MITOS: A Smart Spaces System for Pervasive Computing

George Alyfantis, Stathes Hadjiefthymiades, and Lazaros Merakos

University of Athens, Department of Informatics and Telecommunications,
Communication Networks Laboratory, Panepistimioupolis, Ilisia,
Athens 15784, Greece

Abstract. The popularity of wireless networks has increased in the recent years, as they become a common addition to fixed LAN infrastructures. In this paper we introduce a novel use for a wireless network based on the IEEE 802.11 standard: a smart spaces system that proposes to wireless users to assume optimum locations in order to obtain a satisfactory QoS level when the wireless hot-spot becomes saturated. The system, being kept up to date with the traffic across the network and the location of each user, is capable of making the appropriate proposal urging users to move to new locations at reasonable distances. In that way, user perceived QoS and wireless network load-balancing can be both achieved. MITOS is the name of the developed smart spaces system. We discuss the general system architecture and operation as well as the optimum proposal algorithm. The operation of the system was verified in the wireless infrastructure of our department.

1 Introduction

According to M. Weiser [13], the father of ubiquitous computing, also called pervasive computing [5, 7, 10, 11], computers must disappear from conscious thought, in a way that they are integrated in the every day life. Over the recent years, we have experienced great penetration of wireless LANs. Especially, IEEE 802.11 LANs are met everywhere: in University departments, corporate facilities, factories, even at airports and public places. We can safely state that wireless LANs constitute a ubiquitous technology.

In such environments, where the population of wireless users is dense, it is possible that users experience poor wireless connectivity. This is reasonable if we consider a situation where a lot of users are gathered in an area covered by an access point (AP), and many of them are downloading large files. As a consequence, the wireless bandwidth is exhausted in that area, because the local AP gets congested. Hence, users harm one another and everybody experiences significant network delays resulting to a rather annoying situation. At the same time, there may be other APs in the building not congested at all, as they are installed at places where few users are usually present.

Thus, we can see that traffic among the APs in a building may not be evenly balanced. That is to say, the users do not take advantage of the overall wireless bandwidth; in contrary they use (or better abuse) certain portions of it. In that context it would be very interesting if there were a system that in first place could load-balance the traffic across the wireless network, and finally help users improve their wireless connection experience.

W. Lindner et al. (Eds.): EDBT 2004 Workshops, LNCS 3268, pp. 375–384, 2004.

Given that the position of an AP and the area of its coverage are well defined after installation, there is no other way to succeed load-balancing than to balance the user population across the building. In other words, in case congestion occurs, users have to move to other locations to maintain a satisfactory QoS level.

1.1 Motivation

The motivation of our work originates at a hypothetical scenario presented by M. Satyanarayanan in [11]. According to this scenario, a lady is waiting for her connecting flight at the airport and while waiting she is editing several large documents. While she is waiting for her boarding time she wants to send them via e-mail using the wireless network of the airport. Unfortunately, there are many other users surfing the Web at her gate and the adjacent gate. Thus, her wireless connectivity is poor and she will not have adequate time to send all her messages until she boards the plane. The pervasive computing system of the airport observes the situation and gathers information from various sources. It discovers that only a few gates away, 3 minutes on foot, bandwidth availability is excellent. The system also knows there will be no departures or arrivals for half an hour. Hence, it consults the lady to get there, where she will manage to send her documents before her boarding call. So does the lady and she successfully sends all her messages and gets back in time for her departure.

2 The MITOS Architecture

MITOS is a Smart Spaces System (SSS, S3) developed after the scenario discussed above, but has a much wider scope. Specifically, its purpose is to balance the traffic load across a wireless network within a certain building. The S3, by observing the traffic distribution in the wireless network, is capable of discovering whether congestion takes place in a certain segment of the network. At the same time, it is kept up to date with the current location of each mobile user and the AP he is attached to. If congestion occurs, the S3 locates each affected user and urges him to move to another location, where the S3 has observed that bandwidth reserves are higher. The S3 sends also instructions necessary for the transition from the current to the proposed location.

2.1 System Naming

The word “mitos” has its origins in the ancient myth of Theseus, the son of King Aegeas (after whom the Aegean Sea is named) who went into the den of the half-man, half-bull beast, the minotaur, to kill it. Previous attempts had failed because of the maze in which the minotaur lived. Theseus girlfriend, Ariadne, had the clever idea of giving Theseus a silken thread (“mitos”, in Greek) that he could unwind and use to find his way out of the maze. Since the purpose of the *MITOS* S3 is to find ways in which users with poor connectivity can lead themselves in other locations with better bandwidth, the *mitos* of Ariadne is a very representative symbol for that objective.

2.2 Architecture and Technologies

The MITOS S3 was designed to maintain continuous knowledge of the location of each mobile user and to be aware of the traffic load for every AP in the infrastructure. Moreover, when congestion is observed, an alerting message must appear to the user presenting the S3's Relocation Proposal (RP).

In order for these requirements to be fulfilled, the adopted architecture is composed of three basic entities (Fig.1):

- *Location Manager* - this component is in charge of user management and maintains relevant information (user identification data, current location, handling AP, proposed location and navigation information). Moreover, it manages wireless network traffic information and is responsible for advising users in the case of congestion. All information is maintained in a properly structured database.
- *SNMP Manager* - this component is in charge of gathering data from the APs, in order to compute load metrics, which allow detection of possible congestion. In case that one or more APs are found congested, SNMP Manager sends a full report to the Location Manager to take all the necessary steps.
- *Terminal Client* - this component is installed on the user's machine and actually comprises the interface between the user and the S3. It is constantly connected with the Location Manager to communicate location and connection information. Conversely, the Location Manager transmits messages asynchronously, in order to update the Terminal Client on the S3's RPs. The latter has to present that information to the user in a friendly way.

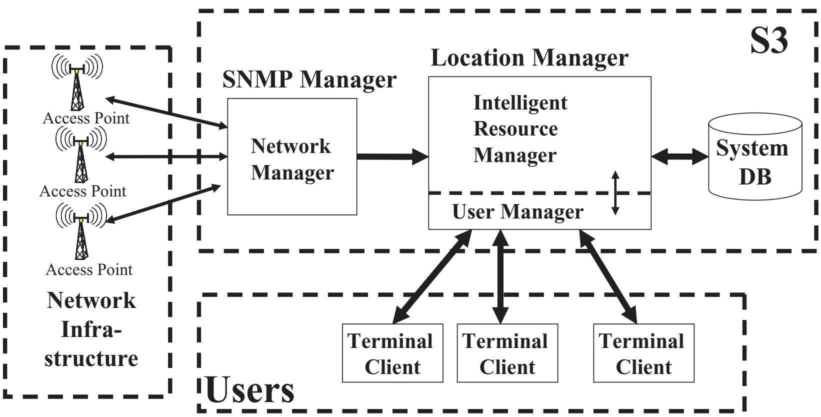

Fig. 1. MITOS S3 architecture and entity-level communication

The Terminal Client connects to the Location Manager through TCP. For the connection between the SNMP Manager and the Location Manager, we have also adopted TCP, as it is connection oriented, and the exchanged traffic reports are of great significance to the S3. Finally, the SNMP Manager, using the SNMP protocol (over UDP) queries the APs for the MIB (Management Information Base) variables necessary for traffic metric estimation. In order to implement the architecture presented above, we have adopted the following technological basis:

- *IEEE 802.11* - the IEEE standard for wireless local area networks [6]
- *Simple Network Management Protocol (SNMP)* - is defined in [2] and is the most prevalent management protocol for IP networks [12]
- *Nibble* - it is an indoor location system developed in UCLA [3, 1]. Its utility to the S3 is obvious, since the location of each user must be always known [8]
- *Java* - it is the “glue” that links together the aforementioned technologies

2.3 System Operation

The Location Manager is responsible for processing data coming from different system layers and resides on a node with increased computing power. All the information is kept in the S3’s database and can be distinguished into:

- User information (e.g., current location, handling AP, relocation proposal and navigation information),
- Network traffic information, and
- Network-building information (e.g., different building locations and topological relationship between them, and locations each AP covers)

The information processed by the Location Manager is provided partly by the Terminal Client and partly by the SNMP Manager. The SNMP Manager sends to the Location Manager detailed traffic reports, in the case of congestion. Subsequently, the Location Manager, updates the database with the recently received report, and takes appropriate actions (described below). The Terminal Client regularly sends signal quality data so that the Location Manager can estimate the current location of the mobile device as well as the AP it is connected to.

2.4 Nibble Indoor Positioning

Nibble is a Java-based system capable of recognizing locations (rooms, corridors, etc) by their signal quality pattern obtained by the WLAN card. It maintains a properly trained XML-formatted Bayesian network and returns symbolic information [4] . Nibble is a stand-alone application [9] and adopts a decentralized architecture. However, for scalability, network change resilience and client flexibility, we enhanced Nibble so that it runs under the control of the Location Manager and estimates location on behalf of the client. Thus, it can be guaranteed that the Bayesian network will be always up to date and the Terminal Client application is lighter and more practical for a variety of mobile devices running Java.

2.5 Terminal Client User Interface

In case the S3 decides that a user should move to a certain location, in order to keep a satisfactory wireless QoS level, the Location Manager sends to the corresponding Terminal Client an appropriate message. Various messages are exchanged between the entities of the S3 by means of an application level protocol devised for this reason.

As soon as the Terminal Client receives this message, an alert indication is provided to the user urging him to move to the proposed location. In Fig. 2 and Fig. 3, we present a prototype implementation of the Graphical User Interface (GUI) provided through the Terminal Client. Such GUI was not designed according to the distraction-free perspective of pervasive computing but assisted us in validating the functionality of the S3.

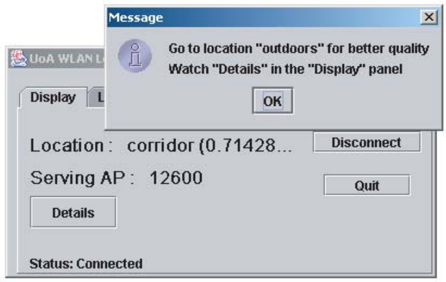

Fig. 2. Message box displaying the MITOS RP to the user

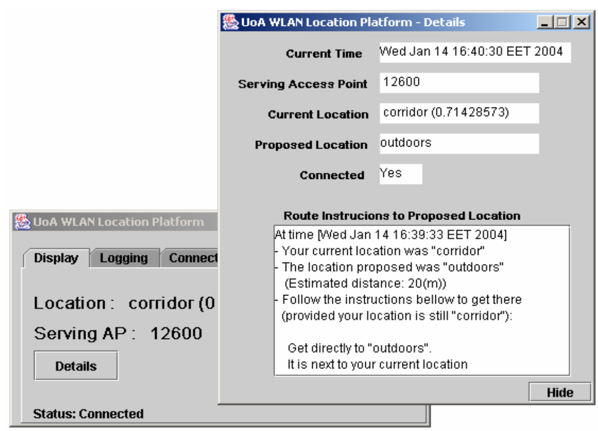

Fig. 3. Instructions for the transition from the current to the proposed location

2.6 Traffic Load Measurements

At the current implementation, the SNMP Manager computes the utilization factor of the wireless interfaces. The load threshold value beyond which we consider that the AP becomes saturated is set to 90, which corresponds to 90% link utilization. We have adopted this value in order to minimize the RP transmission frequency and the user distraction caused, keeping though the desired properties, as will be shown bellow.

The SNMP Manager updates every 500ms (period T) the load metric value (M_k) for every wireless interface k. As a result, the metric values exhibit great fluctuation, since

IP networks are typically subject to bursty traffic. For that reason, we apply a low pass filter in the metric values of each wireless interface taking the average over a number (W) of successive samples in a time window $W \times T$). The load value, L_t^k, of AP k at time t, is calculated as follows:

$$L_t^k = \frac{1}{W} \sum_{i=0}^{W-1} M_k(t - T \cdot i) \tag{1}$$

$M_k(t)$ denotes the metric value for wireless interface k at time t. It is obvious that there is a trade-off between larger and smaller values of W. A small value of W would render the S3 very sensitive to traffic fluctuations, and rather annoying to users who would receive RPs very frequently. On the other hand, if W assumed large values, the S3 would exhibit reduced responsivity in congestion situations. In our system, W assumed the value of 16. This value was selected after a series of experiments and was found capable of smoothing transient traffic load picks (attributed to HTTP requests), while maintaining the responsivity of the system to congestion occurrences.

If, at some time, the SNMP Manager finds the load of one or more interfaces exceeding the aforementioned threshold value, we consider that congestion has occurred. In that case, the SNMP Manager sends a report to the Location Manager indicating the situation.

As soon as the Location Manager receives that report, it knows exactly the traffic distribution among the wireless interfaces in the network. Then, it is responsible of taking action, exploiting the traffic information received, along with the user location information already acquired.

2.7 Location Representation

For the location representation in the S3 we used a graph-based model (Location Graph, LG). The locations identified by Nibble are mapped to LG nodes, and arcs denote the topological relation between such locations. Weights assigned to arcs denote the physical distance or cost for the transition from one location to another. Shortest paths (useful for navigation instructions) are calculated using the Dijkstra algorithm.

2.8 Optimum Location Proposal

In case of congestion, the objective of the S3 is to decongest the most loaded APs and distribute traffic (i.e. users) to neighboring APs. For the determination of the optimum location of each user, two are the factors that must be considered to issue a safe RP to the user: traffic load and distance. For each relocation ($A \rightarrow B$) considered by the system, we have adopted a cost function as follows:

$$\begin{aligned} &\text{Cost}(A \rightarrow B) \\ &= \begin{cases} a \cdot \frac{100 \cdot \text{distance}(A,B)}{\text{maxDistance}} + (1-a) \cdot \text{load}(B), & \text{distance}(A,B) \leq \text{maxDistance} \\ \infty, & \text{otherwise} \end{cases} \\ &\quad a = 0.6, \quad \text{maxDistance} = 200 \text{ m} \end{aligned} \tag{2}$$

The system suggests to the user the location (B) with the lowest calculated cost value. Locations that are more than 200m apart are excluded from the feasible set. Distance is measured in meters, while load is measured in a percent scale, from 0 to 100. The weight factor, a, assumed the value of 0.6 that was arbitrarily selected. Alternatively, the weight factor could be user-specific, contained in user profiles, indicating preference for bandwidth abundance or geographically restricted relocations. The weight factor, a, takes value from the interval [0.05. . . 0.95] in order to always take into account both factors.

The issued RP, apart from the name of the proposed location, contains relevant navigation details, derived from the pre-computed shortest paths. Users can follow the RPs and take advantage of the extra bandwidth available. The S3 sends RPs, indiscriminately, to all users connected to congested APs. However, if all affected users followed the advice of the system it is evident that load balancing across the wireless network would fail.

3 MITOS Assessment

To prove the aforementioned anticipation we performed a series of simulations. We assumed a complex floor layout with fifty locations (corridors, offices and an atrium), 10 APs covering these locations and 100 mobile users. Each AP covers an area of 5 locations. The relocation of users between locations is performed stochastically according to a state transition matrix, resulting in an environment of high mobility. In the building there are areas frequently visited and others that are not popular (defined stochastically).

Table 1. Application Characteristics

Application Type	Session activation probability	Bandwidth requirements (Kbps)	Session duration (sec)
WWW	0.4	500	5
Voice over IP	0.025	64	180
Video	0.005	1000	240

We considered three different types of applications with specific bandwidth and duration characteristics. Each application is invoked independently, according to the probabilities shown in Table 1.

We assumed that all users have the same application activation probabilities, as shown in Table 1, hence, their bandwidth requirements are similar. A RP is adopted stochastically by the receiving user, with probability P. It is also assumed that if a user adopts the RP, he follows the shortest path indicated to him. Furthermore, as soon as he reaches the proposed location, he avoids a new relocation for a certain period of time.

3.1 Simulation Metrics

To assess the MITOS system we defined the following metric:

$$\overline{D}(t) = \frac{1}{N}\sum_{i=1}^{N} D_i(t), \qquad t \in \{1, 2, \ldots, C\} \tag{3}$$

C denotes the simulation time (i.e., the total number of simulation cycles, namely 2000 - cycle duration=5sec), N is the number of APs (namely 10), and $D_i(t)$ the instantaneous bandwidth demand on AP_i. $\overline{D}(t)$ is the instantaneous, average bandwidth demand for all APs (i.e. the desired balanced bandwidth demand). We have also introduced $dev_i(t)$, as shown in (4), to denote the instantaneous deviation of the bandwidth demand on AP_i from the desired balanced demand, $\overline{D}(t)$.

$$\mathrm{dev}_i(t) = \left[D_i(t) - \overline{D}(t)\right]^2, \qquad i \in \{1, 2, \dots, N\} \text{ and } t \in \{1, 2, \dots, C\} \tag{4}$$

$dev(t)$, as shown in (5), is the total instantaneous deviation normalized over the desired bandwidth demand, $\overline{D}(t)$.

$$\mathrm{dev}(t) = \frac{1}{[\overline{D}(t)]^2} \sum_{i=1}^{N} \mathrm{dev}_i(t), \qquad t \in \{1, 2, \dots, C\} \tag{5}$$

Our assessment was based on the metric given in (6). Dev is the time average of the deviation given in (5) over the entire simulation duration.

$$\mathrm{Dev} = \frac{1}{C} \sum_{i=1}^{C} \mathrm{dev}(t) \tag{6}$$

3.2 Simulation Results

In Fig. 4 we can see the Dev metric, as the RP acceptance probability (Pa) increases in the range [0.05. . . 1.0] for the MITOS architecture. As the probability of RP acceptance increases over 0.90, MITOS exhibits a very large deviation, higher than that of the RP-free scheme. This behavior renders MITOS inappropriate for use in a real situation.

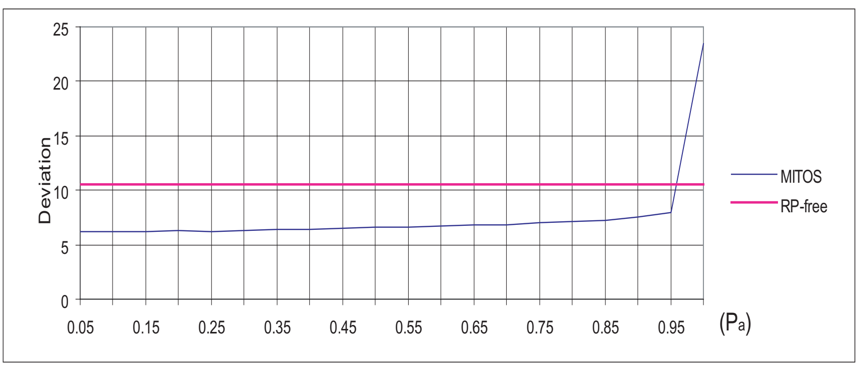

Fig. 4. MITOS simulation results as the probability of RP acceptance increases

To solve the problem of similar user treatment we decided to assign a bias term to each user, assuming random values in the range [-1.0. . . 1.0]. The weight factor, a, in (2) is then modified as shown in (7).

$$a'(\text{bias}) = \begin{cases} 0.95, & a + \text{bias} > 0.95 \\ 0.05, & a + \text{bias} \leq 0.05 \\ a + \text{bias}, & \text{otherwise} \end{cases} \tag{7}$$

With this MITOS enhancement, it was found that the problem mentioned earlier was resolved. The simulation results for the enhanced scheme are shown in Fig. 5. We performed simulations for various load thresholds, namely 10%, 50% and 90%. The resulting trend-lines (fourth degree polynomial regression) represent the trend of traffic load deviation for the different thresholds.

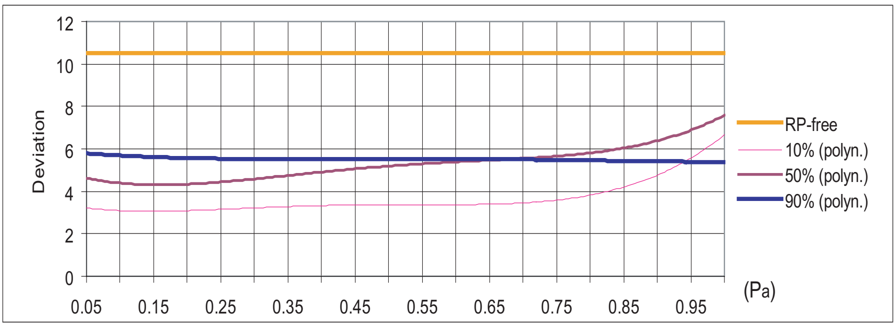

Fig. 5. Enhanced MITOS simulation results

From Fig. 5 we observe that the enhanced MITOS system exhibits significantly better behavior than the RP-free scheme under circumstances.

4 Conclusions

The implemented Smart Spaces System proposes to the users of a wireless network infrastructure, ways to improve their connectivity, being kept up to date with the traffic distribution among the wireless APs, the current location of each user and the AP he is attached to. In case of congestion, affected users are urged to move to a new location in the broader area, if they are willing to keep a satisfactory QoS level. Each RP is accompanied by navigation instructions for the transition from the current to the proposed location. The interest of the implementation lies in the fact that the system is capable of gathering and combining information from diverse layers (low-level congestion data, current location data, distance information, etc.) by using different technologies, according to the objectives of the pervasive computing paradigm. Furthermore, the problem of common user reaction to RPs was addressed.

In the future, we plan to extend MITOS capabilities to integrate game theoretic models or economic models as an overlay component to the S3 architecture, in order to further improve its performance.

References

1. The Nibble location system, May 2001. `http://mmsl.cs.ucla.edu/nibble`.
2. J. Case, M. Fedor, M. Schoffstall, and J. Davin. A simple network management protocol (snmp), ietf rfc 1157. *Networking Working Group*, 1990.
3. P. Castro, P. Chiu, T. Kremenek, and R. Muntz. A probabilistic room location service for wireless networked environments. In *Proc. of Ubicomp 2001*, 2001.
4. D. Fox, J. Hightower, L. Liao, D. Schulz, and G. Borriello. Bayesian filtering for location estimation. *IEEE Pervasive Computing*, July 2003.
5. D. Garlan, D. Siewiorek, A. Smailagic, and P. Steenkiste. Project aura: Toward distraction-free pervasive computing. *IEEE Pervasive Computing*, April 2002.
6. M. Gast. *802.11 Wireless Networks: The Definite Guide*. O'Reilly, April 2002.
7. U. Hansmann, L. Merk, M. Nicklous, and T. Stober. *Pervasive Computing Handbook. The Mobile World (Second Edition)*. Springer, 2003.
8. J. Hightower and G. Borriello. Location systems for ubiquitous computing. *IEEE Computer*, August 2001.
9. G. Papazafeiropoulos, N. Prigouris, G. Marias, S. Hadjiefthymiades, and L. Merakos. Retrieving position from indoor wlans through gwlc. In *Proc. of the IST Mobile & Wireless Communications Summit*, 2003.
10. D. Saha and A. Mukherjee. Pervasive computing: A paradigm for the 21st century. *IEEE Pervasive Computing*, March 2003.
11. M. Satyanarayanan. Pervasive computing: Vision and challenges. *IEEE Pervasive Computing*, August 2001.
12. M. Sidnie and D. Feit. *SNMP: A Guide to Network Management*. McGraw-Hill Professional, 1993.
13. M. Weiser. The computer for the twenty-first century. *Scientific American*, September 1991.