Towards a feasible social-based methodology to manage wireless connectivity context data

Roberto Rigolin Ferreira Lopes
University of São Paulo
São Carlos, Brazil
rigolin@icmc.usp.br

Bert-Jan van Beijnum
University of Twente
Enschede, The Netherlands
beijnum@ewi.utwente.nl

Edson dos Santos Moreira
University of São Paulo
São Carlos, Brazil
edson@icmc.usp.br

***Abstract*—Wireless connectivity context data is composed by date, time, geographical localization, and QoS metrics, to cite the most common. These data are employed, in a particular way, by fundamental techniques for context-aware connectivity management, e.g. mobility predictors, handoff mechanisms and mobility management. For instance, mobility and QoS predictors use, as input, previous georeferenced network context data. Normally, context data are available in hardly updated databases with considerable size. In this paper, we propose a social-based methodology to allow mobile users collaborate to discover wireless connectivity islands. The methodology is composed by methods to gather, combine, summarize and share context data inside the users' social circles. We, also, designed a schema to mashup context data with location-based social media. It is result of a prototyping effort and we focus the discussion on its feasibility and limitations in terms of storage size, power consumption and QoS metrics.**

***Index Terms*—Context-aware connectivity management, network context data, virtual social community.**

I. Introduction

Wireless connectivity engineering is improving bandwidth, coverage and ubiquitous access. However, it is, also, getting expensive, in terms of both capabilities and cost [23], i.e. it is computational complex and has a high monetary price. The computational complexity occurs because the mobile device is in charge of essential tasks, such as: gather, store, synchronize and process connectivity context data. These are basic functionalities of several context-aware wireless connectivity management solutions reported in the literature, e.g. [3], [10], [17], [19], [25], [26].

These investigations employ the current and historical *connectivity context data* to keep the mobile device connected while moving. The information applied is also referred as mobility information, mobility profile or connectivity context data. It is the input for handover mechanisms [7], mobility predictors [21], QoS predictors [18], [24], and mobility management [4] standards and protocols. One of the most common sources of this type of data are public searchable databases, also called *wardriving* [15] databases. Even though, it commonly has a considerable size, become outdated quickly and make available just elementary information, e.g. the ones in WiFi beacon packages [11].

Few efforts have been directed to improve the relevance of this fundamental source of information. In order to achieve this, the context data must be local, mobile, fresh, personalized and social. With this in mind, we have developed a community-centric methodology to handle wireless connectivity context. The focus is on data used to manage IP connectivity in heterogeneous wireless environments [13].

The methodology orchestrates the fundamental methods to manage connectivity in a feedback loop. It can be assisted by a specialized virtual community or by popular on-line social networks, e.g. Twitter. The idea is to allow mobile users to collaborate, with others, in their social circle [12]. As a result, the loop converges to better connectivity experiences for all of them. Their time-spaced set of IP connectivity experience is represented as a digraph called *connectivity path*. These experiences are sent to the context manager, and an algorithm is applied to combine it in a *connectivity graph*. It is the media socialized, i.e. shared by mobile users in their social circles.

This investigation takes into consideration the hypothesis that people have their opportunities of communication and collaboration enhanced inside smart environments, e.g. digital houses, universities and hospitals. Social networks would be an important tool to achieve it. Providing efficient services to enable communication through digital media. Presently, popular virtual social networks provide programming interfaces to embed third part applications, and plugins to allow interaction between common web sites and the virtual community.

We explore these possibilities in two ways: building a specialized virtual community and creating mashups [27] with location-based social media. The first one is a web application which implements the proposed methodology. The social circle is brought to the application using a plugin from Google Friends Connect. The context data is shared through RSS (Really Simple Syndication) feeds [14] channels. The second one, attaches the shared *connectivity graph* to popular location-based social media. This kind of media, usually, mashups the mobile user's localization, in a map, with a message/link. Then, people in their social circle can search for feedbacks related to that specific place. Employing this adds relevance and scalability to the shared context data.

Certain valuable contributions of this paper are the discussion related to the methodology, development, and limitations of our solution. Some implementation issues motivated an analysis of the critical points to make the methodology feasible. The feasibility is examined in terms of storage size, and power consumption. In addition, we deployed our prototype in a university building to verify IP connectivity improvements in terms of QoS metrics, e.g. throughput, and signal quality. The experimental results shows that wireless connectivity experiences are improved, even in a environment with good network coverage.

The rest of this paper is organized as follows. In Section II, the main related works are examined. The section III describes our social-based methodology and the context data structure. Section IV present the main issues faced during the development of the prototype. In section V and VI, the viability of the proposed approach is discussed using quantitative metrics. Finally, conclusions are given in Section VIII.

II. Related work

Recent investigations, concerning wireless connectivity management, combine mobility prediction and network QoS mechanisms to access ubiquitous services in smart environments [22]. Moreover, handoff mechanisms and mobility management have been considered a critical issues in several investigations reported in the literature, e.g. [15], [18], [21], [22]. Nicholson et al. [17] concluded that an ideal connectivity management would be possible if, before the access point in use becomes unreachable, the mobile device must knew which one will use next. Also, had previously completed association, and had already received an IP address from the new network.

In our research, we reuse and adapt these ideas and mechanisms. Placing the *historical context database* as fundamental source of information for the other three components, as shown in Fig. 1 (1). The **mobility predictor** forecast the probable geographical localization of a mobile user, based on past context data. When the user reaches a location where the current network becomes unusable or does not fulfill the applications' QoS requirements, it is necessary to activate a **handoff mechanism**. Also, several handoff mechanisms published are context-aware, e.g. [1], [9], [20]. Finally, in order to keep the transport connections alive it is necessary to make use of a **mobility management** mechanism, e.g. Mobile IP at layer 3. Theoretically these components fulfill the requirements of an ideal connectivity management aforementioned.

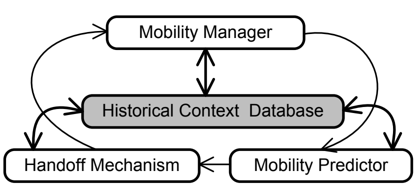


Fig. 1. Fundamental components of the connectivity management.

Complementary research address investigations employing the knowledge of mobility patterns in order to properly allocate network resources [16], [19], [25]. Aiming at enhance the QoS experienced by applications in mobile devices. However, it takes time to construct an efficient mobility model. Because it is necessary a set of past context data as input. Our solution complements these approaches, offering combined context data to build the initial model. As function of time, the model can be adapted regarding the user's mobility patterns.

BreadCrumbs [16] explores the derivative of connectivity of a mobile user to perform context-aware handoffs. Prasad et al. [19] have proposed a framework for modelling and predicting user movements in wireless networks. Both apply a Markov model to predict mobility in wireless networks. The observed networking conditions are stored to construct a personalized mobility model, on the user's mobile device. Both solutions could use our methodology to feed the predictors.

Recent investigations are exploring multiple network interfaces to increase throughput, taking advantage of the diversity of access providers. Extra wireless interfaces can support parallelism in network flows, improve handoff times, and provide communication with nearby peers [5], [17]. Fig. 2 illustrates two typical cases: (1) one network interface and multiple connections; and (2) multiple interfaces with single connections. A hybrid approach, of these two, is also possible, i.e. multiples interfaces with multiples connections.

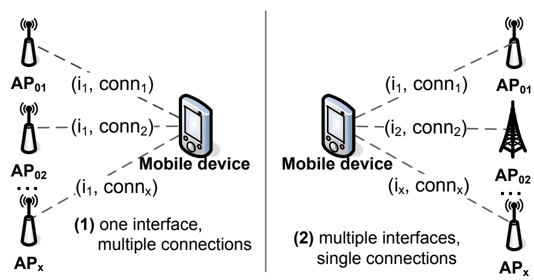


Fig. 2. Two types of connectivity: (1) one interface and multiple connections, (2) multiple interfaces with single connections.

Nicholson et al. [17] explore mobile devices with multiple network interfaces that execute the MAC (Media Access Control) layer in software. Juggler is a link-layer implementation of an 802.11 virtual networking service. The ultimate goal is enhance data throughput. Specially, when wireless bandwidth is superior to that of an access point's wired back-end connection. It is done multiplexing data across many networks, using virtual network interfaces [17], the Fig 2 (2) illustrates this statement. We believe that historical context data would support the decision making to choose the second or the x^{th} access point.

The aforementioned researches are investigating handoffs optimization using mobility prediction, and relying on public or private QoS databases. However, few efforts are direct to provide better quality data in terms of facility to access, handle and update. We believe that these predictors can have better forecasts using our social-based solution, if compared to just use common *wardriving* databases. Also, they would use our solution to delivery connectivity information to others mobile users.

III. Design

A. Methodology

Four essential entities compose the methodology: mobile device, mobile user, social circle and the context manager, as shown in Fig. 3. At the mobile device, *client side*, the mechanisms responsible to manage the IP connectivity are implemented, e.g. just the operational system. People carrying these devices experience distinct quality of service in a shared place, e.g. a building, neighborhood or campus. Finally, the mobile users are able to collaborate, with other ones, in their social circle. The context manager, *server side*, assist them with services. For instance, the users can upload their connectivity experiences, share them, and download the combination of all members' feedbacks [13], to cite two essential services.

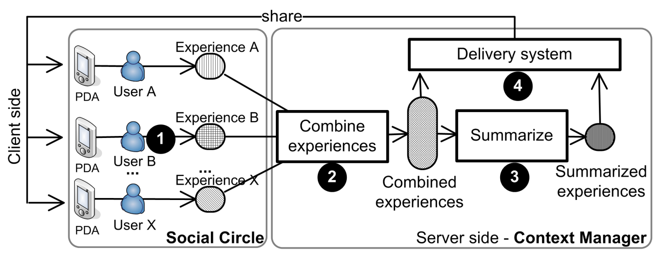


Fig. 3. Methodology: (1) Gather user's network context data; (2) Combine them; (3) Summarize the numerical QoS related data; (4) Share the combination.

Fig. 3 highlights the four fundamental methods disposed in a feedback loop. To start, it is necessary to acquire, store and upload the user's connectivity experiences at a particular location. It is our first method, as shown in Fig. 3 (1). It can be done using the operational system's programming interfaces and supported network protocols. The second method, Fig. 3 (2), is in charge of combining all users' experiences in an adequate format. To accomplish this, we employ a graph-based model that combines all these data in a *connectivity graph* [13]. This model is discussed in subsection III-B.

At a specific place, e.g. around a enterprise facility, several mobile users may have had a connectivity experience. In this case, a search for feedbacks could return many context instances. One way to reduce the size complexity is delivery the quantitative data condensed by statistical tools, as illustrated in Fig. 3 (3). Lastly, this information needs to be accessible to all community members through a delivery system, as shown in Fig. 3 (4). A channel is created in a RSS feeds to distribute the combined and summarized context data, more details are given in subsection IV-B.

B. Network context data and data model

Connectivity experience is a set of time spaced network context data. Including QoS parameters related to an access point used, by a person with a mobile device, at a certain location in a specific time. This data is gathered from different sources, e.g. user profile, operational system, alive connections and access point, as shown in Fig. 4. Sun et al., in [22], defined mobility as logical concept rather than a physical one. In which, mobility means the change of the logical location of network's access points instead of user's geographic position. With this vision in mind, we defined and structured our set of context data, as illustrated in Fig. 4.

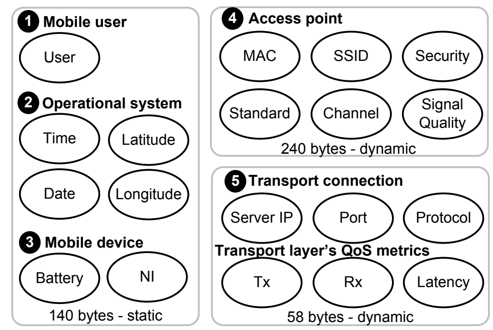


Fig. 4. Example of a set of context data and their sources: (1) mobile user, (2) operational system, (3) mobile device, (4) access point and (5) transport connection.

For Bettini et al. [2] an adequate context information model would reduce the complexity of context-aware applications and improve their maintainability and evolvability. Taking this recommendation, we defined the sources of context data: (1) mobile user, (2) mobile device, (3) operational system, (4) access point and (5) transport connections, as shown in Fig. 4. The data fetched from these sources were specified regarding our development environment, and the requirements of a runnable prototype, more details are discussed in Section V.

Handling this set of data appropriately we can infer some useful information. For instance: who is the person; where and when she or he was connected; which access point was used; which transport connections were opened. At each scan, the context data's size can vary as function of the number of networks available and transport connections opened. Fig. 4 indicates the static and dynamic subsets of context data, in terms of size. At a particular scan, the data related to date, time and mobile user will appear just once in the set, it is part of the static subset. On other hand, the dynamic subset is composed by data associated to the access points and transport connections. The instances, of this subset, will vary as function of the number of access points available, and transport connections opened.

Regarding the data model, in a previous work [13], we defined a graph-based structure to represent connectivity experiences, called *connectivity path* or $G_{path}(V_{path}, E_{path})$. The vertices, V_{path}, are locations where the mobile device switched from an access point to another, called *handoff points*. The edges, E_{path}, represents the access point used and the ones available between handover points, with labelling "$currentAP, \{allAPs\}$". Thus, a G_{path} condenses the wireless connectivity experience of a specific user.

$$G_{conn}(V, E) = Combine(G_{path}[user.length()]). \quad (1)$$

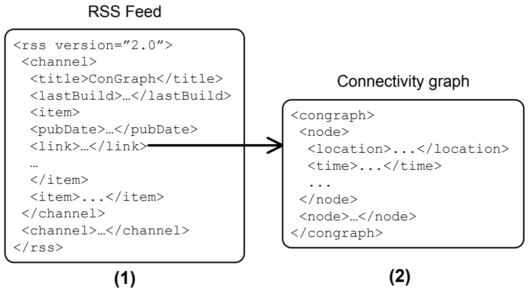


Fig. 5. (1) part of the RSS file. (2) elements of the XML file, which is a textual representation of the connectivity graph.

Given the connectivity paths of all mobile users in someone's social circle, an algorithm is applied to combine these, to a new graph called *connectivity graph* or G_{conn}, as shown in Eq. 1. The vertices, V, are still handover points created by the union of all V_{path}. However, the edges, E, are the users whom used the same access point between two nodes. It is a digraph with labels like "$currentAP, \{allUsers\}$". An textual representation of G_{conn}, i.e. a XML file, is the media shared in the users' social circles, more details in Section IV.

IV. Development

A. Delivery system

Historical network context data needs to be flexible in terms of handiness, freshness and updatability. In order to comply with these requirements, we developed a delivery system using RSS feeds. It provides a standardized XML-based format to publish very often updated applications, e.g. news headlines and blog posts. The most interesting functionalities are: combine digital content automatically, allow users to subscribe to timely updates in channels, and aggregate feeds from distinct sources in one place.

The fundamental idea is create channels to publish connectivity context inside virtual social networks. Fig. 5 (1) shows the basic structure of a RSS feeds file. A channel is defined by the tag $< channel >$ plus metadata related to the date, authorship and content. The content can be set into a timestamped item, tag $< item >$. In our prototype the connectivity graph is referenced in the tag $< link >$, which points to a textual representation of the graph, as shown in Fig. 5 (2).

Each social circle has a channel subscribed by the members. The tag $< lastBuildDate >$ is used to verify if there are updates in the channel. This tag is modified when a new item is added to the channel, which means that a connectivity graph was created or updated. In this way, the users can get updated just reading the newest item available. Moreover, the graph's evolution is recorded and is available.

B. Specialized virtual community

The context manager, earlier reported in subsection III-A, was implemented as a web application with the functionalities described in the methodology. The social circle is a plugin, from Google Friends Connect[1], that brings all the users' friends to the web application, as as illustrated in Fig. 6. In this figure a web interface is shown with user's personal data (1), feeds subscribed to (2), and community members (3). In addition, an abstract schema is highlighted and shows how we can share feeds among distinct virtual communities (5), using a RSS feeds aggregator (4).

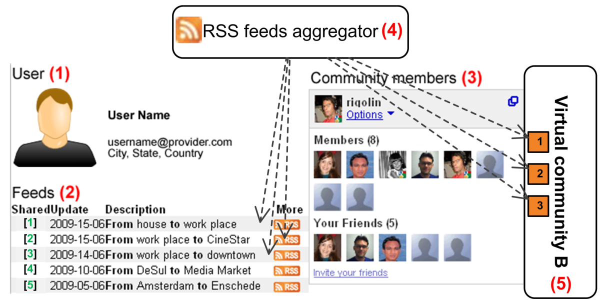


Fig. 6. Web user interface and an abstract view of the distributed application: (1) personal data, (2) subscribed feeds, (3) community members, (4) RSS feeds aggregator, and (5) an virtual community B.

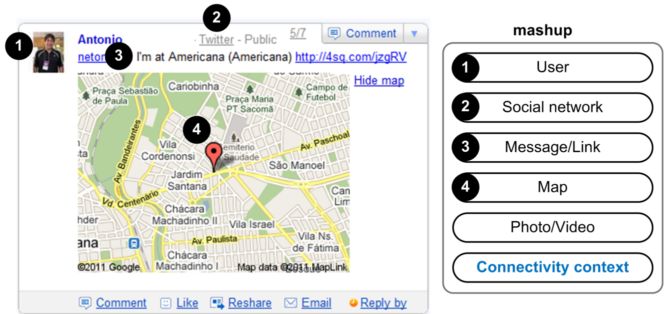
The feeds are available at the aggregator in the web. Then, the applications can create common links to the feeds' URLs. The basic services to handle the context data are embedded in the community, a web application. The whole solution is composed of a web application, the community, and a mobile application at the users' mobile device. These components interact through the Internet using standard HTTP requests. The community is responsible to provide a graphical user interface and a set of basics services described earlier in the methodology, Section III.

C. Mashup with location-based social media

From the idea to create a specialized virtual community, the prototype evolved to a less centralized solution. We observed that people are using location-based services to share feedbacks about specific places inside their social networks. For example, Foursquare[2] is a web and mobile application that allows mobile users to connect with friends and update their location. The application helps people to explore their city, and discover new places. It applies a social-recommendation engine, which provides real-time suggestions based on the user's social graph. Fig. 7 shows a typical mashup shared in social networks. The presentation is composed by the user (1), a social network (2), a message plus links (3), and a map (4).

We argue that network context data can become more relevant when attached to these popular mashups. Fig. 7 (b) shows an abstract view of the main entities in a mashup, and add the connectivity context data. In order to develop it, we identified two basic ways to interact with location-based services, as shown in Fig. 8. First, the user is having an experience somewhere and decides to share it inside his/her social circle. The input is composed by geographical localization, a

[1]http://www.google.com/friendconnect/
[2]https://foursquare.com



(a) real implementation (b) abstract view

Fig. 7. Typical mashup on web: (a) real implementation and (b) an abstract view of the main entities.

message/link and the connectivity context data. In this case, the output would be similar to the one showed in Fig. 7.

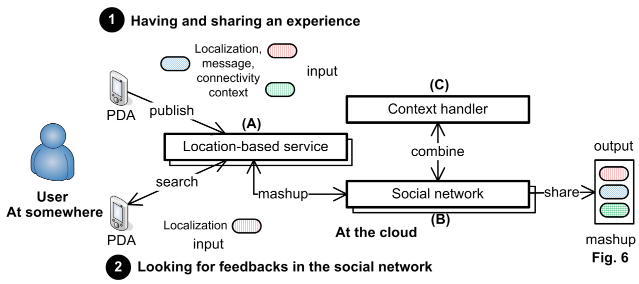


Fig. 8. Two forms of interaction: (1) having and sharing an experience somewhere; and (2) looking for feedbacks in the user's social network.

Another way of interaction, Fig. 8 (2), occurs when the user is somewhere and is interested in find feedbacks from his/her friends in that specific place. In this case, the input is the current localization and the service returns the set of feedbacks in the virtual social network. With this approach the mobile users searches for meaningful feedbacks and, also, gets connectivity experiences there.

The **location-based service** , Fig. 8 (A), can be any application that publishes georeferenced data in social networks, e.g. FourSquare and Google Latitude[3]. The **social network**, Fig. 8 (B), can be any virtual social network that allows embedding third party applications. The **context manager**, Fig. 8 (C), implements the methods described earlier in the methodology, subsection III-A. This module needs to access the social network to get the connectivity data to combine with other ones from the same social circle. The combination is, also, shared using the delivery system described earlier in subsection IV-A.

V. Feasibility

A. Storage size

Gathering network context data arbitrarily is expensive in terms of storage size. Because network scans are performed, frequently, to sense the available access points in the current environment. Each wireless technology does this in a particular way. The scan frequency is strong related to the probability of take a connectivity opportunity, as listed in Table I. This table lists the scan intervals, in seconds, for each probability of finding a new access point. In addition, shows the estimated storage overhead of one mobile device performing scans in continuous 24 hours, for each probability. For example, a hypothetical mobile device scanning in intervals of *250 s*, i.e. with 80 % chance of take an opportunity, will accumulate *194 Kbytes* of contex data at the end of the day.

Opportunity (%)	Scan interval (s) [8]	Size (Kbytes)
20	1500	32.40
40	1000	48.60
60	500	97.20
80	250	194.40
>80	<15	3,240.40

TABLE I
DATABASE SIZE GROWING IN FUNCTION OF THE PROBABILITY OF TAKE A CONNECTIVITY OPPORTUNITY DURING A WALKING (VELOCITY LOWER THAN 8 KM/H).

Our prototype gathers the set of context data, described earlier in Fig. 4. It has a static subset and two dynamics ones, in terms of storage size. The storage size can be estimated by Eq. 2. This equation computes the database size as function of time, numbers of users and scan frequency. For instance, the estimated size in Table I was calculated for one user $u = 1$, performing scans during 24 hours, $t_1 = 86,400s$, in an area with about 2.4 access points per scan [15], $x_1 = 2.4$.

$$\mathbf{D}atabase = \sum_{i=1}^{u} \frac{t_i}{o} * \left(c + \sum_{i=1}^{n} x_i * s_i \right). \quad (2)$$

Where:

u	number of users.
t_i	user's total usage time.
o	scan interval.
c	size of the static part of the context data.
n	total of dynamic sets of data.
x_i	number of the dynamic's subset instances.
s_i	size of this particular data subset.

The database size has a linear grow until 80%, and a disproportional grow for more than 80%, cited as $> 80\%$. In this case, the database size increases about 16 times faster than with 80% of probability. Means that sense more than 80% of the connectivity opportunities, i.e. perform scans in itervals lower than 15 s, is expensive in terms of storage overhead. It is, also, visible that the grow in storage is not proportional to the decrement of the scan frequency, as shown in Fig. 9.

B. Power consumption

Power is another critical resource used to gather connectivity context data. In order to estimate the power consumption we assume that the scanning task for a IEEE 802.11 network interface costs $5ms$ in reception mode at $300mW$ [6]. Using the scan frequencies, earlier listed in Tab. I, it is possible to calculate the power consumption (joule) as function of the chance of find a connection, results are plotted in Fig. 10.

[3]http://www.google.com/latitude/intro.html

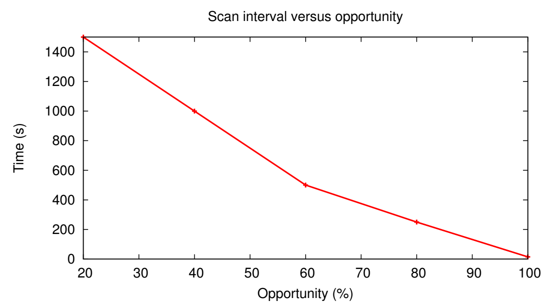


Fig. 9. Scan frequency in function of the probability of make use of a connectivity opportunity.

The behavior is similar to the observed with storage size, i.e. there is a significant grow in power and storage overhead for scan frequencies lower than 15 seconds.

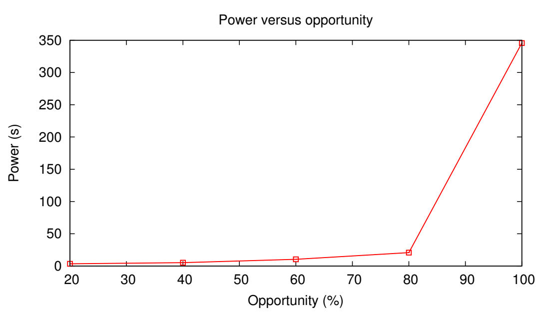


Fig. 10. Power consumption in function of the probability of make use of a connectivity opportunity.

Comparing the two charts in Fig. 10 and 9 it is possible to identify a strong relation between the scan frequency and the power consumption. It is a inverse relation, however, it is not linear. This information is combined with the knowledge of high density handoff areas, from the connectivity graph, to guide a smarter way to perform scans, i.e. do scans more often only in areas with high probability of execute a handoff [13].

C. Saving resources

The connectivity graph, G_{conn}, is used to identify areas with high density of handoffs. It is done by traversing the graph and calculate the number of the edges and users arriving/leaving the vertices, i.e. handoff points. The main idea is to save storage space and energy defining policies to perform scans. For instance, Fig. 11 shows a connectivity graph with 5 nodes, p_1 to p_5, retrieved in our testbed described later in Section VI. There are 3 mobile users walking through the five points in sequence. This path is in an university building and about 210 meters long. Each mobile device used different polices to perform scans, namely:

- **Constant $>$80%**: a constant loop to take more than 80% of the connectivity opportunities, i.e. the scan interval is 15 seconds.
- **Preemptive $>$80%**: performs scans every 15 s when the mobile device is near a handoff point in the connectivity graph. The main idea is just look for connectivity opportunities when the mobile device is at a high density handoff area.
- **Preemptive 80%**: performs scans in intervals of 250 seconds when the mobile device is near a handoff point.

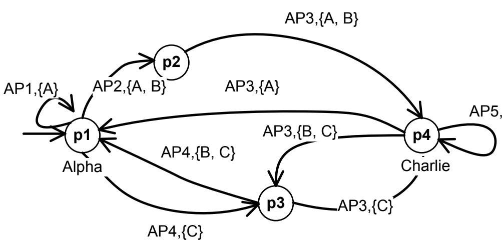


Fig. 11. Connectivity graph created in an indoor testbed [13].

The size of accumulated data as function of time and the power consumption until the end of the path are shown in Table II. Using the policy **constant $>$80%** as base line, the two preemptive policies $>$ 80**%** and **80%**, saved 30% and 75% respectively. It is important to mention that the three policies had equivalent QoS performances. Policies to discover 60%, of the connectivity opportunities, or less had inferior QoS performances than the ones in Table II. For this reason they are not listed in the table. It, also, indicates a trade-off between saved resources and improved QoS performances in WiFi networks.

Method (%)	Storage overhead (Kbytes)	Power consumption (Joule)
Constant >80%	7.31	0.78
Preemptive >80%	5.06	0.54
Preemptive 80%	1.12	0.12

TABLE II
IMPROVEMENTS IN STORAGE OVERHEAD AND POWER CONSUMPTION.

VI. EXPERIMENTAL RESULTS

The comparative evaluation is done for 3 connectivity management mechanisms, namely: **(A)** strongest signal strength (SSS), **(B)** mobility predictor, and **(C)** community-based. **SSS** is the base line for the comparison. It is the current technique employed by common operational systems, e.g. Windows Mobile and Android. The next access point is chosen taking the one with highest signal strength [16]. The **mobility predictor** uses a second order Markov model calibrated with the specific user movement history, similar to the one discussed in [16]. Finally, the **community-based** mechanism employs the same predictor, however, the socialized connectivity graph was used to build the model and to assist the handoff decision making [13].

A. Throughput

These mechanisms were implemented in a prototype. It was used by 3 persons during 4 months in an indoor IEEE 802.11a/g testbed, with about $1,600m^2$ of area, covered by 5 access points. The mobile devices measures the throughput performing get and posts to a dedicated HTTP server, connected to the local network. In the first month, the 3 mechanisms presented similar irregular performances. Then, the mobility predictor was calibrated with the first month data. In the following months we observed significant performance differences between the 3 connectivity management mechanisms.

The community-based method had a superior throughput, about 13 and 23 % higher, than the other two. On average, the performance was 3.8 Mbps against 3.3 and 2.9 Mbps, for mobility predictor and SSS respectively. Fig. 12 shows the histogram, and the tendency function for each method. For SSS, Fig. 12 (A), the majority of the values are concentrated below 3.5 Mbps. On other hand, mobility predictor (B) and community-based (C) methods concentrate the values above 3.0 Mbps, the maximum throughput observed was 4.0 Mbps. Comparing Fig 12 (C) with the others two, we conclude that the proposed solution avoided bad QoS conditions inside the testbed.

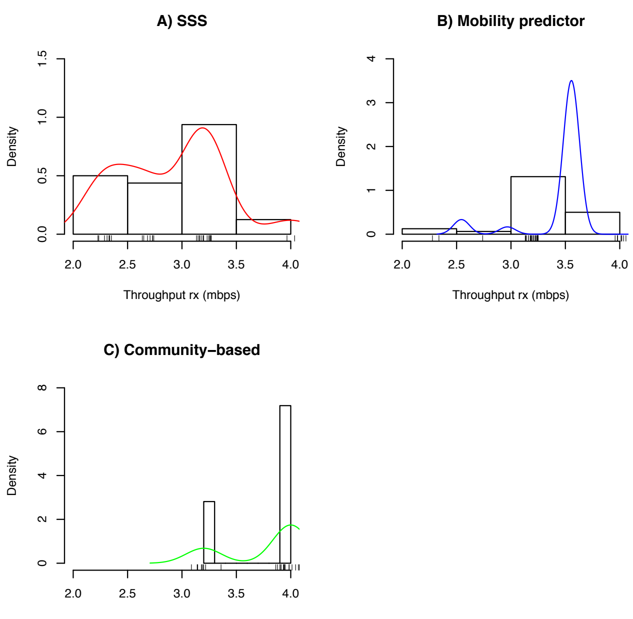


Fig. 12. Histogram of observed throughput (rx), during the experiments, for the three mechanisms: (A) SSS, (B) Mobile predictor, and (C) Community-based.

B. Signal quality

The difference among the mechanisms is also evident when we look to the signal quality. Fig. 13 shows the signal quality as function of time, during free walking in the testbed. The three mechanisms were used through the same path, doing HTTP requests to the server. The average signal quality of the three methods are 55, 64 and 78%, for (A) SSS, (B) mobility predictor and (C) community-based, respectively. This confirms that it is possible to have better connectivity experiences by improving the quality of the context data employed. In addition, the time to discover a wireless environment can be reduced as function of the number of mobile users collaborating.

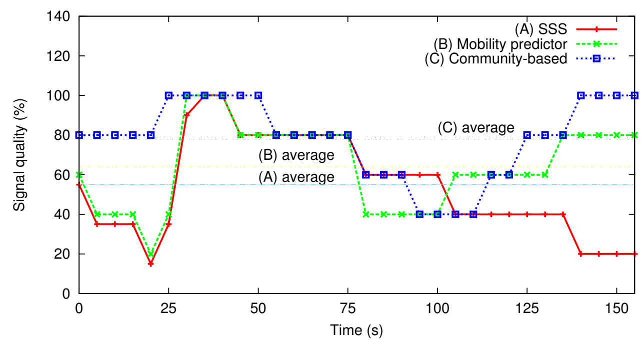


Fig. 13. Signal quality during a arbitrary roaming in the testbed for the 3 methods: (A) SSS, (B) mobility predictor, and (C) community-based.

VII. Acknowledgment

This work was supported by FAPESP (grant 2008/06862-6) and Capes (grant 3993-08-6).

VIII. Conclusions

Context-aware connectivity management is the main issue of several investigation reported in the literature. Some of these researches motivated us to design a social-based methodology to improve the quality of the context data gathered and stored. A prototype has been developed to prove the concept, and experiments were performed aiming at a quantitative evaluation. Other metrics such as freshness and updatability, of network context data, would be improved with intense collaboration of people inside the virtual social circle. Also, the handiness of the data can be enhanced by the efficient delivery system developed.

The primary contribution of this paper is the orchestration of fundamental tasks for context-aware connectivity management in a methodology. It is an user-centric solution, which explores virtual social circles to allow collaboration among mobile users. The socialized connectivity graph can be applied to save storage size, power consumption and QoS, while the user is moving. Finally, the comparative evaluation showed promising quantitative results.

This investigation is unfolding with focus on the usage of socialized context data to find the second or n^{th} access point. In addition, efforts have been done to determine optimal QoS conditions and localization to download the shared RSS feeds.

REFERENCES

[1] P. Bellavista, A. Corradi, and L. Foschini. Context-aware handoff middleware for transparent service continuity in wireless networks. *Pervasive and Mobile Computing*, 3(4):439 – 466, 2007. Middleware for Pervasive Computing.

[2] C. Bettini, O. Brdiczka, K. Henricksen, J. Indulska, D. Nicklas, A. Ranganathan, and D. Riboni. A survey of context modelling and reasoning techniques. *Pervasive and Mobile Computing*, 6(2):161 – 180, 2010. Context Modelling, Reasoning and Management.

[3] J. Chan, S. Zhou, and A. Seneviratne. A qos adaptive mobility prediction scheme for wireless networks. In *Global Telecommunications Conference, 1998. GLOBECOM 98. The Bridge to Global Integration. IEEE*, volume 3, pages 1414 –1419 vol.3, 1998.

[4] P. Chan, R. Sheriff, Y. Hu, P. Conforto, and C. Tocci. Mobility management incorporating fuzzy logic for heterogeneous a ip environment. *Communications Magazine, IEEE*, 39(12):42–51, dec 2001.

[5] R. Chandra and P. Bahl. Multinet: connecting to multiple ieee 802.11 networks using a single wireless card. In *INFOCOM 2004. Twenty-third AnnualJoint Conference of the IEEE Computer and Communications Societies*, volume 2, pages 882 – 893 vol.2, 2004.

[6] C. Desset, N. Ahmed, and A. Dejonghe. Energy savings for wireless terminals through smart vertical handover. In *Communications, 2009. ICC '09. IEEE International Conference on*, pages 1 –5, 14-18 2009.

[7] L. Eastwood, S. Migaldi, Q. Xie, and V. Gupta. Mobility using ieee 802.21 in a heterogeneous ieee 802.16/802.11-based, imt-advanced (4g) network. *Wireless Communications, IEEE*, 15(2):26–34, april 2008.

[8] M. H. Falaki. Wlan interface management on mobile devices. Master's thesis, University of Waterloo, Waterloo, Ontario, Canada, 2008.

[9] A. Hasswa, N. Nasser, and H. Hassanein. A seamless context-aware architecture for fourth generation wireless networks. *Wireless Personal Communications*, 43:1035–1049, 2007. 10.1007/s11277-007-9262-7.

[10] T.-H. Kim, Q. Yang, J.-H. Lee, S.-G. Park, and Y.-S. Shin. A mobility management technique with simple handover prediction for 3g lte systems. In *Vehicular Technology Conference, 2007. VTC-2007 Fall. 2007 IEEE 66th*, pages 259 –263, sept. 2007.

[11] A. LaMarca, J. Hightower, I. Smith, and S. Consolvo. Self-mapping in 802.11 location systems. In *UbiComp 2005: Ubiquitous Computing*, volume 3660 of *Lecture Notes in Computer Science*, pages 87–104. Springer Berlin / Heidelberg, 2005.

[12] N. Li and G. Chen. Analysis of a location-based social network. In *Computational Science and Engineering, 2009. CSE '09. International Conference on*, volume 4, pages 263 –270, aug. 2009.

[13] R. R. F. Lopes, B.-J. van Beijnum, and E. S. Moreira. Managing wireless ip-connectivity experiences as mobile social media. In *International Congress on Ultra Modern Telecommunications and Control Systems (ICUMT)*, pages 154 –161, oct. 2010.

[14] D. Ma. Offering rss feeds: Does it help to gain competitive advantage? In *System Sciences, 2009. HICSS '09. 42nd Hawaii International Conference on*, pages 1 –10, jan. 2009.

[15] A. J. Nicholson, Y. Chawathe, M. Y. Chen, B. D. Noble, and D. Wetherall. Improved access point selection. In *MobiSys '06: Proceedings of the 4th international conference on Mobile systems, applications and services*, pages 233–245, New York, NY, USA, 2006. ACM.

[16] A. J. Nicholson and B. D. Noble. Breadcrumbs: forecasting mobile connectivity. In *MobiCom '08: Proceedings of the 14th ACM international conference on Mobile computing and networking*, pages 46–57, New York, NY, USA, 2008. ACM.

[17] A. J. Nicholson, S. Wolchok, and B. D. Noble. Juggler: Virtual networks for fun and profit. *Mobile Computing, IEEE Transactions on*, 9(1):31 –43, jan. 2010.

[18] P. Pawar, B.-J. van Beijnum, K. Wac, H. Hermens, and D. Konstantas. Towards location based qos-aware network selection mechanism for the nomadic mobile services. In *Consumer Communications and Networking Conference, 2009. CCNC 2009. 6th IEEE*, pages 1 –5, jan. 2009.

[19] P. Prasad and P. Agrawal. A generic framework for mobility prediction and resource utilization in wireless networks. In *Communication Systems and Networks (COMSNETS), 2010 Second International Conference on*, pages 1 –10, 5-9 2010.

[20] C. Prehofer, N. Nafisi, and Q. Wei. A framework for context-aware handover decisions. In *Personal, Indoor and Mobile Radio Communications, 2003. PIMRC 2003. 14th IEEE Proceedings on*, volume 3, pages 2794 – 2798 vol.3, 2003.

[21] L. Song, U. Deshpande, U. C. Kozat, D. Kotz, and R. Jain. Predictability of wlan mobility and its effects on bandwidth provisioning. In *INFOCOM 2006. 25th IEEE International Conference on Computer Communications. Proceedings*, pages 1–13, april 2006.

[22] J.-Z. Sun and J. Sauvola. From mobility management to connectivity management. *Computers and Communications, IEEE Symposium on*, 0:307–312, 2005.

[23] M. Tlais and F. Weis. Centralized mobility prediction support in a hierarchical architecture. In *Wireless and Mobile Communications, 2006. ICWMC '06. International Conference on*, pages 81 –81, 29-31 2006.

[24] K. E. Wac, R. G. A. Bults, B. J. F. van Beijnum, H. Chen, and D. Konstantas. Toward mobile web 2.0-based business methods: Collaborative qos-information sharing for mobile service users. In M. Head and E. Li, editors, *Mobile and Ubiquitous Commerce*, volume 4 of *Advanced E-Business Methods*, pages 60–81. Information Science Reference, UK, May 2009.

[25] E. Yami and V. Vakili. Adaptive bandwidth allocation and mobility prediction in mobile cellular networks. In *Next Generation Mobile Applications, Services and Technologies, 2009. NGMAST '09. Third International Conference on*, pages 343 –348, 15-18 2009.

[26] H. Yokota, A. Idoue, T. Hasegawa, and T. Kato. Link layer assisted mobile ip fast handoff method over wireless lan networks. In *MobiCom '02: Proceedings of the 8th annual international conference on Mobile computing and networking*, pages 131–139, New York, NY, USA, 2002.

[27] J. Yu, B. Benatallah, F. Casati, and F. Daniel. Understanding mashup development. *Internet Computing, IEEE*, 12(5):44 –52, sept.-oct. 2008.