

MetroSim: A Planning Tool for Metropolitan WiFi Networks

Gilberto A. Urueta
Deutsche Telekom Labs
Berlin, Germany
gilberto.urueta@telekom.de

Christopher Thraves^{*}
LaBRI, Université Bordeaux I
Bordeaux, France
cbthraves@labri.fr

Marcin Solarski[§]
Deutsche Telekom Labs
Berlin, Germany
marcin.solarski@telekom.de

Pablo Vidales[×]
Deutsche Telekom Labs
Berlin, Germany
pablo.vidales@telekom.de

ABSTRACT

Massive growth of home wireless networking revives the opportunity to achieve citywide ubiquitous wireless access by building a WiFi overlay based on the principle of *broadband sharing*. This kind of large-scale overlay network poses new challenges for wireless network planning. In this scenario, the number of possible locations to install public hotspots is larger, by order of magnitude, to those managed by existing WiFi planning tools. Moreover, dynamics due to a loose control on the WiFi hotspots (as routers are not 100% controlled by the operator, but by the router owner) means hard constraints on the flexibility of the deployment. This paper describes **MetroSim**: a novel planning tool to drive the deployment of metropolitan WiFi networks based on enabling safe and controlled broadband sharing. MetroSim considers the aforementioned challenges in its design and implementation, as it has been developed specially to help planning this type large-scale metropolitan WiFi overlay networks.

Categories and Subject Descriptors

D.2 [Software Engineering]: General; H.4.0 [Information Systems Applications]: General

Keywords

Software systems, network planning, wireless networks

^{*}Christopher Thraves is partially supported by the French ANR project Alpage.

[§]Marcin Solarski is a Manager of IBM Software Services since December 2008 (email: marcin.solarski@pl.ibm.com)

[×]Pablo Vidales is member of the National System of Researchers (SNI) in Mexico since January 2007.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

SIMUTools '09 Rome, Italy

Copyright 2009 ICST ISBN 978-963-9799-45-5.

1. INTRODUCTION

The massive growth of home wireless networking presents a new opportunity to advance in what for some years back has been seen as a research vision: ubiquitous wireless access. Having a fixed broadband penetration approaching 25 per cent in the EU 15 countries [13], which translates into almost 100 million lines, most of which are terminated with WiFi gateways, creates a large base for a WiFi network. If only these nodes could be part of a shared infrastructure that is cost-effectively built and maintained, the ubiquitous Internet access vision would come true.

The mobile data market is only starting to grow with 7,2 million subscribers in the world. Despite the market saturation with cellular phones, the prevailing way of using the phones is voice service. The mobile Internet is still used by a limited segment of subscribers due to high usage costs resulting from the huge investments in the spectrum license acquisition and infrastructure equipment from cellular operators.

The popularity of WiFi-enabled devices along with less expensive WiFi pricing models, especially those bundle by ISPs with the provisioning of home broadband access for private customers, appears more promising and appealing for the mass market. Unfortunately, the current WiFi coverage is still very scattered and limited to premium locations, meaning no feasible alternative to cellular networks. To make the vision of ubiquitous Internet using the WiFi technology, two points need to be addressed:

1. Significantly higher number of WiFi access points. Because of the small coverage radius inherent to WLAN technology, only increasing the number of WLAN access points can make the total coverage significant.
2. Cost-effective WiFi deployment. Because of large number of required WiFi hotspots, an efficient planning for the WiFi locations must be performed. In addition, the cost of installation of individual gateways should be as low as possible.

The core idea of metro WiFi network based on sharing the broadband connections overcomes the above challenges. First, it leverages on the large number of broadband connections in the residential areas. Second, it reuses the existing broadband infrastructure through secure and controlled sharing of the broadband access.

There are many initiatives tackling these challenges, for example, many governments are pushing the idea of offering citywide wireless access as part of the basic infrastructure, like Paris [6], Berlin, Philadelphia and San Francisco. In the industry, there are two different trends that are being followed. On the one hand, operators are expanding the coverage of cellular-a-like wireless networks such as WiMax, EDGE, or UMTS. On the other hand, small companies like Fon [12] and Sharefi [15] are proposing an alternative: forming communities that share Internet access via WiFi routers.

The latter broadband sharing approach is very promising due to a huge number of WiFi-enabled residential gateways that can potentially become access nodes of the metro WiFi network. However, to make this approach feasible at large, several technical challenges have to be solved:

- *The scalability of the backend functionalities.* A metro WiFi network may contain dozens of thousands of access nodes in a large city. This number may grow by one or even two orders of magnitude if the same service provider has to operate multiple WiFi networks distributed across cities, regions or even countries. As an example, for security reasons, the public traffic is usually encrypted and tunneled across the connections used by the home customers to the aggregation network. These network components need to efficiently handle the load of the concentrated traffic from all the leaf nodes.
- *Capacity bottlenecks in the broadband access network.* Even though that the most frequently deployed version of WLAN technology (i.e. IEEE 802.11g) allows up to 54 Mbps for the gross channel capacity, it cannot be fully used as it is the bandwidth of the access network that is limiting the end-to-end channel capacity. In fact, the downlink bandwidth for ADSL1 networks is limited to 6 Mbps and ADSL2p to 16 Mbps, and the uplink to 3 Mbps and 10 Mbps, respectively. An additional reduction of the available bandwidth comes from the very fact of sharing the broadband connection.
- *Reinforcing security for shared connections.* On opening the broadband network used so far by the broadband customers, additional security mechanisms need to be enforced. For user traceability, the public and private traffic needs to be separated. In addition, no direct communication shall be enabled between users connected to the same access point.
- *Reduction in the control of the infrastructure.* Since the access points are located in such a shared network at the customer premise, limited physical access to the devices is possible. In contrast to a purely public network, the power cycle of the shared access points is controlled by the customers, who may switch off and on the devices as they pleased. Moreover, the maintenance cannot be performed as frequently as for public networks and in case of a malfunctioning hardware, reparation or replacement cannot be done only with cooperation of the broadband customer. All this affects the access point availability.
- *Higher risk of radio interferences.* Due to an increasing number of WiFi access points deployed and their high density in residential areas in particular, as well

as to limited number of non-interleaving radio channels available for the IEEE 802.11 technology, radio interferences may occur more often. Efficient channel allocation schemes with logic analyzing the spectrum usage around the individual access points are needed.

- *Limited location planning.* The planning of the access point location is difficult. First of all, the access points can be placed only at certain locations, i.e. the apartments, where broadband customers live and may install these. Secondly, not all such possible locations can be used in reality as it is the broadband customer that decides to enable such service or not. In addition, if the customer decides for it, he or she is free to chose the detailed location within his apartment or house. The latter has an impact on the signal propagation in this particular location because of the physical diffraction due to local objects and obstacles at the customer premise. Experimental results in [8] gives some insight into this topic. Another difficulty in planning the hotspot location is incomplete and imprecise information on the access distribution available. There is a big amount of access points with a relatively small coverage range at various locations (30 m radius). As a result, the complexity of the deployment problem is high as it demands the analysis of high data volumes. In addition, the selection of good possible points for hotspot deployments bounded by the available demand information.

This paper addresses the last challenge, while considering all the other aspects listed above. MetroSim helps operators in the process of opening up existing broadband connections to nomadic users, obtaining a low-cost and efficient deployment for provisioning mobile Internet access. The idea of MetroSim follows a twofold approach. First, the design of a model representing the locations of users and the broadband infrastructure was conceived. This model includes the implementation of an algorithm to determine a low-cost and efficient deployment. Second, MetroSim includes an application that provides reports and visualization methods to facilitate the decision-making process when planning metro WiFi networks.

The rest of the paper is organized as follows. Section 2 introduces MetroSim and discusses the main design aspects and architecture of this network planning software. Next, the case deployment algorithm employed for the evaluation of MetroSim is described in Section 3. The experiments and evaluation of the tool are discussed in Section 4. Other relevant network planning tools are mention in Section 5. Finally, Section 6 includes conclusions and future work.

2. METROSIM

MetroSim is a specialized software tool that aims to assist in the creation of citywide WiFi deployments. It allows users to analyze locations with broadband access infrastructure and select those with the highest profit according to an objective function. The analysis can be done using different criteria such as (1) deploy hotspots until a predefined budget is spent, (2) using a specific number of routers or (3) trying to cover the whole Internet access demand. Based on this criteria, the user is able to generate deployment guidelines, reports, and visualize the results.

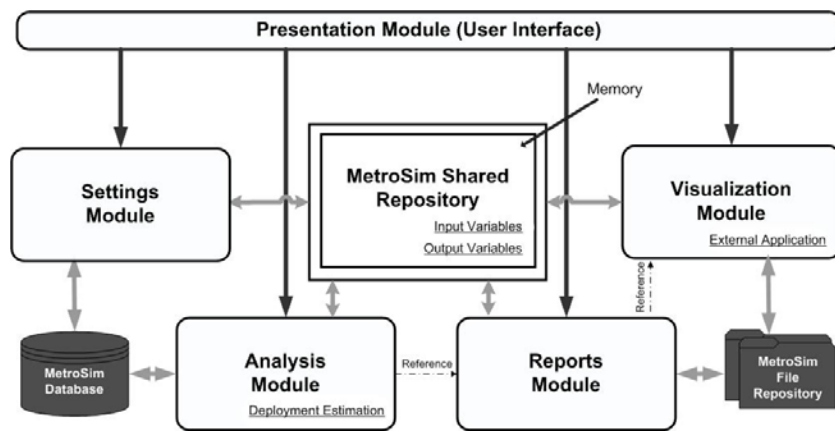


Figure 1: MetroSim Architecture

In the version described in this paper, MetroSim applies an objective function proposed by Thraves et al. in [9]. Nonetheless, our tool is not tied up to the mentioned algorithm due to the modularity of its architecture that allows the use of different solutions. To visualize the output deployment, the results are processed into Keyhole Markup Language (KML) and stored into files. These can be viewed on a real world map using a third party application supporting KML.

2.1 Architecture

After discussing the design principles (previous section) the overall architecture of MetroSim was defined (see Figure 1). This section details the main components of the architecture, as well as the most relevant functional aspects. The leftmost module, in Figure 1, is the **Settings module**: it allows the user to load and modify the system settings for MetroSim, which are stored in the **MetroSim Database**. The system settings include input variables such as possible locations, routers, and Internet access demand at certain locations.

Moreover, the **Analysis module** is in charge of loading the system settings from the MetroSim Database into the **MetroSim Shared Repository**, hence, making it available to other modules. After loading the settings, this module checks all possible locations to select the most appropriate one for sharing the broadband access, forming a cost-efficient overlay WiFi network. The responsible component for this estimation is called **Deployment Estimation** that implements a deployment algorithm to evaluate the possible locations (the case deployment algorithm used in MetroSim, for evaluation, is described in Section 3).

In the case algorithm, in order to suggest a potential deployment the following aspects are considered: a limited budget, number of deployments, and until the Internet demand is fulfilled. The result is a list of suggested locations and the final demand that will be covered when all listed locations are deployed (all the results are stored in the MetroSim Shared Repository).

After the execution of the analysis, the **Reports module** starts the process of generating reports and storing them in the **MetroSim File Repository**. The Reports module receives input from the Analysis module, stored in the MetroSim Shared Repository, and based on this data generates

the following reports:

- *Initial demand.* This report shows the access demand in the area before any deployment. This report requires two input variables: the possible locations selected by the user and the initial demand. Both are retrieved from the *MetroSim Shared Repository*.
- *Remaining demand.* This report has the same functionality as the previous one, with the difference that in this report the demand considers the suggested locations for deployed. As the previous report, it needs all possible locations and the final demand.
- *Suggested locations.* This report contains the final list with the suggested deployments.
- *Used and unused locations.* This report shows the number of locations that were considered for the deployment and those that were not used. It requires the input variables possible locations and suggested locations.
- *Demand covered.* This report presents the demand that was fulfilled at the end of the analysis. It is calculated with the difference between the remaining demand and the initial demand.
- *Cost of deployments.* This report includes the cost of deploying the routers at all suggested locations.

The analysis can be done using the following constraints: a budget, a number of deployments or until the Internet demand is satisfied. The result is a list suggesting locations and the final demand that will be generated if the locations were deployed. Both are stored in the *MetroSim Shared Repository*.

With the exception of the last report, all the others can be visualized in a real world map. In order to start this process, the Reports module stores the results of the calculation in the repository and sends a reference to the **Visualization module**. The Visualization module uses this information to generate the visualization requested by the user. The output of this module is a KML file. This format consists of a XML-based language schema for modeling and storing

geographic features. Using this format, geographical applications like Google Earth can be used to display the results in a real world map. These files are stored in the MetroSim File Repository.

Finally, the interaction between the system and the user is managed by the **Presentation module**. The user is able to manage the access to every module independently. The Presentation module is independent from all other components of the system, hence, it can be easily modified without changing the system logic. The current version uses a graphical user interface to interface with the users.

2.2 Design Rationale

The design and specification of MetroSim triggered decisions that could face the emerging challenges pose by city-wide WiFi networks. These decisions determine the overall system structure of MetroSim. In the following, these key issues are listed, as well as the approach that was used to overcome each of them.

1. The architecture has to facilitate the development of different software properties. First of all, scalability in terms of adding new features and minimizing development effort to alter the main system. Second, ease of modification of the developed features, thus, the system can be enhanced and maintained. Finally, support of reuse of the system's features which can stimulate future implementations.
2. The deployment results have to be visualized in a real world map.
3. The deployment algorithm proposed in [9] has to be modified and enhanced to fit into a module of the present planning tool.

The first challenge involved the design of the overall architecture. To achieve this objective, the design went through different stages. In a first stage, the architecture consisted of independent modules connected through "pipes". This independence allows each module to be modified without affecting the others. Furthermore, since each module performs an entire independent task, they can be reused in future implementations. The communication among modules is done through pipes that connect the output coming from a module serving as the input of another. The whole design was based on a traditional architecture style known as *Pipes and Filters* described in [3]. In this book, the authors presented two additional architecture styles, which are also mentioned in this section: the *Object-Oriented Organization* and the *Repository*.

The early design of MetroSim solved many of the issues but presented a number of disadvantages. First of all, the communication among modules required copying the whole data from one module to the other. Second, in order to send the information, a process to parse this data was needed adding overhead to the whole process. Finally, a common denominator to transport the data was needed to interchange information.

To deal with these problems, a combination of the architecture styles *Object-Oriented Organization* and *Repository* were introduced to the design. The first one, involve the use of object-oriented techniques to encapsulate and manage data. The use of objects brought the following changes

to the design: the information was encapsulated and kept organized, the parsing of objects was replaced with the process of accessing object attributes, and an object repository to exchange information was implemented. This last change is part of the *Repository* architecture, which includes a central data structure operated by independent components.

The implication of using an object repository was a modification in the communication method. The design of the pipes was modified and adapted to transport references of objects. These objects were stored in a central data structure which is kept in memory. The latter avoids the need of copying all the information from one module to another. Further, the use of a central data structure accessed by independent modules provides scalability to the system. Additional modules can access the repository without modifying other components.

The main drawback of the design was the amount of memory required by the system. As an alternative to the previous, the use of databases and random access files as a central data structure could reduce this burden. Nevertheless, access to physical storage is traditionally considered as the most time-consuming task.

Finally, the user requires interaction with the system. Therefore, a new module connecting the user and the system was included in the overall design. Since every module works independently, users are responsible for deciding the moment to execute each one. Nonetheless, each module has a sequential order that guides this decision. The detailed architecture of the application will be described in the next section.

The second challenge consisted in the visualization of the resulting metro WiFi network in a real world map. The alternatives evaluated for this process were: (a) using actual commercial or open-source geographical information systems, (b) development of an application for the complete process, (c) development of an application that uses commercial services available in the market. The evaluation of these approaches was based on cost and time for development.

The primary disadvantage of the first alternative was the need to create or purchase real world maps. Moreover, in order to integrate the application to the current system, it had to be understood and modified with the cost of time and effort. In a similar way, the second alternative required also the creation or purchase of real world maps. Moreover, it required a higher time for the development process compared to the previous alternative.

3. CASE DEPLOYMENT ALGORITHM

A core piece of MetroSim is the one responsible for computing the deployment considering all input datasets. A case deployment algorithm based on [9] was implemented in MetroSim (i.e. *Analysis module* in Figure 1).

For the case algorithm, the goal is to cover as much demand as possible, while constraint by a budget. The demand function N uses data from the Internet traffic (kbps) in the collocated cellular network to calculate a demand value for each possible deployment location. It is assumed that the data traffic in a cell can reflect the local need for data transmission in the deployment area.

In order to process the demand values associated to a zone, which corresponds to an specific coverage area, these are loaded from the MetroSim database into a matrix structure. For example, the matrix value i, j represents a point in the

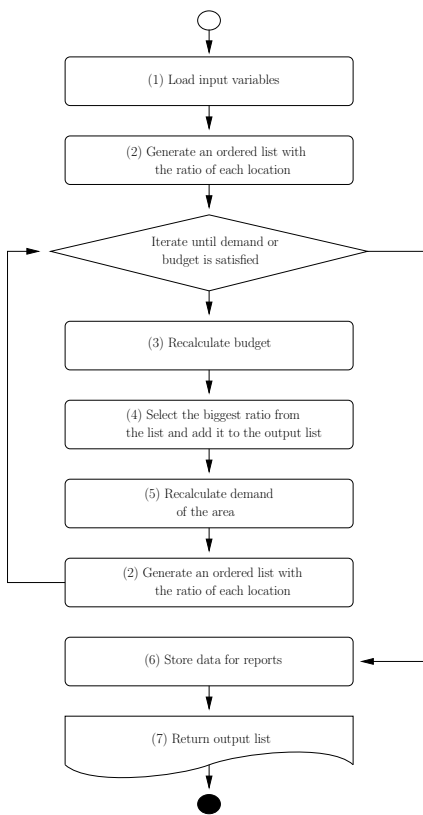


Figure 2: Case deployment algorithm

total area and the value $m_{i,j}$ corresponds to the demand at this point. The size of the geographical region for the deployment directly influences the size of the matrix, hence the memory demand to process the demand.

For the case of Berlin, to simulate a metropolitan WiFi network that covers the complete city, which has an area of $\sim 900 \text{ km}^2$, a matrix of (8000,18000) would be needed. According to our experiments which consisted of monitoring the memory needed to load different sizes of matrix, less than 2 GB of RAM would be required, well below from the memory capacity of commercial computers. Moreover, MetroSim can reduce its system requirements by lowering accuracy in the analysis (i.e. geographical distance between demand points).

A router $w \in R$ is defined by its cost, type, coverage and capacity, using tuples. These router characteristics are based on an experimental assessment of WiFi networks in urban areas performed by SolarSKI et al. [8]. A deployment D has associated a total cost of deployment $C(D)$. The cost function $C(D)$ is determined by the unitarian cost of the router based on market prices, plus an additional component that includes incentives, installation and operation costs. Finally, the total budget available for a particular metropolitan deployment is determined by the user. Next, the deployment algorithm is explained.

In **step (1)**, all possible deployment locations are loaded into RAM from the MetroSim database. Each location is represented as a tuple of latitude, longitude, associated address, attributes of the router at this location, and the data related to the available broadband access (e.g. speed of the

DSL connection). Figure 2 shows the main steps of the case algorithm.

Then, during **step (2)** the following equation is computed for each router $w \in R$, placed at point $p \in P(p, w)$:

$$\frac{1}{C(p, w)} \int_{F(p, w)} \max\{N(q) - D(q), 0\} dq, \quad (1)$$

where $F(p, w)$ represents the area in which a router w can guarantee minimum connectivity service when placed at point p , and $C(p, w)$ is the cost of placing w at location p . Finally, $N(q)$ represents the actual existing demand at point q and $D(q)$ the demand covered by w at point q . Therefore, $\max\{N(q) - D(q), 0\}$ computes the new demand at point q after place w in p .

Based on this list, an evaluation is performed and the most appropriate locations are selected (iteration) until all possible demand is fulfill or the available budget is finished. Also, in **step (3)**, the budget is reduced for every new deployment. Next, in **step (4)**, the location with the most benefit is chosen based on the list generated in step (2) and inserted into the output list. The selection of a location triggers a modification of the demand, which is reflected in **step (5)**.

To calculate the new demand, the coverage ratio for each possible location has to be recalculated. This is shown in Figure 2 by the arrows pointing back to step (2). Finally, **step (6)** stores the results generated by the case deployment algorithm to generate different report, and **step (7)** returns the output list with the suggested metropolitan WiFi network to be deployed.

4. EVALUATION AND RESULTS

This section discusses the evaluation of MetroSim. As it is explained above, MetroSim may be executed until a predefined budget is satisfied, or until the complete access demand is fulfill. The results in this section consider the latter criterion.

The dataset used to evaluate MetroSim contains information describing the real distribution of broadband connections in a neighborhood of Berlin. The data includes completely anonymous geographical locations of exactly 29,658 DSL-lines that represent the set of possible places to locate a router. These addresses were grouped by its zip codes, getting as a result 23 different zip codes. Hence, the region to study is chosen accordingly with its zip codes. In this case, the zip codes contain in average around 1,289 DSL-lines. MetroSim is executed several times increasing exponentially the number of zip codes in the input, i.e., first it is executed with 1 zip code, and then with 2, 4, 8, 16 and 23, were 23 is the biggest region for this data.

The service usage in cellular networks may be a starting point to estimate the future demand of nomadic users in an open broadband network. Pursing this prospect, the initial demand is computed using information collected from a cellular network located over the same geographical area in Berlin. It is based on the IP traffic data gathered from the cellular base stations. For each base station, its total IP traffic is assigned to its coverage zone. And then, the initial demand is computed summing up over all the coverage zones.

For this experiment the router characteristics are set to 35 m for the radius coverage, and 54 Mbps for the service

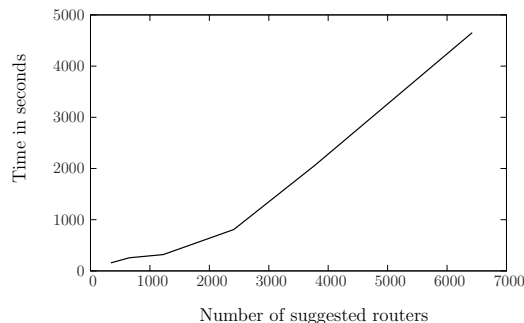


Figure 3: Executing Time: Time taken by MetroSim executing the experiments

capacity. Moreover, the bandwidth offered by the DSL-lines contains 9 different values, 384 *Kbps*, 768 *Kbps*, 1 *Mbps*, 2 *Mbps*, 5 *Mbps*, 6 *Mbps*, 16 *Mbps*, 25 *Mbps* and 50 *Mbps*. Nevertheless, only the 25% of the bandwidth is considered since a loss in capacity due to external factors (i.e interference) is assumed. The radius of the router was based on the experimental assessment of WiFi networks in urban areas performed by Solarski et al. [8]. The service capacity was taken from existing commercial technical specifications.

The experiments were executed in a Fujitsu-Siemens laptop, with *Intel® Pentium® M Processor 1.73 GHz* of clock speed and 0.99 *GB* of *RAM*. In this context, the main performance evaluation of MetroSim is the computing time. The time spent by MetroSim executing the experiments is shown in Figure 3. The X-axis represents the number of suggested locations and Y-axis the time spent by MetroSim computing such locations. The number of suggested locations is directly related with the amount of iterations that MetroSim performs over its algorithm. Each time that a new suggested location is added to the result, the algorithm review and compare all the possibilities. Thus, the growth on the time used to end up with a result is directly affected by the size of the list of possible locations. The results shown that the computing time is an important point to improve for the case of huge inputs. In the case in which MetroSim suggest around 6400, it takes more than 1 hour and 15 minutes to compute a solution. However, taking advantage of the MetroSim architecture, it is possible to change the algorithm to one that deal better with such a extreme cases, for instance use an evolutionary algorithm.

Derived from the real data used in these experiments, it is possible to conclude with interesting comments about the challenges mentioned in the introduction. Figure 4 shows the coverage ratio of the service demand by the service provided for a network build following MetroSim. The X-axis represents the number of suggested locations and Y-axis the service provided by the suggested locations divided by the total service demand. The results showed that a huge effort is needed to cover the complete demand in cellular networks using broadband networks. When MetroSim can chose all the DSL-lines in the studied region, it proposes 6, 423 possible locations and the coverage ratio is only 0.4280, less that 1/2 the total demand. Moreover, it is possible to observe that the growth of the rate is too slow even when the possible locations increase exponentially, basically due to the different capabilities (lets say coverage and service provided) between a cellular network and broadband network. Never-

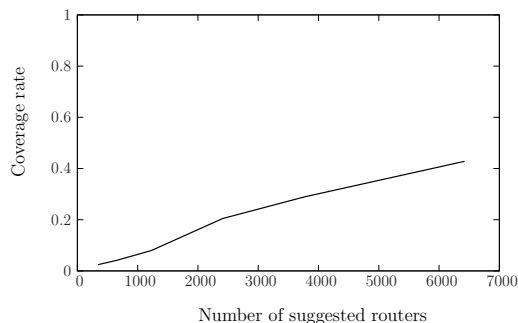


Figure 4: Coverage Ratio: Coverage rate of the demand computed using cellular networks by the WiFi network provided by MetroSim

theless, it lets an interesting door open, to study a mixed solution were customers can move from a cellular network to a broadband network depending on the type of service they requires, normal telephony service or IP service.

Figure 5 shows the evolution of the service capacity of the deployments suggested by MetroSim while the number of suggested routers is increasing. The X-axis represents the number of suggested locations and Y-axis the service provided by the suggested locations measured in *Kbps*. It is possible to interpret the growth as a linear growth, however, using the information about the coverage ratio also it is possible to notice that still there are a huge amount of service demand to cover. Hence, the routers are used at its best discarding intersections among its coverage region. When the service provided is closer to the service demand, it will be more difficult to open up a router such that its coverage region is not intersecting an already covered area. In such case the growth would be slower.

Finally, one important MetroSim contribution is the visualization of the results. As it was explained in section 2, the results can be visualized using geographical applications like Google Earth. Figure 6 shows the visualization of the results of an experiment in which MetroSim proposed 13, 706 locations. Figure 6(a) shows the locations evaluated by MetroSim to open up the routers, also in this figure there is a note space in which may be included specific information about the locations. Figure 6(b) shows the initial service demand in the region, a third dimension is used to express the size of the service demand. Figure 6(c) shows the unfulfilled service demand after open the locations suggested by MetroSim.

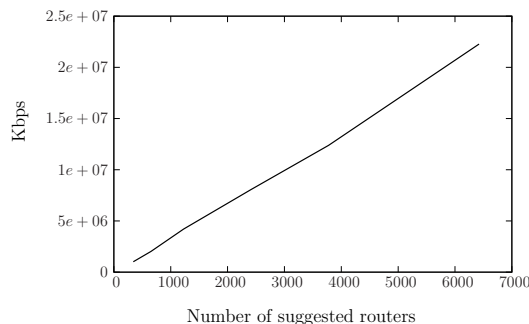
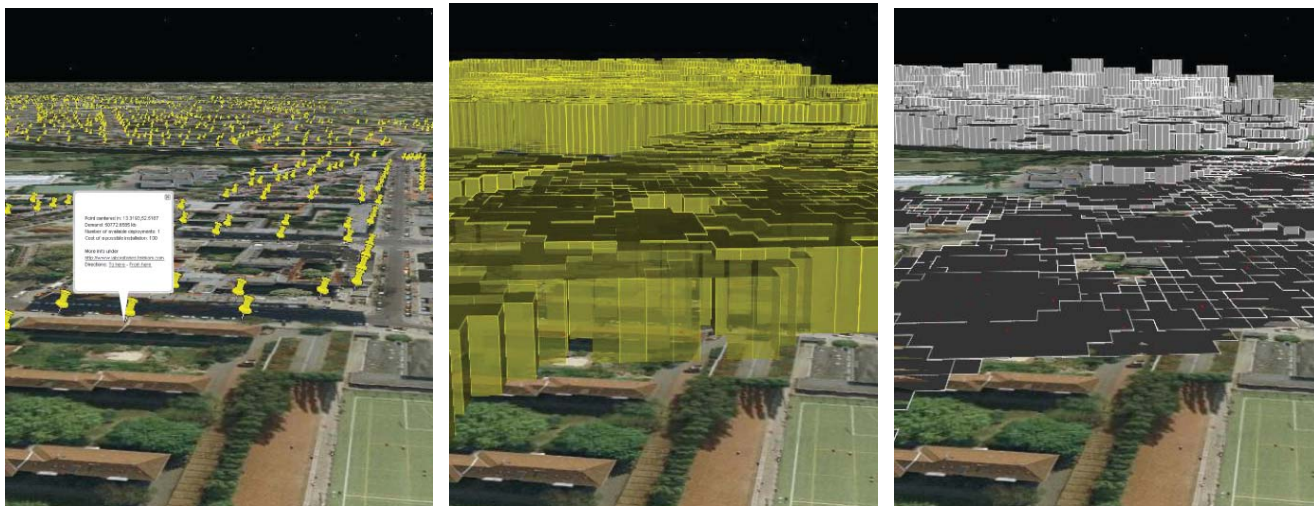


Figure 5: Service capacity of the deployment

Figure 6: MetroSim visualization



(a) Potential locations for WiFi hotspots.

(b) Initial demand function.

(c) Unfulfilled demand after deployment.

5. RELATED WORK

Therefore, the third alternative was chosen since the cost and development effort of the solution was the lowest. It was decided to create a visualization module that processes and stores the results into a file. The structure of this file has a format that can be visualized by an external geographical application (e.g. Google Earth). The disadvantages of this approach is the dependency to an external application that constraints the format and the possibilities of display. However, this disadvantage can be reduced by using open standards common to several applications. MetroSim uses OpenGIS Keyhole Markup Language 2.2 Encoding Standard (OGC KML) to generate the files.

The last design principle involved the implementation of the solution described in [9] that selects locations from existing broadband infrastructure to build an overlay WiFi access network. Section 3 explains this algorithm in detail. The module that implements the case deployment algorithm is the core of the system, however, the modularity of the tool allows the use of other solutions and algorithms, which may consider similar input and output variables.

In the area of broadband sharing and citywide deployments several works have been done. Crowcroft et al. [7] presented a mechanism to enable safe WiFi sharing with legitimate guests. They described how to architect a citywide cooperative network based on secure tunneling of the data and discuss some security concerns related to this type of deployments. Also, Hakegard et al. [4] investigate the possibility of provide open broadband wireless access using fixed broadband access lines (privately owned). They concluded that it is feasible to provide outdoor coverage based on the concept of sharing households' Internet access. Solariski et al. [8] evaluated the performance of the IEEE 802.11 technology in urban environments, and analyzed how it is affected by typical conditions such as inter-floor connectivity inside buildings, vertical height of the router's location, and indoor router's location. Bharat Rao et. al. [5] analyzed possible business models when creating wireless community based networks.

From the indoor point of view, Amaldi et al. [2] tackled the problem of appropriate positioning of indoor WiFi access points in order to achieve network effectiveness. The authors describe a similar approach to the one presented in this work. However, in [2] the hotspots can be installed after planning for the optimum coverage, whereas in the present scenario the hotspots need to be located in places provided with a broadband connection. Moreover, the solution presented in this work considers various constraints like budget and demand, while in [2] they focus on achieving optimum network capacity by reducing interference.

There are other tools available in the market with planning and optimization capabilities to generate successful wireless deployments. For example, special applications focus on WiFi deployments. This is the case of RF3D WifiPlannerTM provided by Psiber Data Systems Inc. [14]. This application is able to create WLAN plans in complex building environments trying to maximize network efficiency. Another type of applications are specially designed for cellular networks. This is the case of the set of tools provided by Equilateral [11]. This set consist of a suite of software tools for wireless network engineering that optimize the network efficiently with minimal capital expenditure in infrastructure. A third group of applications focus on indoor and outdoor coverage. The application AirMagnet Survey PRO [10] is an example of the aforementioned. To predict the outdoor coverage, this tool integrates with different visualization applications including Google Earth. This characteristic reduces the need of images such as city or country maps. A fourth group of applications are designed for the visualization of citywide deployments. These applications are used to show the current status of the deployment of a region.

Even though the above applications are similar to the present tool, several differences can be mentioned. In first place, MetroSim is constrained by home-user locations providing broadband infrastructure, thus, it is not possible to locate the routers in every possible point inside a map. Second, the number of candidate sites that have to be analyzed could scale considerably since the tool was built to analyze

citywide deployments. Third, MetroSim does not require the input of maps since the analysis is done by using geographical coordinates exported into files that can be visualized with a different application. Finally, MetroSim has a particular objective. It considers constraints such as budgeted or a particular number of deployments to fulfill an expected demand of a region while minimizing the cost of the whole deployment.

6. CONCLUSION AND FUTURE WORK

This paper described MetroSim, a user-friendly planning tool to assist operators on building citywide WiFi access networks. This work discusses the fundamental challenges for the deployment of metropolitan wireless networks and it addresses them in MetroSim. To evaluate this tool, several experiments were performed using full-anonymous data from a subset of Berlin's DSL infrastructure. The results show that MetroSim scales well in real scenarios and that it can generate low cost deployment options that are close to the optimum.

In addition, MetroSim provides a reporting tool that includes information about the remaining demand, suggested locations, used and unused locations, demand covered and cost of the deployment, as all this information is relevant to the operator. In terms of visualization, MetroSim generates the results in a flexible format allowing its presentation using external visualization tools such as Google Earth, which was used as an external application. The results are shown in real maps where the demand can be evaluated in a three-dimensional view at particular locations (see Figure 6).

An interesting aspect of the tool is its modularity. MetroSim has separate modules for visualization, reporting and modeling allowing the replacement of the modeling algorithm. Different methods may be plug into MetroSim and the generate results can be reported and visualized in an efficient way. This enables operators to play with parameters such as granularity, efficiency and cost, using MetroSim algorithm or any other available solution to this deployment problem [1].

7. REFERENCES

[1] L. E. Agustín-Blas, S. Salcedo-Sanz, P. Vidales, G. Urueta, A. Portilla-Figueras, and M. Solarski. A Hybrid Grouping Genetic Algorithm for Citywide Ubiquitous WiFi Access Deployment. In *Proceedings*

of the IEEE Congress on Evolutionary Computation (submitted), May 2009.

- [2] E. Amaldi, A. Capone, M. Cesana, F. Malucelli, and F. Palazzo. WLAN Coverage Planning: Optimization Models and Algorithms. In *Proceedings of the IEEE 59th Vehicular Technology Conference (VTC 2004)*, May 2004.
- [3] D. Garlan and M. Shaw. An introduction to software architecture. Technical report, Carnegie Mellon University, Pittsburgh, PA, USA, 1994.
- [4] J. Hakegard, B. Myhre, P. Lehne, T. Ormhaug, V. Bjugan, M. Mondin, M. Elkotob, and F. Steuer. Scenarios and Wireless Performance and Coverage. Technical report, OBAN Project deliverable, March 2005.
- [5] B. Rao and M. Parikh. Wireless Broadband Drivers and their Social implications. *Technology in Society*, 25:477–489(13), November 2003.
- [6] Reuters. Paris Wants Wireless Internet Access Across City, July 2006. <http://www.freepress.net/news/16387>.
- [7] N. Sastry, J. Crowcroft, and K. Sollins. Architecting Citywide Ubiquitous Wi-Fi Access. In *Proceedings of the 6th Workshop on Hot Topics in Networks (HotNets-VI)*, November 2007.
- [8] M. Solarski, P. Vidales, O. Schneider, P. Zerfos, and J. P. Singh. An Experimental Evaluation of Urban Networking Using IEEE 802.11 Technology. In *Proceedings of 1st IEEE Workshop on Operator-Assisted (Wireless Mesh) Community Networks (OpComm 2006)*, September 2006.
- [9] C. Thraves, G. Urueta, P. Vidales, and M. Solarski. Driving the Deployment of Citywide Ubiquitous Wifi Access. In *Proceedings of First International Conference on Simulation Tools and Techniques for Communications, Networks and Systems (SIMULTOOLS 2008) industry track (SIMULWORKS 2008)*, March 2008.
- [10] www.airmagnet.com.
- [11] www.equilateral.com.
- [12] www.fon.com.
- [13] www.websiteoptimization.com/bw/0810/.
- [14] www.psiber.com.
- [15] www.sharefi.com.