From Opportunistic Networks to Opportunistic Computing

Marco Conti, IIT-CNR
Silvia Giordano, SUPSI
Martin May, Technicolor Research
Andrea Passarella, IIT-CNR

ABSTRACT

Personal computing devices, such as smartphones and PDAs, are commonplace, bundle several wireless network interfaces, can support compute intensive tasks, and are equipped with powerful means to produce multimedia content. Thus, they provide the resources for what we envision as a human pervasive network: a network formed by user devices, suitable to convey to users rich multimedia content and services according to their interests and needs. Similar to opportunistic networks, where the communication is built on connectivity opportunities, we envisage a network above these resources that joins together features of traditional pervasive networks and opportunistic networks fostering a new computing paradigm: opportunistic computing. In this article we discuss the evolution from opportunistic networking to opportunistic computing; we survey key recent achievements in opportunistic networking, and describe the main concepts and challenges of opportunistic computing. We finally envision further possible scenarios and functionalities to make opportunistic computing a key player in the next-generation Internet.

INTRODUCTION

The future Internet will be characterized by a pervasive diffusion of devices with heterogeneous capabilities and resources. We envisage a physical world saturated by fixed and portable devices with computing and communication capabilities. Users will carry personal mobile devices (smartphones, PDAs, cameras) bundling several wireless interfaces, supporting computationally intensive tasks, and powerful tools to produce multimedia content. Other types of devices with networking capabilities will be also available in the environment (sensors, fixed cameras, etc.) featuring more specialized resources. Resource-rich users' personal devices, as well as devices spread in the environment, will be the physical components of a pervasive networking environment saturated with distributed resources, which could, in principle, be *pooled together* and *used collectively* to provide end users functionalities much richer than what is available on their individual mobile devices only. This calls for new networking solutions that orchestrate, compose, and manage heterogeneous resources possibly spread over a large number of diverse (mobile) devices, enabling their use as a joint pool. Such enhanced functionality will be exploited to realize, for example, content-centric services (e.g., sharing of user generated content), multimedia services (e.g., composition of audio/video clips with pictures and text gathered on different user devices), personal and environmental services (e.g., healthcare and environmental monitoring), and social-oriented services (e.g., participatory sensing or mobile social networking). This is what we define as a mobile pervasive future Internet.

Opportunistic (self-organizing) networking [1] is the first step in realizing this view. Relying exclusively on wireless infrastructures (e.g., cellular, WLAN, or WiMAX networks) to realize the mobile pervasive future Internet vision does not seem appropriate, as it is very unlikely that wireless infrastructures alone will be able to provide enough bandwidth and coverage to the huge number of devices spread in the environment. In opportunistic networks, such as in a mobile ad hoc network (MANET), the devices spread across the environment form the network. Events such as long disconnections and network partitions are the rule, and no simultaneous multiop paths can be guaranteed. In this type of networks, the mobility of devices is an opportunity for communication rather than a challenge. Mobile nodes communicate with each other even if an end-to-end route connecting them never exists. Nodes have scarce or no knowledge about the network topology; routes are built dynamically, while messages are en route between the sender and the destination(s), and any possible node can opportunistically be used as the next hop, if it is likely to bring the message closer to the final destination(s). Human social structures are at the core of opportunistic networking solutions. Humans carry mobile devices, and human mobility generates communication opportunities when two (or more) devices come into contact.

This work was partially funded by the European Commission under the FP6 HAGGLE (027918) and FP7 SCAMPI (258414) Projects.

0163-6804/10/$25.00 © 2010 IEEE

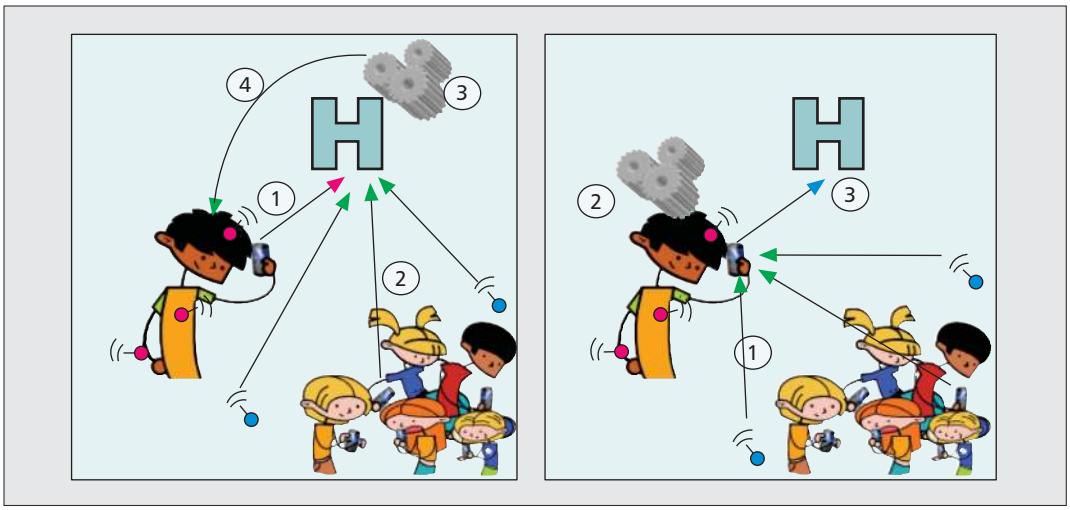

Figure 1. *Opportunistic computing in a healthcare scenario.*

Several application areas would benefit from the opportunistic communication and computing paradigm: pervasive healthcare, intelligent transportation systems, and crisis management are just a few notable cases.

By embedding social relationships between persons in the electronic world made up of their devices, data can circulate in the future Internet following the trust relationships inherited from human social structures.

Exploiting devices' contact opportunities for communication is only a first step. When two devices come into contact, albeit opportunistically, it is also a great opportunity to share and exploit each other's (software and hardware) resources, exchange information, cyber forage, and execute tasks remotely [2]. This opens a new computing era: the *opportunistic computing* (OC) era. In OC each user can avail not only of the resources available on its own device, but can also opportunistically leverage on other resources of the environment, including those on other users' devices, in a trustable and secure way. Users can compose the functionality of the different resources available in the network, enjoying much richer functionality than that available on their own devices. Opportunistic computing thus generalizes the concept of opportunistic networking by considering the opportunistic use of *any* resource available in the network.

The *human factor* is a key dimension of OC, too. By embedding human social structures inside the electronic world of the users' devices we can try to regulate the behavior of users' devices as the users behavior is socially regulated in the real world. An important contribution to characterizing the social context comes from multimodal sensing devices, which enable capturing rich sensing information that can be used to infer user activities [3].

Several application areas would benefit from the opportunistic computing paradigm: pervasive healthcare, intelligent transportation systems, and crisis management are just a few notable cases [2]. For example, in a pervasive healthcare scenario, novel techniques are required for continuous remote (i.e., out of the hospital) and multiparametric monitoring of patients. This can help to significantly reduce the hospitalization of patients with chronic diseases. In this case, typically, a patient is equipped with his/her conventional personal device (e.g., the mobile phone he/she already owns), where vital signals sensed by the personal body area network are collected and sent to the remote medical center. As illustrated in Fig. 1, this data can also be enriched by opportunistically exploiting other resources available in the environment, like environmental sensors, or on the devices of other mobile users (e.g., other patients or care-givers equipped with mobile devices). By exploiting the functionality of the other devices available in the environment, each personal mobile device can thus be able to collect multiparametric data including not only patient biochemical and functional parameters, but also the history of contacts with other people, stimuli from the environment, relevant environmental measures, and so on.

As shown in Fig. 1, in the left frame, without OC, a user who wants to benefit from other resources has to rely on centralized solutions. For example, the healthcare monitoring system might need to elaborate data from environmental sensors around the monitored user. This will provide information about the conditions of the environment that might contribute to explain healthcare data monitored by her/his body sensors. In addition, data coming from other users around her/him might provide a better view of the social environment in which samples have been taken by her/his body sensors. Getting a clear picture about the user's health condition based on this possibly large set of data requires data to be:

- Sampled by triggering the different devices in the environment
- Transported to the central point responsible for analysis (steps 1 and 2)
- Centrally elaborated (step 3)

Finally, results of the elaboration might need to be sent back to the user, possibly to trigger some action on her/his side (step 4).

Opportunistic computing solutions will radically simplify all of these operations, as shown in the right frame. The monitored user's devices, using distributed OC features, will automatically exploit resources of the other devices in the environment and of other users around to take samples and elaborate them locally, without requiring the intervention of any other central

Many concepts behind opportunistic computing come naturally from those studies on opportunistic networking, even if opportunistic computing moves forward from simple communication issues by considering opportunistic exploitation of (pools of) resources.

controller. Both data sampling and elaboration will be mapped to service components available on the devices, which will be composed and executed in a distributed fashion. This will bring several advantages. On one hand, OC will provide a more *effective* solution, as the distributed OC solutions will be able to identify more easily than a central controller the exact set of services on devices around the monitored user required to complement information coming from the user's own monitoring sensors. On the other hand, this will result in a much more *efficient* solution, as the overhead in terms of network traffic will be drastically reduced.

In this article we discuss the evolution from opportunistic networking to opportunistic computing. In the next section we provide a survey on opportunistic networking research and results by focusing on the EU Haggle project. We then highlight the lesson learned from several years of research in opportunistic networking and how this naturally drives OC. We then introduce the OC concepts, and discuss the key challenges and research directions. The final section concludes the article by proposing possible future trends in this research area.

Opportunistic Networking

Opportunistic networking has received much attention in recent years as a disruptive network technology [1]. Many concepts behind OC come naturally from those studies on opportunistic networking, even if OC moves forward from simple communication issues by considering opportunistic exploitation of (pools of) resources.

Several projects worked on studying challenging opportunistic network issues (e.g., mobility models, forwarding, data dissemination), with particular attention to realistic case studies and real experiments.

In the framework of the design of the future Internet, some projects work on opportunistic networking related problems: the delay-tolerant network (DTN)-based activities; such as the European FET-SAC projects — ANA, Bionets, Haggle — the FET-PERADA SOCIALNETS project and the FIRE-N4C project; or, in the United States, some NSF NeTS FIND projects. The majority of those projects mainly aim to solve one specific issue, for example, routing or networking function composition. An exception, from this standpoint, is the groundbreaking Haggle project (http://www.haggleproject.org), funded by the European Commission under the FET SAC initiative. Haggle designed and developed solutions for communication in autonomic/opportunistic networks by studying the properties of the main networking functions, including message forwarding, security, data dissemination, and mobility models. Among projects in opportunistic networking, Haggle is the first effort toward a complete investigation that covers all the main aspects of opportunistic networking, and for this reason it has a fundamental seminal role in OC. In the following we will thus describe the key results achieved by Haggle, in view of their enabling role in the realization of the OC vision. Specifically, we focus on two main areas:

- The study, analysis, and modeling of opportunistic network characteristics
- The design and implementation of innovative architectures and protocols for opportunistic networks

Characterization of Opportunistic Networking

Opportunistic networks are based on the store-carry-and-forward paradigm [1]. Specifically, node mobility is exploited as an opportunity to deliver data among disconnected parts of a network. When a node has data to transfer toward another node (or set of nodes), and no network path exists, connecting the sender and the receiver(s), any possible encountered device (e.g., cell phones and PDAs users carry in their pockets) represents an opportunity to forward the messages toward the receiver. Nodes store messages they have to forward and carry them until encountering another node deemed more suitable to bring the message (closer) to the eventual destination(s). Therefore, it is clear that in opportunistic networks, node mobility plays a crucial role as it permits to bridge disconnected *clouds* of nodes, and ultimately enable end-to-end communications despite connectivity impairments. A key contribution of the Haggle project has been the study, analysis, and modeling of the contact patterns among devices. Pair-wise contacts between users/devices can be characterized by means of two main parameters: contact durations and inter-contact times. The *duration* of a contact is the time during which a tagged couple of mobile nodes are within reach of each other, and thus have the possibility to communicate. An *inter-contact* time is instead the time between two contact opportunities of the same couple of tagged devices. While the contact duration directly influences the capacity of opportunistic networks because it limits the amount of data that can be transferred between nodes, the inter-contact time affects the feasibility of opportunistic networks and the delay associated with them, as they impact on the frequency of opportunities for *moving* messages from one node to another.

To characterize contact durations and inter-contact times occurring in real-world environments, the Haggle project launched an intensive set of measurement studies to collect traces of human contact patterns that represent contact opportunities among their devices. Several traces have been collected using Bluetooth enabled devices carried by volunteers, such as students and researchers in their university and laboratories, participants at some international conferences, or people randomly selected in a public place in a major city. Background software collected contacts between users by automatically establishing a Bluetooth connection when two users met.

Haggle traces were then analyzed to answer two fundamental questions:

1. To provide a characterization of the temporal properties of devices (human) mobility with special attention to the contact time (i.e., the distribution of the contact duration between two nodes), and the inter-contact time (ICT) i.e., the distribution of the

time between two consecutive contacts between nodes
2. To investigate the impact of the above temporal properties on the behavior of routing/forwarding protocols in opportunistic networks

As far as point 1, the Haggle investigations showed that both distributions can be well approximated by heavy-tailed distribution functions approximately following power laws over a significant timeframe [4]. Furthermore, the results derived in the Haggle project indicate that, assuming Pareto distributions for contact and inter-contact times, the performance of simple routing protocols (e.g., flooding) are generally very poor. Such forwarding protocols do not use any information about previous contacts, nodes' identities, or the context in which users are operating. Instead, they follow statically computed rules that limit the number of replicas of each message or the number of hops messages are allowed to travel through. It has been analytically proven that the expected delay of this class of forwarding algorithms is infinite under the heavy-tailed inter-contact times distribution found in the traces. This is a very important result which motivated, inside the Haggle project (and in the research community), extensive research activity on novel and effective routing protocols for opportunistic networks trying to exploit knowledge about users' behavior with a special attention to human social structures. Furthermore, these results also generated a major debate in the scientific community, where many experiments have been conducted trying to confirm or contradict the power law assumption. In particular, special emphasis has been devoted to the distribution of the inter-contact times between devices as fundamental for the behavior of an opportunistic network: the more frequently nodes get in touch, the more opportunities for exchanging messages. Now, there is a general agreement on the fact that the distribution of the aggregated inter-contact times follows a truncated power law (i.e., a power law) with a final exponential cutoff. The analysis of human mobility patterns has then progressed to also analyze other important features. Indeed, human social structures have a key impact on human mobility by affecting the mobility spatial properties, (i.e., where people move), which can be characterized by the distribution of the size of human jumps (or flights), defined as the path between two consecutive waypoints (i.e., locations where a node stops for a while between two consecutive movements). Also in this case, many experiments have been conducted; again, results indicated that a truncated power law can reasonably approximate the jump size distribution.

The understanding of the basic characteristics of human mobility is fundamental not only for designing efficient protocols for opportunistic networks but also to develop accurate models representing how humans behave, to be used for evaluating networking protocols (either through analysis or simulations) under realistic mobility conditions. Developing accurate models able to provide a realistic approximation of human movements is the other research step in the characterization of opportunistic networks. Recently, there have been many studies aimed at providing realistic mobility models, which can be classified into two main categories. In *social-based* models node movements are decided based on social relationships between users. On the opposite end, *location-based* solutions use only the notion of preferred locations to set up the commuting schedule of nodes, while the social dimension is neglected in the location-based models. In the framework of the Haggle project the HCMM model has been developed [5], which joins social and location attractions, and (at the same time) incorporates the three driving forces of human movements that have been identified:

- User movements are conditioned by their social relationships.
- Users tend to visit just a few locations, where they spend the majority of their time.
- Users prefer shorter paths to longer ones [6]; that is, users usually travel over short distances and sometimes move farther away.

HCMM was designed as an evolution of an existing social-based mobility model, called CMM [7], from which it inherits the social graph structure. Differently from CMM, HCMM integrates the social nature of mobility with its spatial dimension. HCMM merges the social and spatial preferences that have been found to characterize real user movements by letting nodes move only to those locations that have a social value for them. These locations are those in which people with whom users share social relationships are more likely to be found. From an operational standpoint, HCMM initially divides nodes into communities, and places each community into one cell of the grid scenario considered. All nodes assigned to the same social community share social relationships between each other. These nodes can share social relations with nodes assigned to a different community as well, based on the so-called rewiring probability. These social relationships determine node movements. In fact, nodes move from one community to another according to the following rule: the more popular a location is among "friend" nodes, the more likely a node will move toward that location. Additionally, according to what has emerged from the analysis of real user traces, when making movement decisions in HCMM, popularity is balanced by the proximity of the location; thus, closer communities are more likely to be visited than farther ones. Performance results have shown that HCMM is able to capture the statistical properties of human mobility patterns, and therefore it has been extensively used to analyze the performance of the Haggle solutions.

HCMM was designed as an evolution of an existing social-based mobility model, called CMM, from which it inherits the social graph structure but, differently from CMM, HCMM integrates the social nature of mobility with its spatial dimension.

Innovative Architecture and Protocols for Opportunistic Networks

The Haggle approach is more oriented to the human way of communicating (and, more generally, the way communities of any type of entities communicate), rather than related to the technological aspect of the communication. This manifests itself in both the definition of innovative architectures and the design of novel net-

Haggle architecture is layer-less: it integrates a general application communication framework with the network protocol. The result is an application driven message forwarding that allows exploiting human factors (context-based forwarding).

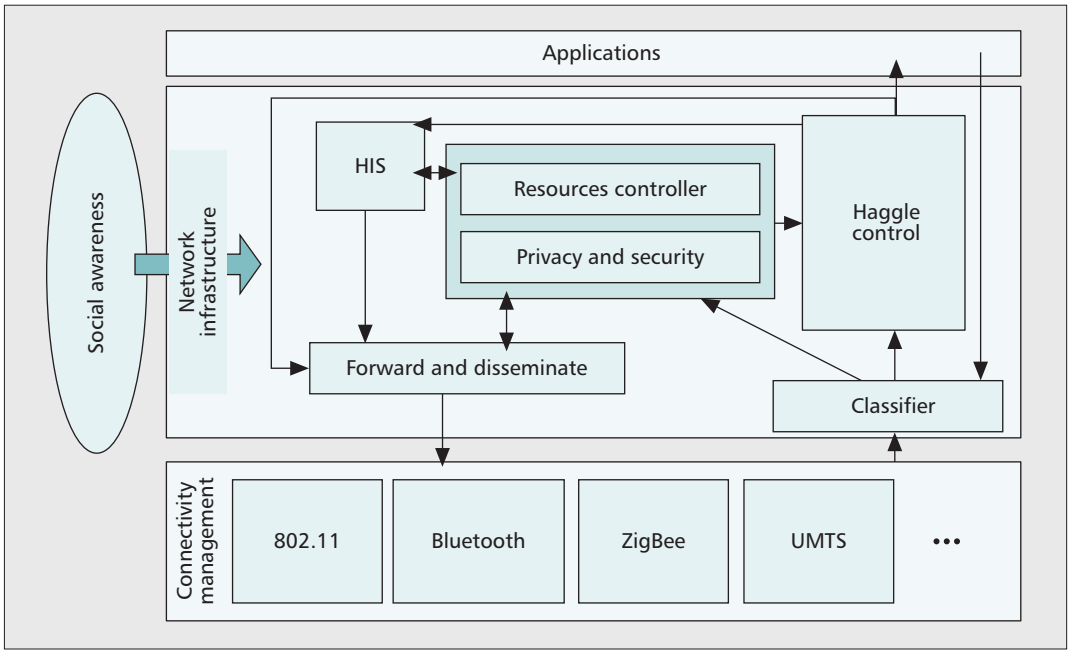

Figure 2. *Haggle conceptual architecture.*

working approaches amenable to opportunistic networking environments.

Architecture — The Haggle architecture is layer-less: it integrates a general application communication framework with the network protocol. The result is an application-driven message forwarding that allows human factors to be exploited through context-based forwarding. The conceptual design of the Haggle node architecture, illustrated in Fig. 2, is based around the notion of a Haggle Information Space (HIS), which contains the data at each user device. Applications can inject new data into the HIS, register interest in incoming data by specifying matching criteria for particular attribute-value pairs, and search the HIS for existing data. Among other data, the HIS collects context information that is used for forwarding.

The forwarding module (forward and disseminate) is designed to cope with a self-organizing, dynamic, volatile, peer-to-peer communication environment: a forwarding decision is performed independently for each data item, exploiting connectivity opportunities and using the context information coming with the data itself (and, in particular, the application concerned), as well as context information contained in the HIS. This very flexible design allowed the definition of very powerful context-aware forwarding algorithms, as explained below. The forwarding module performs neighbor discovery using the available network interfaces, and, when necessary, connects to neighbors that are identified as potential next hops.

The Haggle control module decides whether to accept an incoming data into the HIS, provides an application programming interface for applications, and controls the forwarding module. It is the brain of the Haggle node (by analogy, the HIS is the memory). The control module interacts with all the other modules to decide what should be forwarded and what should be kept in the HIS or discarded. The classifier has the role of classifying the incoming data from the connections below, and distributing them for security check and for the actions of the control module.

Starting from the conceptual architecture in Fig. 2, Haggle has implemented the node architecture shown in Fig. 3. By instantiating this architecture at each node, Haggle nodes are able to implement the networking protocols described in the next section. Several managers compose the node architecture, each for a specific function (e.g., forwarding, security, connection). Managers can make use of modules to implement specific operations related to a given protocol (e.g., the forwarding protocols discussed in the next subsection are modules of the Forwarding manager). The managers interact accordingto an event-based model. Managers generate and subscribe to events, according to a pub/sub paradigm. A central entity (named DataStore) is responsible for managing subscriptions and events, by collecting and dispatching them to subscribed managers. The DataObject is the main element of the Haggle architecture. It is a data structure that contains any type of data: from messages to be sent on the network to internal data exchanged by managers. DataObjects can be associated to events, so that managers can exchange structured data items between them. The DataStore implements a generic pub/sub interface that is oblivious to the event generator(s), making it possible to easily add/remove managers to/from the architecture [8].

Innovative Protocols — In this section we briefly describe the key contributions of Haggle as far as protocol design is concerned. We separately present results in four key areas (i.e., forwarding, data dissemination, security, and applications).

Forwarding Algorithms — Routing in opportunistic networks is surely one of the most compelling challenges, due to the scarce knowledge of the topological evolution of the network. In a socially-aware environment like opportunistic networking, this can be complemented by the context in which the users communicate. Context information, such as users' work addresses and institutions, the probability of meeting with other users, or visiting particular places, can be exploited to identify suitable forwarders based on context information about the destination. In [9] the authors identify three classes for the main routing approaches, based on the amount of knowledge about the context of users they exploit:

- *Context-oblivious*
- *Partially context-aware*
- *Fully context-aware*

Context-oblivious protocols do not exploit any contextual information about the status or behavior of the devices, users, and environment. Partially context-aware protocols do exploit context information, but assume a specific model for this context. When the environment matches these assumptions, they perform very well. But if the environment happens to be different from what they assume, their operation might not be correct. Finally, fully context-aware protocols learn and exploit the context around them. They may not be as efficient as partially context-aware protocols in the conditions for which the latter have been designed, but thanks to learning features, they are much more adaptive. Figure 4 qualitatively shows the advantage of context awareness: partially context-aware protocols, such as Prophet [10], which use the history of encounters to calculate the probability that a node can deliver a message to a particular destination, reduce by about one order of magnitude the network overhead compared to context-oblivious protocols such as Epidemic [11], while the delay is less than three times higher [12]. Similarly, a fully context-aware protocol that exploits social contextual information, further reduces — by about another order of magnitude — the overhead, achieving delay smaller than twice that of Epidemic.

Haggle partially (Bubble Rap [6]) and fully (HiBOp [13] and Propicman [14]) context-aware protocols have been seminal protocols for opportunistic network forwarding:

- They exploit context information to make the forwarding function more efficient: they look for nodes that show increasing match with known context attributes of the destination. High match means high similarity between the node's and destination's contexts and therefore high probability for the node to bring the message in the destination's community (possibly to the destination).
- They also consider that people are not likely to move around randomly. Rather, they move in a predictable fashion based on repeating behavioral patterns at different timescales (day, week, and month). If a node has visited a place several times before, it is likely to visit this location again in the future.

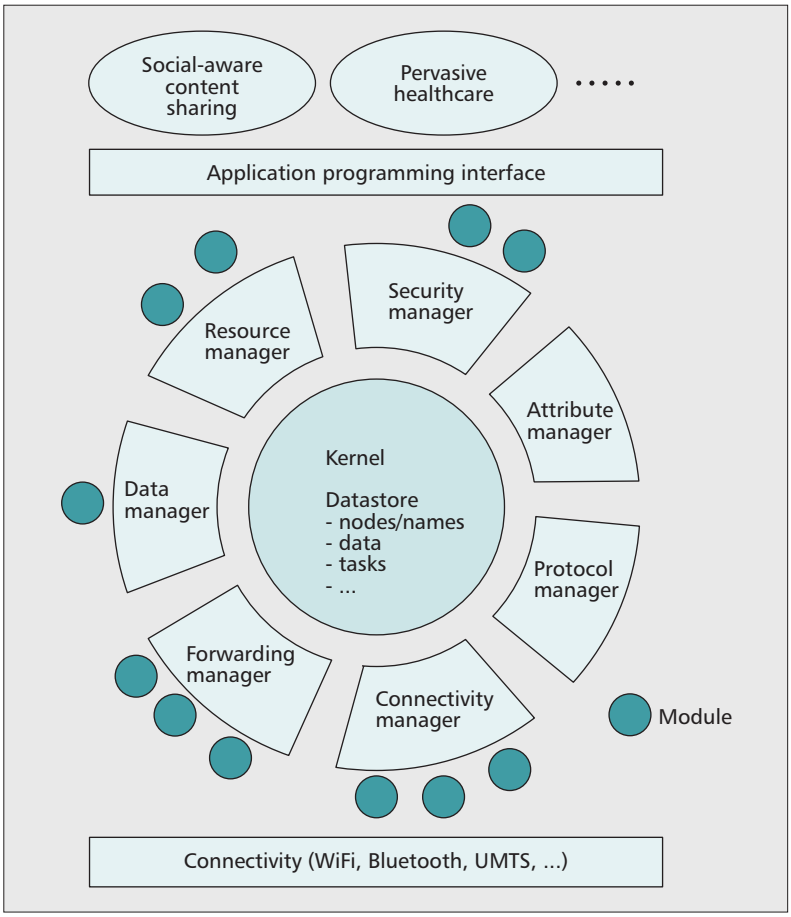

Figure 3. *Haggle node architecture.*

Bubble Rap assumes that relationships among users follow a precise model of social structure (clustered in cliques) and that nodes' social connectivity degrees (within each clique) are highly non-homogenous. In other words, the number of social links each node has toward other nodes in its clique is highly variable, and is distributed according to power laws (which have been observed in real social networks). Nodes in disjoint cliques can communicate thanks to shared members of their cliques (i.e., users being part of different social groups). The main idea behind Bubble Rap is automatically inferring the parameters of the underlying social structure and exploiting the structure properties to select paths. Messages are pushed up in the starting community toward higher-rank nodes (i.e., more sociable users) until a contact with the destination's community is found. Pushing messages up stores messages in the most popular nodes that have more chances to get in touch with the destination's community.

While Bubble Rap works under the assumption of a specific structure of users' social relationships, HiBOp and Propicman infer, as a side effect, social relationships between nodes from context information dynamically gathered at each node.

Propicman stores at each node a profile of the node's user. Profiles are exploited upon contact opportunities to forward messages. The main idea is to look for increasing matches

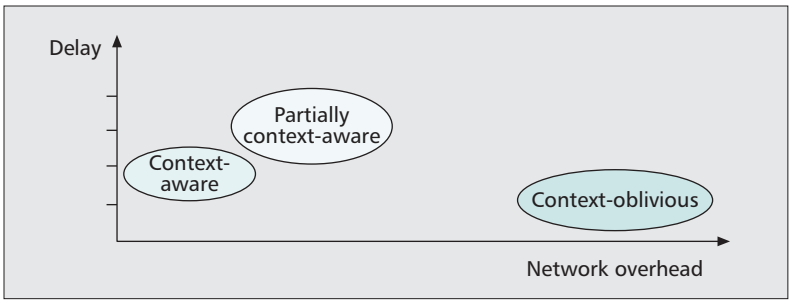

Figure 4. *Delay and network overhead of context-oblivious, partially aware, and fully aware.*

between the profile of the destination user and the profile of encountered users. When a user showing a higher match is encountered, the message(s) addressed to the destination are handed over to that node. A distinctive feature of the Propicman approach with respect to Bubble Rap and HiBOp is exploiting decision trees to select next hops. Propicman is augmented by schemes for limiting the number of messages injected in the network (SpatioTempo) and predicting best forwarding conditions (CIPRO).

HiBOp distinguishes between different contexts (the context of the node, the context of its neighborhood, the historical context). To select next hops, HiBOp looks (as Propicman does) at context information related to the user of the encountered node, but also at historical context information related to that user. In other words, HiBOp is able to understand if a node is a good forwarding candidate also based on the similarity between the destination, and the context (the set of nodes) with which the encountered user is typically in touch. This allows HiBOp not only to forward through users similar to the destination, but also through users often in touch with users similar to the destination. Thanks to this feature, HiBOp infers social relationships between users and users' communities, and implicitly learns the network social structure defined by the users' habits. Finally, HiBOp includes mechanisms to control the messages' replication rate by dynamically selecting the number of copies in the network.

Data Dissemination — Data dissemination is a natural follow-up of research on forwarding algorithms. One of the most interesting use cases for opportunistic networks is the sharing of content available on mobile users' devices. The overall networking environment is seen as full of content pieces available on devices, possibly generated by the users themselves (e.g., photos, clips, tweets), which might be of interest to other users in the network. As the network is intermittently connected, content producers and consumers might not be connected at the same time, and might even be unknown to each other. Therefore, it is not possible, as in a conventional MANET, to establish a path between producers and consumers first, and then disseminate the content. Forwarding and dissemination have to be carried out at once: each piece of content must be opportunistically transferred or replicated on any encountered node, according to policies identifying whether such transfer or replication increases the probability of moving it toward interested users. According to this view, Haggle has investigated data dissemination solutions for opportunistic networks. Specifically, Haggle has designed, implemented and evaluated a utility-based framework for data dissemination called ContentPlace [15].

The ContentPlace design is based on two key assumptions. First, users in an opportunistic network can be logically grouped according to the type of content they are interested in. Second, users movements are driven by their social relationships; that is users move to a certain place because they wish to meet other people with whom they have social relationships, who are likely to be at that place at that particular time. Specifically, ContentPlace assumes users can be grouped in social communities, and their movements are strongly tied with the social communities to which they belong. Based on these assumptions, the key idea of ContentPlace is exploiting information about social relationships between users to drive data dissemination. Each node fetches content available on encountered nodes if this content:

- Is possibly of interest to other nodes the node is likely to meet soon, because of social relationships between the respective users
- Is not already widely available in the social communities to which the user belongs

From a technical standpoint, ContentPlace implements this high-level idea through a utility-based framework. Whenever two nodes meet, they exchange a summary vector describing pieces of content available locally. Then each node computes a utility value for each piece of content either stored locally or available on the encountered peer. Finally, those pieces of content are ranked by decreasing utility, and each node stores in its local buffer the topmost *useful* pieces until the buffer is full. This clearly may require fetching some pieces of content from the encountered peer, and dropping some other pieces previously stored locally. The definition of the utility function is the key component of the framework. Haggle has defined utilities so as to express the importance of content pieces for local users and the communities to which they belong. Furthermore, the utility definition takes into account the cooperation level between different communities, and this allows different dissemination policies (from completely selfish policies to fully social policies) to be realized in the common framework of ContentPlace.

Several social-aware policies have been defined and tested in [15]. Results show that in scenarios in which nodes mix a lot, all policies perform the same (at least in terms of dissemination effectiveness), as the mixing is so high that, sooner or later, all pieces of content become available wherever. In cases where communities are more closed, and just a few nodes (termed *travelers*) bridge them, social-aware policies significantly outperform non-social-aware policies in terms of hit rate, speed of dissemination, and overhead. Among the social-aware policies, the one performing best works as follows. Nodes others than travelers behave greedi-

ly, while travelers set the weights of the communities they will visit according to the probability of visiting them when leaving the current community. The greedy behavior of *residential* users guarantees that content remains available in the community. The altruistic behavior of travelers guarantees that content can be circulated and brought to all communities with interested users.

Security — In opportunistic and pervasive networks, mobile users operate on the move in open, possibly adversary, public environments (e.g., work, public transports, public places). Since users are at the same time service consumers, providers, or relays, it is crucial that they can be held accountable for their actions and can obtain reputation information about other users in advance in order to assess possible future interactions avoiding misuses and attacks.

Privacy is currently one of the main concerns as in opportunistic network protocols the context information exchanged among nodes might possibly include sensitive information about the users. Other topics related to the security field in opportunistic networking include cooperation enforcement, encryption, and robustness against denial of service (DoS) attacks to routine operations. A few results related to these topics are available in the literature (see Haggle deliverable D4.3 for an overview [8]). For example, in the framework of the Haggle project, a cooperation enforcement scheme tailored to opportunistic networks based on reward mechanisms has been proposed. This scheme copes with all the security problems of typical reward-based mechanisms, including protection against poisoning attacks, cheating actions, and unfairness. The work in [16] analyzes the effect of a wide range of security attacks on routing operations in opportunistic networks built on dropping packets, flooding, falsifying routing tables, and counterfeiting message acknowledgments. The main finding is that *multipath* opportunistic routing is very robust *by design* (i.e., even without any form of authentication). The main reason is because all of these attacks result in nodes' unavailability or path disruptions, which are already considered characteristic features of the opportunistic network by multipath routing protocols.

To the best of our knowledge, no full-fledged solutions exist to ensure privacy, confidentiality, and integrity in opportunistic networks. However, privacy can be enforced in opportunistic networks by exploiting a key feature of opportunistic networking environments, the concept of *community*. Indeed, users typically move together with friends or within a social group; hence implicit trust relations exist with the nodes nearby, which can be exploited to provide accountability, trust, and privacy. Novel security designs are therefore needed in order to use trust relations to secure communications in opportunistic networking scenarios. *Key management* in opportunistic networks constitutes the basic security block for this. In general, in an opportunistic network users can be members of different communities, and the same (physical) opportunistic network can support multiple communities at the same time. If an authority-based key management system is in place,[1] a simple scheme can provide privacy support to context-aware opportunistic protocols by guaranteeing that context information about any node is exposed to members of the node's own community only. Note that allowing even unknown users of a known community to know selected information is usually not perceived as a privacy threat as it happens (e.g., in social networking sites). The trusted authority of the key management scheme is also in charge of registering users to communities and enforcing public policies to allow users to be part of the community. It also defines keys (either symmetric or asymmetric) reserved for communications between members of the same community. Upon registration, users can decide the set of context information they wish to expose to other community members and receive the keys for communicating in the community. While research on key management for MANETs has received a lot of attention, just a few papers elaborate on how to tailor this body of work to opportunistic and delay-tolerant networks. Some authority-based solutions have been proposed by exploiting identity-based cryptography (IBC) because IBC provides features particularly suitable for disconnected environments. Specifically, solutions based on IBC just require a trusted private key generator (PKG) for the distribution of private keys, while the public key of a node can be computed by knowing the node's identifier only. Ordinary nodes have only to contact PKGs once to obtain their own certified *private* key. The work in [9] shows that, thanks to these features, IBC solutions:

- Reduce the number and frequency of interactions with authorities with respect to standard solutions based on trusted key servers
- Allow communications between ordinary nodes and the trusted authority (to retrieve keys) to be asynchronous with respect to communications between ordinary nodes (to exchange data)

Therefore, IBC solutions are particularly suitable for disconnected networking environments. Identity-based cryptography has been applied in [16] to develop the Haggle security primitives required to preserve privacy within trusted communities that use context-based forwarding protocols. Specifically, the authors of [16] extend the work in [14], which is prone to dictionary attacks, by developing a solution based on hash functions with salting for thwarting dictionary attacks. The new solution enables forwarding while preserving user privacy by allowing secure partial matches in the header and enforcing payload confidentiality.

Applications — The opportunistic networking paradigm is suitable for supporting several types of pervasive applications: mobile social networking, sharing of user generated contents, pervasive sensing, and pervasive healthcare are just some notable cases. Inside the Haggle project different types of applications have been developed to test the effectiveness of the opportunistic network paradigm (see deliverable D6.4 [8]). *Photo-share* is a good example of a service that allows the sharing of user generated multimedia content: the application allows users to share

Since users are at the same time service consumers, providers or relays, it is crucial that users can be held accountable for their actions and can obtain reputation information about other users in advance in order to assess possible future interactions. This in order to avoid misuses and attacks.

[1] *Authority-based solutions rely on trusted authorities in charge of distributing and managing keys. These authorities can be online or offline, and can be distributed in the network so as to avoid the problem of a having a single point of failure. Ordinary nodes have to receive keys from the authorities before joining the network.*

Mobile on-the-move social networks have a special role in testing the effectiveness of Haggle solutions, as the human factor is a key/critical element in the opportunistic paradigm. MobiClique is the most important example of this class of applications that has been considered within Haggle.

with other users pictures taken with mobile phone cameras without using any networking infrastructure. Haggle *Electronic Triage Tag* is an application designed to support a medical team in a disaster scenario. Specifically, in this scenario the Haggle protocol stack is used to forward to a coordination point the medical information related to the victims by exploiting the devices of the medical team as data relays.

Mobile on-the-move social networks have a special role in testing the effectiveness of Haggle solutions, as the human factor is a key/critical element in the opportunistic paradigm. MobiClique [2] is the most important example of this class of applications that has been considered within Haggle. MobiClique allows people to maintain and extend their online social networks through opportunistic encounters in real life. MobiClique builds (on top of a locally maintained social network structure) a framework for collaborative forwarding relying uniquely on opportunistic contacts. A fully functional prototype is implemented on top of the Haggle reference implementation together with a simple Facebook desktop application for initial user profile setup. The initially proposed applications for MobiClique include social networking, unicast and multicast messaging. Recently, MobiClique has been extended with an *Epidemic Voting* application. The voting application contains a list of topics for voting. A vote on a topic is a user's opinion expressed as "liked" or "did not like" for simplicity. A user is allowed to vote once for each topic during a limited voting period. The testing of the voting application was carried out during SIGCOMM 2009 in Barcelona using a testbed of 100 smartphones running MobiClique distributed to conference participants. In the test, vote topics are presentations and social events at the conference, and the voting period for each topic started 10 minutes after the beginning of each voted events and ends two hours after. Votes are disseminated epidemically to all MobiClique devices. Users can display the currently available snapshot of the collected results for each topic only after having given their vote. The MobiClique prototype uses Bluetooth technology for both data exchanges and the identification of human-to-human contacts, as it limits the communication range to 10–20 m.

The SIGCOMM 2009 experiment provided two types of information: the effectiveness of data dissemination using the opportunistic paradigm and the structure of the social network defined by the participants of a conference. In the case of the data dissemination statistics, the experiment pointed out the limitations of the current technology in implementing opportunistic networking protocols. Indeed, the average number of users reached by each piece of information disseminated by MobiClique is about 66 percent in simulation experiments, while in the real experiment the success rate was about 26 percent. Several factors limit the application performance in a real environment: the low performance of a Bluetooth system, varying users activity (i.e., devices turned off), and users' mobility.

The second contribution of the experiments was related to comparing the structure of the online social network with the on-the-move social network (i.e., the network of contacts among the conference participants during the conference). The structure of the on-line social network is obtained by exploiting the Facebook profiles of each participant, including the basic personal details (name, city, country), the list of friends, and the list of Facebook networks and groups. The friendship graph has a single giant connected component of 65 persons. The most connected node has 19 friends while 8 participants have no preexisting connections at all with the other participants on Facebook. The average degree is 4.36. On the other hand, the structure of the on-the-move social network was estimated from the devices' contact patterns during the experiment. Specifically, the graph was created by establishing a link between any two nodes that during the experiment have more than one contact. This contact graph is very dense, and on average each participant meets 50 percent of the other users involved in the experiment, and the average distance between two nodes is only 1.42, while the graph diameter is 3. This is somehow expected because human mobility also creates contact opportunities among users that are not part of an online social community.

Lessons Learned and Open Challenges

As summarized in the previous section, research on opportunistic networks, and the work in the Haggle project, started in the first place by characterizing the fundamental properties of opportunistic networks, and evolved by addressing the key problem of routing and forwarding in intermittently connected networks, data dissemination, security, privacy, and applications design. From these activities, it has been possible to learn several lessons about research on opportunistic networks, and identify areas that still deserve further investigation, as well as new research areas and challenges.

The first aspect to be highlighted is the fact that *human behavior must be considered the premier contextual information* to be exploited in the design of opportunistic networking solutions. As the topology of opportunistic networks is very unstable and dynamic, only using topological information to build network protocols — the typical approach in the Internet and in MANETs — is not enough in this environment. Topological information has to be complemented with contextual information, which could help nodes to estimate information such as contact opportunities, and frequency of meetings. As opportunistic networks are typically formed out of mobile users devices, the behavior of the users in terms, say of movement patterns, and interest in contacts is *the* key information to be gathered and analyzed in order to build useful representations of the environment where the protocols will operate.

Among the possible information describing the users' behavior, *information about their social relationships and interactions* are particularly useful. Knowledge about social relationships allows

protocols to learn the social network of users, which can be exploited in several ways. On one hand, the social network indicates the different roles of the users in the network. For example, it highlights the social importance of people in terms of number of social links with other users. This can be exploited, for example, in the dissemination process to place content on *social hubs* (users with the highest number of contacts) as a way to speed up the dissemination process. Furthermore, the social network of users is a strong predictor of future meetings among them, as people typically move to meet other persons with which they have social relationships. Predicting contact opportunities between nodes is clearly a fundamental piece of information in opportunistic networks. Finally, the user social network can be exploited to understand the level of trust between users, and therefore the level of trust and reliability of communications; assuming a higher level of social trust between two users can be mapped in greater reciprocal willingness to carry each other messages, and thus greater reliability of communications involving their nodes. Specifically, results in the field of social anthropology show that social links of an *ego* are organized according to shells of increasing size and decreasing average tightness (Fig. 5). Relationships in inner shells are more frequent and tighter from a social perspective, and thus are expected to be characterized by a higher level of trust. In more detail, experimental evidence from real measurements presented in [17] (and confirmed by several other studies) highlights at least four well-identified shells. The innermost shell corresponds to the set of persons having the tightest social link with the ego, and amounts, on average, to about five individuals. The next shells include individuals with less and less tight links. An interesting finding is that the size of the shells increases approximately according to a factor of 3, up to a maximum of 150 individuals. The number 150, also known as the *Dunbar number*, represents a sort of social cognitive capacity limit for humans, and is related to the size of the portion of the neuro-cortex used to *store* information about social relationships. The theory about the *capacity of the social brain* has also been confirmed by other studies by the same authors, looking at social structures of primates. While the numbers change, the *relation* between the size of the neuro-cortex and the maximum number of individuals in the ego social network is an invariant.

The above remarks clearly hint at another emerging aspect, the *strong link between opportunistic networking and mobile social networks*. This aspect has multiple facets. On one hand, mobile social network *applications* will benefit from an efficient opportunistic networking environment, which will permit to unleash their full potential. As shown by the MobiClique example, mobile social networking applications have much greater potentiality than simply running Facebook (or other social networking sites) on mobile devices. In opportunistic environments such applications may build dynamic social networks among users that share, possibly for some limited periods of time, a common interest or a goal, and provide social networking services to them. On the other hand, the properties of the mobile social network of the users can be exploited by all components of opportunistic networking platforms to build and optimize networking services such as forwarding and data dissemination.

The final lesson we wish to highlight is the fact that *opportunistic resource use*, central in OC, seems to be a very promising concept for enabling future pervasive networks, as discussed in the next section. In opportunistic networks nodes contribute and avail of each other's resources in terms of connectivity and storage space, with the goal of sending information or content between non-connected endpoints. This idea proves very effective in an environment characterized by high dynamism, unstable topologies, and intermittent connectivity. It enables connectivity opportunities to be optimized, and self-organizing networks to be built out of mobile devices that are more and more pervasively available (e.g., smartphones and PDAs). It is therefore sensible to extend this concept, and *consider a general opportunistic resource usage paradigm*, in which not only connectivity and storage, but any resource available on mobile devices can be contributed and opportunistically used when needed. This is the baseline idea of the OC paradigm, which we discuss in detail in the next section. Addressing this novel paradigm is definitely one of the most intriguing open challenges in the field of opportunistic networking.

Opportunistic Computing: The Services Approach

Opportunistic computing envisions an environment with a multitude of devices carried by users (e.g., smartphones and PDAs) or spread in the environment (e.g., sensors, fixed cameras) with each device having a number of resources (several wireless interfaces, a lot of memory, powerful CPUs, components able to generate multimedia content). The resulting environment therefore contains a multitude of heterogeneous resources that can potentially be used by mobile users to implement the required function. The key point is how those resources can be accessed and combined in an optimal and secure way. Specifically, the goal of an OC platform is to enable each user not only to use the resources available on its own device, but also to opportunistically exploit a composition of the other resources in the environment, including those on other users' devices, in a trustable and secure way. Different from pervasive computing and conventional service-oriented computing approaches in mobile environments, but, as in opportunistic networking, OC assumes that the network is extremely dynamic with unstable topologies; therefore, it is necessary to opportunistically use the contacts with other devices to carry on networking and computing tasks. Thus, OC *adds* to pervasive computing the opportunistic feature and consequently the *human factor* (e.g., information about the social relationships among users). In this way users will be able — even in *challenged networking conditions* — to enjoy services much richer, in terms of function-

Opportunistic computing envisions an environment with a multitude of devices carried by users or spread in the environment, with each device having a number of resources (several wireless interfaces, a lot of memory, powerful CPUs, components able to generate multimedia content).

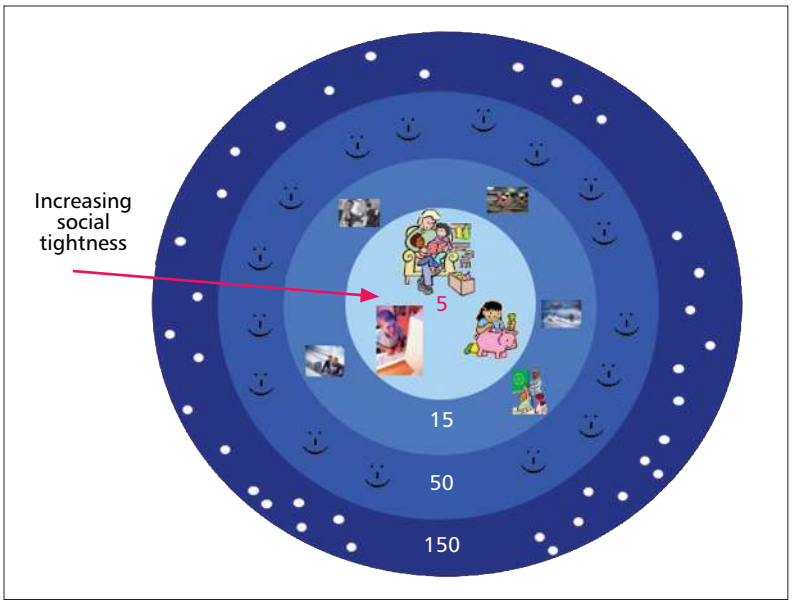

Figure 5. *The circles of acquaintanceship organized as hierarchically inclusive levels.*

ality, than what can be available on their own individual devices. We can also say that OC extends the concept of opportunistic networking to the concept of opportunistic resource usage, thus allowing users not only to simply communicate, but also to avail of complex services despite long disconnections and severe partitions of the network.

Conceptual Architecture for Opportunistic Services

To implement the above vision, we need a platform enabling each individual user to opportunistically exploit the resources available in the environment. A promising approach to achieve this is based on using the *service* concept as the key abstraction to build such a platform. Each resource available in the environment (from hardware resources like sensors, cameras, and CPU, to soft resources like databases, pieces of code implementing particular functions, pieces of contents, etc.) can be seen as a service that local and remote applications can invoke. In this new paradigm the users' devices become service prosumers (i.e., service providers and consumers at the same time).

The conceptual architecture of an OC service platform is shown in Fig. 6. At the lower layer of this architecture are the local resources, those available on a device, abstracted as a set of basic services the applications can use directly or that can be compbined to form complex and richer services. The network itself is a service that can be used to access other services available on remote nodes. Specifically, in an OC environment, the network (each network interface) is a special resource, which is managed through opportunistic and pervasive networking technologies. By exploiting the network service, a device can build a local abstraction (proxy) of each remote service. In this way an application running on a device not only can use and compose the services available locally, but the service platform makes it possible to use and compose both local and remote services. The simplest case is when a service can be directly implemented by exploiting *locally available* resources. In this case the service platform running at a particular node can work in isolation from the other nodes of the network. However, in the general case, not all required modules/resources are available locally, and hence the service platform will *opportunistically look* in the network for the missing services. The platform will discover the required modules/resources, identify the optimal way of invoking them, and take care of gathering and combining the results and presenting them to the application in a suitable format.

It is worth noting that although there is some conceptual similarity between OC and the areas of service-oriented architectures (SOA) and cloud computing, the opportunistic networking environment is different from what is assumed by SOA and cloud computing; hence, completely novel technical solutions need to be designed.

Research Challenges

The grand challenge to realize the above vision can be summarized as follows:

The design and evaluation of models and algorithms for secure shared service provision in an opportunistic networking environment.

This needs significant rethinking of the solutions developed for mobile and pervasive computing systems, as these systems generally assume that remote resources can be accessed through stable network connections, and network disconnections are exceptions. Providing services in opportunistic networks entails addressing several challenging problems related to identifying, accessing, and using remote services. The knowledge of when and where services are available inside the opportunistic networks (service discovery) is a key element of the service platform. Due to the dynamic nature of the environment, it is not sufficient to discover which node is offering a service; a key challenge is to predicting the service availability taking into consideration when and for how long a node can meet the device providing a given service and its resources availability (e.g., the amount of energy or the traffic load on that node). Remote services might be available with different stability levels, depending on how long and how frequently the local and remote nodes will be in touch (either directly or through an opportunistic multihop path). The stability of remote resources will be a key parameter that the service infrastructure shall consider to deliver services to applications. Resource constraints of individual nodes (including energy), the context in which such nodes operate (with special attention to the social context), and services' distribution are the other key ingredients for selecting the remote node(s) from which to request a service. These elements will be used to answer a set of basic questions when two nodes come in contact to decide whether to request a remote service: What information can they exchange? What resources are they willing to share? What incentives (e.g., eco-

nomic, community-based) and security guarantees (e.g., privacy, trust) would they have to do so? What is the cost of using a remote resource? How many replicas of the same service need to be requested to achieve a given quality of service level?

To answer these questions, contextual information can be used to provide more precise estimates of when and where resources will be available. Among contextual information, a key role is reserved for social awareness: the awareness about the social structures of users relationships, the probability of users communicating and getting physically in touch with other users' and so on. With this knowledge we can predict which users/devices a node will meet in the future and hence which remote resources will be available in the future and when; social structures also constitute a key ingredient to guarantee a secure access to remote services. For example, trust relationships embedded in social structures can be exploited to build novel trust relationships in services access. Indeed, in OC environments new security policies need to be designed to control the access to remote resources.

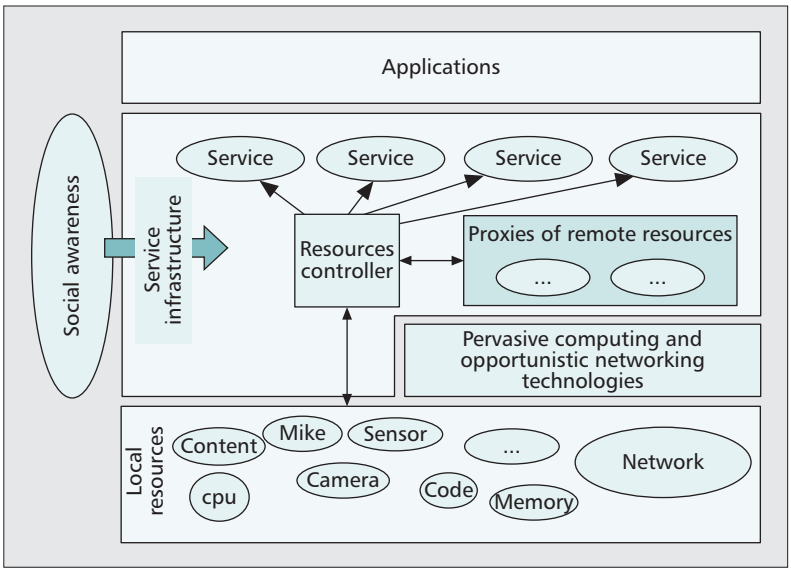

Figure 6. *Opportunistic computing conceptual architecture.*

Research Activities on Opportunistic Computing

Opportunistic computing is an emerging paradigm that builds on the results of several research areas. Opportunistic networking is the first step in realizing this paradigm, but OC moves forward from simple communication issues to develop a framework to enable collaborative computing tasks in networking environments where long disconnections and network partitions are the rule. Mobile and pervasive computing has tried to achieve this objective too, but in the presence of a significant degree of connectivity among the computing devices. The autonomic computing and communication concept has some similarities with OC, as in both cases it is required to be self-adapting to the conditions of the surrounding environments. However, the opportunistic environment targeted within OC generates new challenges not present in autonomic computing. Opportunistic computing also builds on social networking structures as enabling mechanisms for the service platform. Specifically, OC requires models of social interactions between users and ways to exploit them in networking protocols for opportunistic networks — see, for example, the research activities carried out in the FET-PERADA SOCIALNETS project (http://www.social-nets.eu projects). The human factor (i.e., information about the social relationships among users) is extremely relevant to OC. A special role in achieving awareness of the social context is played by *multimodal sensing*. Almost any last-generation smartphone includes cameras, accelerometers, microphones, and so on. Recent research projects — see, for example, the Metro-sense project at Dartmouth [3, 18] and the NeTS-FIND project *Network Innovations for Personal, Social, and Urban Sensing Applications* — show that readings from multimodal sensing devices can be used to infer precise information about the social behavior of the users and the social environment around them. Multimodal sensing will thus make it possible to infer the dynamically changing social environment of users and exploit this information to optimize OC solutions.

Opportunistic networking, autonomic communication and computing, social networking, and multimodal sensing are enablers for OC; however, realizing the OC concept requires tackling several new research challenges for developing a set of middleware services that cope with disconnections and heterogeneities, and provide the applications with uniform access to data and services in a disconnected environment. To the best of our knowledge, currently only two projects are directly addressing these challenges: the National Science Foundation (NSF), *Distributed Opportunistic Computing* (DOC) project, and the EU FIRE *Service Platform for Social Aware Mobile and Pervasive Computing* (SCAMPI) project. The two projects share the goal of exploiting all available resources in an opportunistic environment to provide a platform for the execution of distributed computing tasks. To achieve this result, some milestones have been identified. The first one is related to defining the policy for remote service invocation when the application running on a device (seeker) generates the request for the execution of a service that is not locally available. In this case the seeker has to decide from which device running the requested service (provider) it has to request the service execution. Due to the challenging environment (the contact time between a seeker and a provider is often not long enough for service completion, and the next inter-contact time may be arbitrarily long), several invocations of the same service can be spawned by the seeker from different providers to minimize service completion time.

The choice of which providers to contact and the total number of parallel executions of the same service define the service invocation policy

The choice of which providers to contact and the total number of parallel executions of the same service define the service invocation policy used by the seeker. The goal is to identify the optimal number of service invocations to minimize the expected service time.

used by the seeker. The goal is to identify the optimal number of service invocations (i.e., number of replicas of the same service) to minimize the expected service time (i.e., the time interval between when a request is generated at the seeker and when the seeker receives the output results). This is a challenging problem that has been investigated in [19]. In this work, the authors have investigated efficient and effective schemes for service replication and have identified an optimal policy for the invocation of service components in an OC environment. This is the first step to provide a platform for the execution of distributed computing tasks. Starting from this result, several other milestones still need to be achieved. First of all, there is the need to identify effective policies for opportunistic services composition, i.e., how to compose the services available in the environment to implement a richer and complex service by taking into account where (on which devices) each single service component is available, and the contact and inter-contact time statistics between the seeker and the other devices hosting the service components. A promising approach to tackling this challenge is based on extending to the opportunistic environment the graph theoretic techniques for service composition in pervasive environments [20]. Before addressing services composition, service components available in the network must be identified. To this end, mechanisms for service discovery and notification are basic functions of the service platform. Clearly, using service components available on remote devices generates several privacy and security concerns. Mechanisms for establishing reputation among devices and disseminating reputation information in the opportunistic environment represent an interesting direction to address security issues. Reputation mechanisms have already been investigated in the mobile ad hoc networking field, but long disconnections and sparse networks set new challenges. In addition, as we have discussed for opportunistic networks, exploiting the human social structures — trying to regulate the behavior of users' devices as the users behavior is socially regulated in the real world — constitutes a very promising direction to tackle privacy and security issues in opportunistic environments.

The above mechanisms and methods are necessary to implement the service platform, but do not provide any indication about the type of services the platform is able to support, and the quality of experience provided to users. Answering this question requires the development of analytical and simulation models to investigate the quality of service that the distributed platform can provide to the applications.

Conclusions

After the eras of ubiquitous computing and pervasive computing, a new era, that of opportunistic computing, is coming. Opportunistic computing takes advantage of opportunistic networking solutions, as communication is built by exploiting connection opportunities. Opportunistic computing extends the concept of opportunistic networks by enabling the opportunistic use of *any* resource available in the environment. Specifically, OC leverages the human pervasive network, and allows users to opportunistically access a large number of services and resources available on other users' devices. This is made possible by exploiting knowledge of the human social relationships between users, to predict which resources will be available, when they will be available, and at which stability level. This is a very powerful vision, as it allows combining and making use of the enormous amount of hardware and software resources in the network in a distributed ad hoc fashion. Furthermore, it extends the mainstream area of social networking, bringing and fully unleashing it in mobile pervasive networks.

In this article we have discussed the main results in opportunistic networking, and how they pave the way to OC. Then we have presented the concept of OC, and highlighted the key related research challenges.

The new era of opportunistic computing has just started. Several challenges must be faced to answer key questions: What is the limit for exporting resources to other users? Where do security and privacy enter into the game? How can we design and model an environment that is so complex and multifaceted? These challenges are extremely exciting from an intellectual standpoint, and, if addressed, will enable a wealth of innovative application scenarios. Therefore, we can foresee that OC will be a major research direction in the coming years.

References

[1] L. Pelusi, A. Passarella, and M. Conti, "Opportunistic Networking: Data Forwarding in Disconnected Mobile Ad Hoc Networks," *IEEE Commun. Mag.*, vol. 44, no. 11, 2006.

[2] M. Conti and M. Kumar, "Opportunities in Opportunistic Computing," *IEEE Computer*, vol. 43, no. 1, Jan. 2010, pp. 42–50

[3] A. Campbell *et al.*, "The Rise of People-Centric Sensing," *IEEE Internet Comp.*, July 2008, pp. 12–21.

[4] A. Chaintreau *et al.*, "Impact of Human Mobility on Opportunistic Forwarding Algorithms," *IEEE Trans. Mobile Comp.*, vol. 6, no. 6, June 2007, pp. 606–20.

[5] C. Boldrini and A. Passarella, "HCMM: Modeling Spatial and Temporal Properties of Human Mobility Driven by Users' Social Relationships," *Comp. Commun.*, vol. 33, 2010, pp. 1056–74.

[6] P. Hui, J. Crowcroft, and E. Yoneki, "BUBBLE Rap: Social-Based Forwarding in Delay Tolerant Networks," *Proc. ACM MobiHoc*, May 2008.

[7] M. Musolesi and C. Mascolo, "Designing Mobility Models Based on Social Network Theory," *ACM Mobile Comp. Commun. Rev.*, vol. 11, no. 3, 2007, pp. 59–70.

[8] Haggle Project, http://haggleproject.org/index.php/Deliverables

[9] M. Conti *et al.*, "Routing Issues in Opportunistic Networks," in *Middleware for Network Eccentric and Mobile Applications*, B. Grabinato, H. Miranda, and L. Rodrigues, Eds., Springer, 2009, pp. 121–47.

[10] A. Lindgren, A. Doria, and O. Schelen, "Probabilistic Routing in Intermittently Connected Networks," *ACM Mobile Comp. Commun. Rev.*, vol. 7, no. 3, 2003, pp. 19–20.

[11] A. Vahdat and D. Becker, "Epidemic Routing for Partially Connected Ad Hoc Networks," tech. rep. CS-2000-06, Duke Univ., Comp. Sci. Dept., 2000.

[12] H. A. Nguyen, and S. Giordano, "Context Information Prediction for Social-Based Routing in Opportunistic Networks," June 2010, SUPSI-TR062010.

[13] C. Boldrini, M. Conti, and A. Passarella, "Exploiting Users' Social Relations to Forward Data in Opportunistic Networks: The HiBOp Solution," *Pervasive Mobile Comp.*, vol. 4, no. 5, Oct. 2008, pp. 633–657.

[14] H. A. Nguyen, S. Giordano, and A. Puiatti, "Probabilistic Routing Protocol for Intermittently Connected Mobile Ad Hoc Networks (PROPICMAN)," *Proc. IEEE WoWMoM/AOC*, June 2007.
[15] C. Boldrini, M. Conti, and A. Passarella, "Design and Performance Evaluation of ContentPlace, a Social-Aware Data Dissemination System for Opportunistic Networks," *Comp. Net.*, vol. 54, no. 4, Mar. 2010, pp. 589–604.
[16] A. Shikfa, M. Önen, and R. Molva, "Privacy and Confidentiality in Context-Based and Epidemic Forwarding," to appear, *Comp. Commun. Net*.
[17] W.-X. Zhou *et al.*, "Discrete Hierarchical Organization of Social Group Sizes," *Proc. Biological Sci.*, vol. 272, no. 1561, pp. 439–44.
[18] N. D. Lane *et al.*, "Mobile Phone Sensing: A Disruptive Technology for the App Phone Age," *IEEE Commun. Mag.*, Sept. 2010.
[19] A. Passarella *et al.*, "Minimum-Delay Service Provisioning in Opportunistic Networks, to appear, *IEEE Trans. Parallel and Distrib. Sys.*
[20] S. Kalasapur, M. Kumar, and B. Shirazi, "Seamless Service Composition in Pervasive Environments," *IEEE Trans. Parallel Distrib. Sys.*, vol. 18, no. 7, July 2007, pp. 907–18.

Biographies

Marco Conti (marco.conti@iit.cnr.it) is a research director of the Italian National Research Council, and he is the head of the Ubiquitous Internet group at IIT-CNR. He published more than 250 research papers and three books related to computer networks. He is Editor-in-Chief of *Computer Communications* and Associate Editor-in-Chief of *Pervasive and Mobile Computing*. He is chair of the IFIP working group WG 6.3. He has served as general/program chair for several conferences, including IEEE PerCom, IEEE WoWMoM, IEEE MASS, ACM MobiHoc, and IFIP TC6 Networking. He is on the editorial boards of *IEEE Transactions on Mobile Computing*, *Ad Hoc Networks*, and *Journal of Communication Systems*.

Silvia Giordano [SM] (silvia.giordano@supsi.ch) is a professor at SUPSI, and head of the Networking Laboratory. She is co-editor of the book *Mobile Ad Hoc Networking* (IEEE-Wiley 2004). She has published extensively on journals/magazines/conferences in the areas of QoS, traffic control, wireless, and mobile ad hoc networks. She is an editor of several important journals/magazines and on the steering/organizing/technical committee of many major conferences/workshops. She is a board member of ACM-N2Women, and a senior member of ACM and IFIP-WG6.8.

Martin May (Martin.May@technicolor.com) received his Ph.D. in 1999 for his work on Internet QoS mechanisms at INRIA, Sophia Antipolis, France. Afterward he was with Sprintlabs, and then he founded a startup in France. After selling the company at the end of 2003, he joined ETH Zurich as a lecturer. In 2008, he joined the Thomson Research Laboratory in Paris, where he is now director of strategy for the Media Delivery Research Program. He is an area editor of *ACM Computer Communication Review* and *Computer Communications*.

Andrea Passarella (andrea.passarella@iit.cnr.it) has a Ph.D. in computer engineering and is with IIT-CNR, Italy. He was a Researcher at the Computer Laboratory, Cambridge, United Kingdom. He works on opportunistic networks, with an emphasis on content-centric architectures, services, routing protocols, and mobility models. He is Program Co-Chair for IEEE WoWMoM 2011, and on the PCs, among others, of IEEE MASS and PerCom. He was Co-Chair of IEEE AOC 2009, TPC Co-Chair of ACM MobiOpp 2007, Vice-Chair for ACM REALMAN (2005–2006), and IEEE MDC (2006). He was Workshops Co-Chair for IEEE PerCom and WoWMom 2010. He is on the Editorial Board of *Pervasive and Mobile Computing* and the *International Journal of Autonomous and Adaptive Communications Systems*.

The new era of opportunistic computing has just started. Several challenges must be faced to answer key questions. These challenges are extremely exciting from an intellectual standpoint, and, if addressed, will enable a wealth of innovative application scenarios.