportunistic network, corresponding to about 12-19% of the total traffic in the simulation.

We believe that these results show a very important point. In fact, knowing in advance if a new resource will soon be available, applications can develop smarter policies that better fit their goals. This opens up the possibility to a more uniform usage of the network resources, which will lead to a better QoS and will reduce the load placed on the strategic parts of the network.

The Wi-Fi delivery ratio allowed us to prove the efficiency of our solution as a means to reduce the load on the 3G network, enhancing the use of low-medium range connectivity solutions. Instead, Fig. 3 shows the impact of enabling predictions on the *delivery ratio*. We can notice that the results for medium and high priority messages are the same, while only low priority messages suffer from a reduction in the delivery. Although the results may seem counterintuitive at first, the behavior they delineate is expectable and it is a direct consequence of the cache limit.

When an application generates a message, if valid predictions of future contacts with ferry nodes are available, DisService will store the message in its cache and delay its delivery, waiting for the next ferry to approach. All this is, of course, necessary in order to exploit the Wi-Fi connection with the ferry node and avoid transmissions via 3G. As cache fills up, low priority messages will be discarded, thus affecting negatively the delivery rate. However, an increase in cache size would mitigate this effect. As appears in Fig. 3, the difference between the number of messages delivered with predictions enabled and those delivered with predictions disabled drops significantly with a cache capacity of 15 messages or higher.

These observations and results demonstrate that the exploitation of future connectivity resources requires a higher memory usage. In session 4 we had discussed the costs in terms of computational power to set up a prediction model. Now, the experimental results allow us to point out the different cache management that arises from the exploitation of predictions of future contacts with potential connectivity resources. Because of it, messages tend to occupy the cache for longer times when the prediction feature is enabled and, consequently, a higher number of low priority messages need to be discarded. For our experiments we have considered a scenario where DisService provided each application relying on it with a static cache size. Nonetheless, we believe that a dynamic management of the cache share could help reducing the effect that enabling predictions has on the memory management.

Finally, we would like the reader to note that we tried to design the experiment so that it mirrors the behavior of a public means of transportation, like the bus in the scenario, in a realistic way. To this end, we included random inter-arrival times and variable routes in our simulations. In addition, the efficacy of the theorem, and in turn of our solution, does not depend on the pattern chosen. To verify this, we ran tests changing the lengths of the short and long paths and varying the type of pattern, i.e., changing the number of times the bus would travel the short and long paths before repeating the scheme. The results were all comparable with the ones of the experiments we reported.

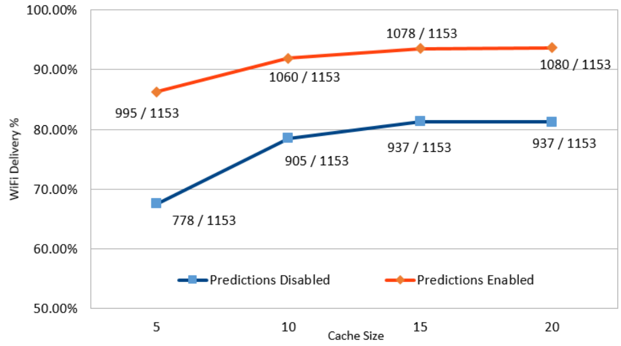

Fig. 2. Ratio of messages delivered via Wi-Fi to the sink node against the cache size; predictions enabled and disabled.

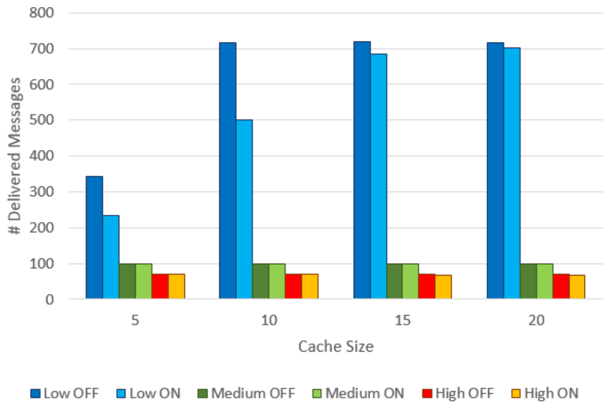

Fig. 3. Number of messages correctly delivered to the sink node against the cache size; predictions enabled and disabled.

VI. RELATED WORKS

A significant number of studies related to Opportunistic Networking and Information dissemination in Smart Cities assume the existence of a fixed infrastructure to support mobile nodes and ease their connectivity [14] [15] [16]. However, infrastructures of that kind are seldom available in modern cities and, even when available, they might not cover the whole urban area. Therefore, solutions that rely on their availability to fulfill the information dissemination cannot be generally applied, as they would not be able to operate efficiently in realities devoid of such supporting infrastructures. In contrast, one target of our work is to develop a general solution, capable of operating properly in most scenarios. As a result, we tried to keep the list of assumptions on the operating environment as short as possible, avoiding to rely on any existing fixed network infrastructure, unless it is widespread in many realities, e.g., free Wi-Fi APs that allow the access to the wired city network.

The scientific research identified many techniques which have been proved successful under other challenging environments with characteristics similar to Smart Cities, such as Tactical Edge Networks, Wireless Sensor Networks or disaster recovery scenarios [10]. These techniques comprise, but are not limited to, message prioritization, restricting information dissemination boundaries [17], optimizing cache size [18], identifying strategic nodes for the success of the dissemination process, and battery life-saving strategies [19]. In order to support distributed applications in Opportunistic Networks, DisService provides a large set of methodologies and tools for the smart communication resources management which implements these techniques.

To the best of our knowledge, these studies did not consider the problem of sharing the limited network and local resources among several applications that simultaneously request to access them. To answer this question, DisService implements a prioritization mechanism: applications can specify a different priority level for each message they generate that, in turn, will have access to different shares of the available resources.

Previous research works already recognized the importance of studying nodes' contact patterns in the context of opportunistic networking [20], especially focusing on social-based forwarding [21] [22]. In [23], we demonstrated the importance of extending those ideas to take advantage of highly mobile nodes to support the dissemination process and we identified many reasons to foster the use of predictions about future contacts. Among them, we listed a better resource usage, more informed and intelligent data fusion techniques, and more energy-efficient communication strategies.

In [24], the authors proposed a solution which leverages the history of nodes' past positions to predict their future locations and to enable predictive QoS routing in Mobile Ad-hoc Networks. The study assumes that mobile nodes have access to the GPS to take accurate measurements of their coordinates and that their clocks are all synchronized. Each node in the network periodically broadcasts data about its movements to all the other nodes, so that they can reconstruct the history of its past positions and predict its future location. The packet routing decisions are then based on the future network topology, computed from the predicted locations of all the nodes. This approach incurs in a large network resource consumption, as nodes need to flood the network to dispatch the data relative to their position. Also, using the GPS to retrieve geographic coordinates increases the energy demand.

The approach to enable predictions about future contacts with other communication resources that we presented in chapter 4 of this paper exploits a pattern detection algorithm which extrapolates periodic patterns from time series. The scientific community recognized the problem of periodic pattern mining in data series to be very challenging and of great importance in many applications [25]. Many studies focused on mining synchronous and asynchronous periodic patterns to design approaches which are resilient to noise, to shifts in the data series, or to the presence of samples which do not belong to any pattern, hence polluting the data. Proposed solutions are effective, but their cost in terms of resources consumption is significant. However, the operating contexts of communication middleware functioning in urban environments are very different from the ones traditionally considered in those studies. DisService is designed to operate in these environments and exploit their peculiarities to provide simple, nonetheless effective approaches which can be implemented on resource constrained devices.

VII. CONCLUSIONS AND FUTURE WORKS

Smart cities will provide many services and applications that can improve the quality of life of citizens. However, the increasing traffic demand requires a communication paradigm that enables applications to operate and interact effectively. The scientific interest concerning the study of Opportunistic Networking in the context of Smart Cities is steadily increasing as it can exploit many characteristic of the nodes, such as social aspects, mobility patterns, and the application context, to realize a smart management of the scarce communication and memory resources.

In this paper, we presented a solution which takes advantage of an efficient mathematical approach to discover periodic patterns in the contacts with highly mobile nodes, such as the ones identified by the public means of transportation equipped with wireless communication interfaces. The results obtained collecting statistics from a simulated environment allowed us to state that predictions can effectively be used to design policies that significantly reduce the traffic conveyed over expensive, bandwidth-limited connection solutions, like the 3G cellular network. This would support the cellular network in withstanding the ever increasing amount of mobile traffic flowing through.

Experimental results also emphasized the cost, in terms of memory consumption, placed by the use of predictions. However, we think that the knowledge about future contacts with other nodes makes it possible also to design smarter resource management policies, thereby reducing their impact on the memory usage. Future works will study the impact of using such policies to implement algorithms for the dynamic allocation of the memory shared between multiple applications on the dissemination process.

The evaluation of our algorithm with traces from real measurements or generated with domain specific simulators would be extremely interesting. This would permit us to test its effectiveness with different scenarios and multiple traffic loads, and assess its scalability. Moreover, the use of traces coming from real measurements would give us the chance to test the robustness of our approach against several forms of noise (such as the presence of varying inter-arrival patterns or of contact outages with the ferry nodes). This is a very relevant matter that we would like to study more in depth.

Another interesting research topic will study if the process of collecting statistics to measure the average quality of the links can produce a more accurate representation of the network and will investigate scenarios where the applications could take advantage of it. Finally, the promising results obtained with the simulated experiments rose the interest of some companies in testing an implementation of our solution; this will give us the chance to test it in the context of a real urban environment.

REFERENCES

[1] Global Health Observatory (GHO), "Urban population growth", http://www.who.int/gho/urban_health/situation_trends/urban_population_growth_text/en/index.html.

[2] Barry McSweeney, "ICT Supporting the Smart Economy: The Case of Ireland", in The Government Report, Technology Actions to Support the Smart Economy, chapter 2.2, pp. 141-151, 2009.

[3] Cisco Systems, "Cisco Intelligent Public Transportation Solution Increases Operational Efficiency and Enhances Commuter Experience", http://www.cisco.com/assets/sol/scc/connectedVision/content/intelligent_public_transportation.pdf.

[4] E. Ferro, B. Caroleo, M. Leo, M. Osella, E. Pautasso, "The Role of ICT in Smart City Governance" International Conference for E-Democracy and Open Government, Krems, Austria, 2013.

[5] G. Iosifidis, L. Gao, J. Huang, L. Tassiulas, "An Iterative Double Auction Mechanism for Mobile Data Offloading", 11th International Symposium on Modeling and Optimization in Mobile, Ad Hoc, and Wireless Networks (WiOPT), May 2013.

[6] I. Augé-Blum, K. Boussetta, H. Rivano, R.Stanica, F. Valois, "Capillary Networks: A Novel Networking Paradigm for Urban Environments", Proceedings of UrbaNe '12, pp. 25-30, 2012.

[7] A. Mtibaa, K. Harras, "CAF: Community aware framework for large scale mobile opportunistic networks", Computer Communications, Vol. 36, No. 2, pp. 180-190, 2013.

[8] D. Karamshuk, C. Boldrini, M. Conti, A. Passarella, "Human mobility models for opportunistic networks", IEEE Communications Magazine, Vol. 49, No. 12, pp. 157-165, December 2011.

[9] C. Boldrini, M. Conti, A. Passarella, "Design and performance evaluation of ContentPlace, a social-aware data dissemination system for opportunistic networks", Computer Networks, 54(4):589-604, 2010.

[10] A. Martìn-Campillo, J. Crowcroft, E. Yoneki, R. Martí, “Evaluating opportunistic networks in disaster scenarios”, Journal of Network and Computer Applications, November 2012.

[11] A. M. Vegni, C. Campolo, A. Molinaro, T. D.C. Little, “Modeling of Intermittent Connectivity in Opportunistic Networks: The Case of Vehicular Ad hoc Networks”, In book: Routing in Opportunistic Networks, Springer-Verlag, 2013, pp. 179-207.

[12] M. Marchini, M. Tortonesi, G. Benincasa, N. Suri, C. Stefanelli, "Predicting Peer Interactions for Opportunistic Information Dissemination Protocols", in Proceedings of the 17th IEEE Symposium on Computers and Communication (ISCC 2012), Cappadocia, Turkey, 1-4 July 2012.

[13] N. Suri et al., “Peer-to-Peer Communications for Tactical Environments: Observations, Requirements, and Experiences”, IEEE Communications Magazine, Vol. 48, No. 10 (Special Issue on Military Communications), pp. 60-69, October 2010.

[14] Y. Xian, C.T. Huang, J. A. Cobb, “Look-ahead routing and Message Scheduling in delay-tolerant networks”, proceeding of: The 35th Annual IEEE Conference on Local Computer Networks, LCN 2010, Denver, Colorado, USA, pp. 40-47, 10-14 October 2010.

[15] V.D. Le, J. Scholten, P.J.M. Havinga, “Unified routing for data dissemination in Smart City networks”, 3rd International Conference on the Internet of Things, 24-26 Oct 2012, Wuxi, China. pp. 175-182.

[16] Y. Zhang, Z. Wang, J. Li, M. Song, Y. Teng, B. Liu, “Opportunistic networks architecture with fixed infrastructure nodes”, Proceedings of the 2012 international Conference on Pervasive Computing and the Networked World, ICPCA/SWS'12, pp. 862-868, Instanbul, Turkey, November 28-30, 2012.

[17] H. A. Nguyen, S. Giordano, “Routing in Opportunistic Networks”, International Journal of Ambient Computing and Intelligence (IJACI), Volume 1, Issue 3, 2009.

[18] N. Ristanovic, G. Theodorakopoulos, J.Y. Le Boudec, “Traps and Pitfalls of Using Contact Traces in Performance Studies of Opportunistic Networks”, IEEE INFOCOM, pp. 1377-1385, 2012.

[19] C. Yu, B. Lee, H. Y. Youn, “Energy efficient routing protocols for mobile ad hoc networks”, Wireless Communications and Mobile Computing, Volume 3, Issue 8, pp. 959-973, 2003.

[20] M. Conti, S. Giordano, M. May, A. Passarella, “From opportunistic networks to opportunistic computing”, IEEE Communications Magazine, Vol. 48, No. 9, pp. 126-139, September 2010.

[21] V. Conan, J. Leguay, T. Friedman, “Characterizing Pairwise Inter-contact Patterns in Delay Tolerant Networks”, Proceeding in Autonomics '07, Article No. 19, 2007.

[22] J. Niu, J. Guo, Q. Cai, N. Sadeh, S. Guo, “Predict and spread: An efficient routing algorithm for opportunistic networking”, in Proceedings of IEEE WCNC 2011, pp. 498-503, 28-31 March 2011.

[23] A. Mazzini, C. Stefanelli, M. Tortonesi, G. Benincasa, N. Suri, “DisService: Network State Monitoring and Prediction for Opportunistic Information Dissemination in Tactical Networks”, in Proceedings of IEEE MILCOM 2010 Military Communications Conference, 31 October - 3 November 2010, San Jose, CA, USA.

[24] S. H. Shah, K. Nahrstedt, “Predictive location-based QoS routing in mobile ad hoc networks”, IEEE International Conference on Communications. ICC 2002, Volume 2, 2002.

[25] J. Assfalg, T. Bernecker, H.P. Kriegel, P. Kröger, M. Renz, “Periodic Pattern Analysis in Time Series Databases”, 14th International Conference, DASFAA 2009, Brisbane, Australia, pp. 354-368, April 21-23, 2009.