POWER MANAGEMENT

G. Anastasi
Department of Information Engineering, University of Pisa, Italy

M. Conti, E. Gregori, and **A. Passarella**
CNR-IIT Institute, Italy

Keywords: Energy Harvesting, Power Management, Reactive Policies, Proactive Policies, Mixed Policies, WiFi Hotspot, IEEE 802.11 PSM, Pervasive Systems, Ad Hoc Networks, Sensor Networks

Contents

Summary

The proliferation of mobile devices (laptops, PDAs, smart phones) and the development of wireless technologies are producing a revolutionary change in our information society. Wireless Internet access and mobile computing are now developing as the natural complement and evolution of the Internet success story. Despite this trend, many technical problems have still to be faced for this scenario to become an established reality. Power management is one of the most critical issues as the energy available in a mobile device is probably the most critical resource. In principle, either increasing the battery capacity, or reducing the power consumption, could alleviate energy-related problems. However, projections on progresses in battery technologies show that only small improvements in the battery capacity are expected in next future. Harvesting energy from the surrounding environment is a very interesting and promising direction, but currently scavenging just provides limited amounts of energy. Therefore, it is vital to manage energy available at mobile devices very efficiently. Among the components of a mobile device that contribute to power consumption (CPU, video, hard-disk, network interface, etc), the impact of the wireless network interface becomes more and more relevant as the device size decreases. It is thus extremely important to design energy-efficient networking protocols and applications. In this contribution we provide an up-to-date state of the art of power management techniques proposed for mobile and pervasive computing environments with special emphasis to techniques aimed at reducing the energy consumed by the wireless interface.

1. Introduction

Wireless Internet access and mobile computing are establishing as the natural complement and evolution of the Internet success story. Nowadays is already common experience to access the Internet wirelessly via Wi-Fi enabled laptops and PDAs, or through cellular phones. Projections show that in few years the number of mobile connections and the number of shipments of mobile terminals will grow yet by another 20-50 percent. With this trend, we can expect the total number of mobile Internet users soon to exceed that of the fixed-line Internet users. The proliferation of mobile devices is producing a revolutionary change in our information society. Laptops, smart-phones and PDAs, equipped with wireless technologies, support users in accomplishing their tasks, accessing information, or communicating with other users anytime, anywhere.

Despite this trend, many technical problems have still to be fixed for this scenario to become an established reality. Power management is one of the most critical issues as the energy available in a mobile device is probably the most critical resource. In principle, either increasing the battery capacity, or reducing the power consumption, could alleviate energy-related problems. However, projections on progresses in battery technologies show that only small improvements in the battery capacity are expected in next future. Harvesting energy from the surrounding environment is a very interesting and promising direction. In addition, harvested energy is renewable and pollution free. However, only in the very long term we can expect to be able to scavenge enough energy from the environment so as to forget about batteries. If the battery capacity cannot be improved significantly, and scavenging just provides limited amounts of energy, it is vital to manage power utilization efficiently, by identifying ways to use less energy preferably with no impact on the applications.

In a nutshell, power management for mobile devices is a must, and each (hardware or software) component of a mobile device should be designed to be energy efficient. This contribution provides an up-to-date state of the art of power management techniques for mobile and pervasive computing environments, and highlights some open issues. Section 2 discusses battery technologies and energy harvesting techniques. Section 3 presents a general framework (and the related strategies) for power management in a mobile device. These strategies can operate at different layers of the system architecture including hardware, operating system, networking protocols, and applications. At the operating system level techniques for hard-disk management, CPU scheduling and screen blanking have been proposed. Techniques implemented at the application level include disconnected operations, remote task execution, agent-based computing, and exploitation of the application semantic. Techniques for adapting the application behavior to the changing level of energy also fall in this category.

Among the components of a mobile device that contribute to power consumption (CPU, video, hard-disk, network interface, etc), the impact of the network interface becomes more and more relevant as the device size decreases. In a laptop the percentage of energy drained by the wireless interface is about 10% of the overall system consumption. This percentage grows up to 50% when we consider small-size devices like PDAs. This difference can be justified if we consider that small-size mobile computers frequently have no hard disk and limited computational resources. On the other hand, the wireless interface provides almost the same functionalities as in a laptop or desktop PC. It is thus extremely important to design power-efficient network protocols and applications. In Section 4 we survey the most relevant power management policies for reducing the power consumption of the wireless interface in an infrastructure-based scenario, i.e., when the mobile device accesses the network infrastructure (e.g., the Internet) through an access point. Session 0 we also consider power management in other wireless scenarios (i.e., ad hoc and sensor networks).

2. Storing and Harvesting Energy

The spectrum of mobile computing devices is quite large. They range from laptops to sensor nodes that have a size of few tens of mm3. Hereafter, we divide mobile devices in the following three general classes.

- high-end devices (e.g., laptops)
- medium-size devices (e.g., PDAs and smart phones)
- small-size devices (e.g., sensor nodes)

Power requirements highly depend on the device type, and on the task the device is designed for. For example, laptops require rechargeable, high-capacity batteries, but dimension and weight are not primary concerns. On the other hand, sensor nodes are typically not reachable after the sensor network is deployed (e.g., they may monitor radioactive or polluted zones, they may be put below buildings to monitor seismic waves, etc.). Therefore, unless energy-harvesting techniques are implemented, sensor nodes are designed to work unattended until the battery is completely exhausted. Thus, batteries for sensor nodes are typically not rechargeable, and must be very small. To summarize, due to the high variety of devices, the spectrum of battery technologies used in mobile devices is wide.

Today, the most widely used technology is Lithium-Ion Cells. These batteries are available in many different form factors. Moreover, they can be restored at nearly full capacity for many recharge cycles. At the state of the art, lithium batteries are used as an off-the-shelf component, and can be found in devices of almost any class.

Device	**Capacity (mAh)**	**Volume cm^3**
Laptop	~ 4000 mAh	~200 cm^3
PDA	~ 900 mAh	~50 cm^3
Sensor node	~500 mAh	~10 cm^3

Table 1: Typical characteristics of lithium batteries used for different device types

The trade-off between capacity and size guarantees a reasonable lifetime to devices such as laptops, PDAs or smart phones. However, for wearable computers and sensor nodes, form factors of commercial batteries might not be sufficient. Current research efforts are devoted to miniaturize sensor nodes and wearable devices as much as possible. For example, the SmartDust project of the University of California at Berkeley has developed prototype sensor nodes as small as few tens of mm3. In this scenario, borderline technologies are currently investigated, in order to provide sufficient energy in so small form factors. Also, solutions to scavenge energy from the environment have been investigated. Harvesting techniques will be discussed in detail in Section 2.1.

Despite the large number of research activities in this field, researchers agree that battery capacities are today one of the main limiting factors to the development of the mobile computing paradigm. More important, in the near future the difference between battery capacities and energetic requirements of mobile devices is expected to become even deeper. Figure 1 shows the performance improvements for different components of a mobile device in the time period 1990-2000 (improvements are expressed as multiples of the performance in the year 1990). This trend is confirmed by more recent studies that refer to the time interval 1990-2003. The energy density in batteries has registered the lower performance increase in the last decade.

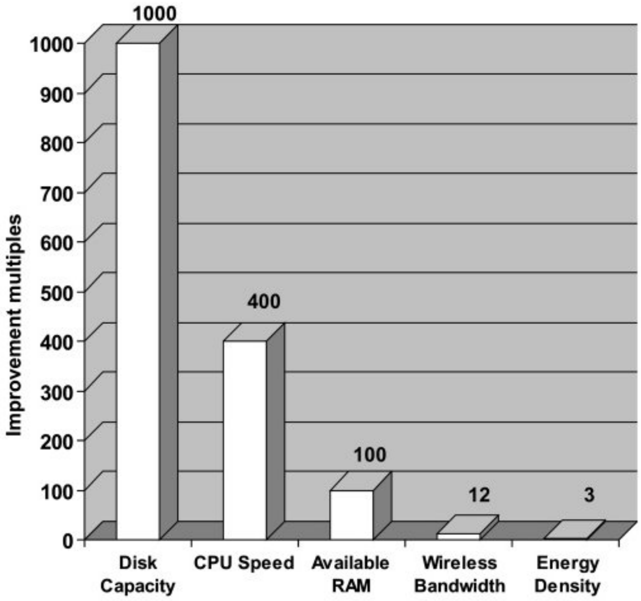

Figure 1: Increase of components' capabilities from 1990 to 2000. This plot has been derived from results in Starner (2003)

Figure 1 suggests that power management is a key enabling factor for the development of mobile computing. However, to design effective power management policies, batteries' properties like *Rate Capacity Effect* and *Relaxation Effect* need be taken into account. The energy that a battery can provide depends not only on its capacity, but also on the profile of the current it supplies (throughout referred to as I_d). If I_d is constant and greater than a certain threshold (referred to as the *rated current* of the battery) the energetic efficiency can be dramatically low. Specifically, it is well known that the voltage of a battery drops down while it is supplying current. Eventually, a *cut-off voltage* is reached, after which the battery is no longer able to provide energy. It has been shown that, if I_d is constant over time, the more I_d grows beyond the rated current, the less energy is provided before reaching the cut-off voltage (i.e., the lower is the energetic efficiency). This phenomenon is known as the *rate capacity effect*. To overcome this problem, many authors propose to drain current in *pulsed mode*. This allows us to achieve a higher energetic efficiency. In practice, a constant current I_d is drained for a time interval t_{load}. Then, the battery is left idle for a time interval t_{idle}, and so on. During t_{idle} the battery is able to recover the voltage lost during the previous t_{load}. This phenomenon is known as the *relaxation effect*. By exploiting this effect, if a pulsed scheme is used, the energy provided before reaching the cut-off voltage is higher than in the case of continuous drain. Obviously, the performance improvements depend on the battery capacity and the ratio between t_{load} and t_{idle}.

2.1. Harvesting Energy from the Environment

The idea of scavenging energy from the environment to feed electronic devices is not new. For example, electronic calculators powered by light sources are sold since long time ago. The new challenge is how to harvest enough energy to sustain the operation of mobile devices. Investigating this direction is very important, for several reasons. Firstly, energy harvested from the environment is pollution free. Secondly, being renewable, it potentially allows devices to run unattended for virtually unlimited time.

Energy harvesting for mobile computing is still in early stages, and is gaining momentum in the research community. A first research direction is collecting energy from electromagnetic fields. The most popular and developed example is getting energy from light sources via solar cells. It is already possible to embed solar cells into thin plastic sheets that, for example, can be laminated onto laptops cases. Unfortunately, current technology allows conversion efficiency just between 10% and 30%, thus requiring too large surfaces to produce reasonable amounts of energy. Should conversion efficiency sufficiently improve, in many cases this technology could replace batteries for devices such as laptops. It is also possible to harvest energy from RF signals. Actually, this is the way passive RF tags work. This approach can be extended to more complex devices, as well. For example, researchers are trying to feed sensor nodes through the RF signal sent by a reader. While the physical principles is exactly the same as in the RF tags, the power required for feeding a whole sensor device is quite higher, making such a system a challenging one.

Other possible sources of energy harvesting are thermal gradients. The Carnot cycle is the physical principle behind this approach. For example, the Seiko Thermic wristwatch exploits the thermal gradient between the human body and the environment. Also in this case, the conversion efficiency is the main problem, especially when the thermal gradient is small.

Radioactivity has also been proposed as a source of energy for small devices. The typical limited size of the radiating material avoids safety and health problems. This technology is particularly suitable for devices operating with very limited power (i.e., tens of μW) for very long times. Indeed, the limit in time of such a system is governed by the half-life of the radiating material, which can be in the order of hundreds of years.

Mechanical movements can be exploited to scavenge energy, as well. For example, vibrations in the environment can be converted through piezoelectric materials. Research in this field is already quite developed, so that it has been possible to feed an off-the-shelf Mica2Dot Mote operating at a 1% duty cycle just by means of such a system. Human movements can be also used to collect energy. Self-winding wristwatches date back a long ago, as they have been diffused since 1930s. More recently (1997), the same principle has been used to build windup radios to be used when battery availability is an issue. In January 2005, Nicholas Negroponte announced the “One Laptop per Child” project to build $100 laptops for education in emerging countries. These laptops can use windup systems to store energy. Finally, it has also been proposed to harvest energy by heel strikes during people walks. It has been proved that this approach can produce average power in the order of 250-700 mW, thus representing a very promising direction.

Even though, in the very long term, energy harvesting techniques might represent the main power source for mobile devices, in the meanwhile the conversion process is not efficient enough. Energy scavenging can thus be used just to power very simple devices (such as RFID), or as a complementary power source, e.g., to replenish a battery in the background. In general, the main issue seems not to be the amount of *energy* that can be collected through harvesting (which is virtually infinite), but the amount of *power*, which is quite limited. Therefore, energy harvesting and power management are two key principles around which mobile computing should be designed. Since i) battery capacities are not expected to keep the pace of electronics demand, and ii) power available by scavenging the environment is quite limited, judicious management of power resources will still be a must for a long time.

3. General Approaches to Power Management in Mobile Devices

A computer can be seen as a set of hardware and software components servicing user requests. Hardware components need energy to operate. Instead, software components do not consume power by themselves, but they impact on the power consumption as they control the activity of hardware components.

A large effort has been devoted to design and manage hardware components in an energy-efficient way. The most commonly used approach consists in focusing on a specific component (e.g., the hard disk, the network interface, ...), and deploying a *dedicated power manager* guaranteeing that the component provides the required services, while optimizing the energy spent in providing them. For example, laptop screens are usually blanked after a pre-specified inactivity interval, and resumed when a new activity is detected (e.g., a mouse movement or a key press). In other words, a power manager is “attached” to the screen, monitoring its activity, and switching it in a low-power mode when it is not used for a while. In principle, power managers could be implemented either in hardware or in software. However, due to their inherent flexibility, software managers are usually preferred. Obviously, when designing a power management mechanism we should take also care of the additional power consumption of the power manager itself.

Hereafter, we survey the main power management approaches presented in the literature. To select the most appropriate power manager it may be worthwhile to know how the different hardware components of the mobile device contribute to the overall power consumption. Therefore, in Section 3.1, we introduce the breakdowns of the power consumption in typical mobile devices.

3.1. Energy Characteristics of Mobile Devices

In the past years, several studies have been done to measure the power drained by various components of a laptop. In 1996 Udani and Smith showed that the CPU, wireless network interface, hard disk and display require approximately 21%, 18%, 18% and 36% of the total power, respectively. Other researchers presented later similar results. More recently, several works have confirmed the large impact on the overall system consumption due to the wireless interface, CPU, and hard disks.

The power breakdown of medium-size devices, such as PDAs, is quite different. Usually, such systems do not have hard disks, and use flash memories to store persistent data. Typically the wireless network interface consumes about 50% of the total power, while the display and the CPU contribute for about 30% and 10%, respectively.

Finally, sensor nodes drastically differ from both laptops and PDAs, and hence their power breakdown is completely different. In addition, the power consumption in sensor nodes may greatly depend on the specific task implemented by the sensor network. Typically, the wireless network interface and the sensing subsystem are the most power-hungry components of a sensor node. On the contrary, the CPU accounts for a minor percentage of the power consumption.

To summarize, several hardware components of a computing device need power management. In the past years, many researchers have focused mainly on hard disks and CPUs, as they account for a great portion of the total power consumption in laptops. More recently, increasing interest has been devoted to the networking subsystem, as it is the main source of power consumption for medium and small-size mobile devices (e.g., PDA). In the following of this section we survey power-management approaches that apply to different hardware subsystems. Then, in Section 4, power-management approaches tailored to the networking subsystem will be discussed in depth.

3.2. Framework for Power Management

Figure 2 shows a general scheme for power management. To better understand Figure 2, let us define the concept of power-manageable component (PMC). A PMC is a hardware component that provides a well-defined kind of service. For example, hard disks, network interfaces, CPUs can be seen as power-manageable components. A fundamental requirement of a PMC is the availability of different *operating modes*. Each mode is characterized by a specific power consumption and performance level. For example, a hard disk has high power consumption when disks are spinning, and data can be accessed very quickly. When disk heads are parked, the power consumption is much lower, but the access time becomes very high, since disks must be spun up again before accessing data.

In Figure 2 a power-manageable component is represented as a *Service Provider* (SP). Each item in the *Queue* (Q) is a request for an SP service. A *Service Requester* (SR) — representing, in an aggregate way, the SP users — issues requests through the Queue. The *Power Manager* (PM) *observes* the status of the Service Requester, Queue and Service Provider (that is, the status of the system), and issues *commands* to the Service Provider, dynamically switching it between its operating modes. The algorithm used by the Power Manager to decide the Service Provider operating modes is referred to as *policy*. The goal of a policy is to make the Service Provider handle items in the Queue by: i) spending the minimum possible energy, and ii) still providing an acceptable Quality of Service (QoS) to users. It must be pointed out that transitions between different SP modes usually have a cost. Specifically, each transition requires a time interval during which the Service Provider consumes energy but is not able to service any request. Furthermore, the lesser the power consumption of the low-power mode, the higher the time interval to switch to a high-performance mode. For each low-power mode *M*, a *break-even time* (T_{BE}^{M}) can be defined. When the Service Provider is switched to low-power mode *M*, it should not switch back to a high-power mode before

T_{BE}^{M} seconds. Otherwise, the energetic advantage of using the low-power mode is overwhelmed by the transition cost. Hence, the Service Provider can be modeled as a server with vacations. During vacation periods, the Service Provider operates in low-power modes and it is not able to service requests in the Queue. The vacation lengths depend on the particular mode the Service Provider is operating in, and on the policy implemented by the Power Manager. Specifically, for each power mode *M*, T_{BE}^{M} represents the minimum vacation length. The Power Manager may increase the vacation length, based on the observed system status. Therefore, if d^{M} denotes the vacation length related to the operating mode *M*, we can express d^{M} as follows:

$$d^{M} = T_{\mathrm{BE}}^{M} + f_d\left(policy\right) \qquad (1)$$

where $f_d\left(\bullet\right)$ denotes the delay introduced by the selected power management policy.

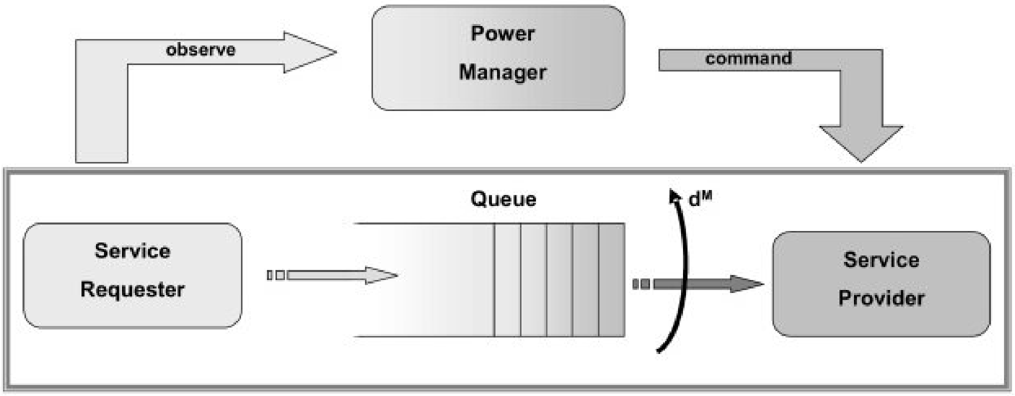

Figure 2: High-level system model. This figure has been derived from Benini (1999)

It is worth noting that a typical drawback of any power management policy is a negative impact on the QoS perceived by SP users. With reference to Eq.(1), energy saving increases with the increase of d^{M}. However, d^{M} also impacts on the delay introduced by power management to service times, thus decreasing the QoS. For example, when a hard disk is spun down, the user must wait for a long time at the first access. Therefore, the power management design must trade-off between these contrasting requirements.

The system represented in Figure 2 is completely isolated, i.e., the behavior of SR, SP and PM neither affects, nor is affected, by any other computer component. It is possible to enrich this model to achieve a more detailed characterization of power management for mobile devices. Specifically, the Power Manager could observe not only the status of the Service Requester, Queue and Service Provider, but also the status of other computer components (i.e., interactions between different computer components could be taken into consideration).

3.2.1. Possible Strategies

The core of any power management system is the behavior of Power Managers. Typically, two complementary approaches have been used: a *reactive* approach and a *proactive* approach.

With reference to Figure 2, the reactive approach focuses on dynamically driving the Service Provider in the various operating modes, based on the knowledge of the system status (i.e., the SP, SR and Q status). This approach is usually referred to as *Dynamic Power Management* (DPM). Solutions of this type range from simple timeout policies, to policies based on dynamically estimating the profile of requests issued by the Server Requester. These solutions do not modify the profile of SR requests (in order to optimize the power consumption of the Service Provider), but they dynamically react to the requests profile by selecting the appropriate mode of the Service Provider. This is why this approach is referred to as *reactive approach*. Reactive policies are surveyed in Section 3.3.

The proactive approach is aimed at modifying the workload issued to the Service Provider, i.e., it modifies the profile of SR requests. Policies based on this approach are aware of the impact that the workload shape has on the SP power consumption, and hence modify the workload to optimize the performance. For example, some compilers modify the execution order of instructions to reduce transitions between 0s and 1s on the buses' lines. Obviously, it must be assured that the Service Provider provides the same results after processing either the original or the modified workload.

As the reactive and proactive approaches are orthogonal, some solutions exploit a combination of them. Hereafter, we refer to these solutions as *mixed policies*. Mixed policies implement algorithms that dynamically select the Service Provider operating mode, based on the workload shape. If possible, they also modify the workload in such a way that low-power operating modes could be exploited more efficiently than with the original workload. Section 3.4 is devoted proactive and mixed policies.

-
-
-

Bibliography

Anastasi G. et al. (2005). Power Management in Mobile and Pervasive Computing Systems. *Algorithms and Protocols for Wireless and Mobile Networks* (ed. Azzedine Boukerche). CRC-Hall Publisher. [This is a comprehensive survey on power management techniques in mobile and pervasive systems].

Pervasive (2005). Energy Harvesting and Conservation. *IEEE Pervasive Computing* **4**.[This is a magazine issue completely devoted to energy harvesting and conservation techniques for portable devices].

Starner T. E. (2003). Powerful Change Part 1: Batteries and Possible Alternatives for the Mobile Market. *IEEE Pervasive Computing*, pp. 86-88. [This paper discusses possible alternatives to using batteries in power portable devices and computers].

Benini L. et al. (2000). A Survey of Design Techniques for System-Level Dynamic Power Management. *IEEE Transactions on VLSI Systems* **8** pp. 299-316. [This is a survey on system-level approaches to dynamic power management].

Benini L. et al. (1999). Dynamic Power Management of Electronic Systems. *System Level Synthesis*. [This paper presents techniques, models and industrial standards related to Dynamic Power Management]

Conti M. (2003). Body, Personal, and Local Wireless Ad Hoc Networks. *Handbook of Ad Hoc Networks* (ed. M. Ilyas Editor). CRC Press, New York. [This is a tutorial on Ad hoc, Personal and Body Area Networks].

Akyildiz I (2002). Wireless sensor networks: A survey. *Computer Network* **38**, 393 - 422. [This is a comprehensive survey on wireless sensor networks].

Krashinsky R et al. (2005). Minimizing Energy for Wireless Web Access with Bounded Slowdown. *ACM/Kluwer Wireless Networks* **11**, pp. 135-148. [This paper proposes the BSD protocol described in Section 4.1.3].

Kravets R. et al. (2000). Application-driven Power Management for Mobile Communication. *ACM/Baltzer Wireless Networks* **6**, pp.263-277. [This paper introduces the Communication-based Power Management Protocol described in Section 4.2.2].

Anand M. et al. (2005). Self-Tuning Wireless Network Power Management, *ACM/Kluwer Wireless Networks* **11**, pp. 451-469. [This paper introduces the STPM protocol described in Section 4.3.1].

802.11 (2006). http://grouper.ieee.org/grups/802/11/. [This is the web site of the IEEE 802.11 Working Group for WLAN standards].

802.15.4 (2006). http://www.ieee802.org/15/. [This is the web site of the IEEE 802.15 Working Group for WPAN]

Biographical Sketches

Giuseppe Anastasi is an associate professor of Computer Engineering at the Department of Information Engineering of the University of Pisa, Italy. He received the Laurea (cum laude) degree in Electrical Engineering, and the PhD degree in Computer Engineering, both from the University of Pisa, Italy, in 1990 and 1995, respectively. His research interests include mobile and pervasive computing, ad hoc and sensor networks, and power management. He is a co-editor of the book Advanced Lectures in Networking (LNCS 2497, Springer, 2002), and has published more than 50 papers in the area of computer networking and pervasive computing, both in international journals and conference proceedings. He is on the editorial board of the *Journal of Ubiquitous Computing and Intelligence* (*JUCI*), and on the steering committee of the *IEEE International Conference on Pervasive Computing and Communications (PerCom)*. He has served as general chair for the 6th *IEEE Symposium on a World of Wireless, Mobile and Multimedia Networks (WoWMoM 2005)*, and as program chair of the 4th *International Workshop on Mobile Distributed Computing (MDC 2006)*, and the *1st International Workshop on Sensor Networks and Systems for Pervasive Computing (PerSeNS 2005)*. He has also served on the Technical Program Committee of many international conferences including *IEEE PerCom* (2003), *IEEE ISCC* (2004-2007), *IEEE AINA* 2006.

Marco Conti is a senior researcher at IIT, an institute of the Italian National Research Council (CNR). His research interests include Internet architecture and protocols, wireless networks, ad hoc networking, mobile and pervasive computing. He co-authored the book ”Metropolitan Area Networks” (Springer, London 1997) and is co-editor of the book ”Mobile Ad Hoc Networking” (IEEE-Wiley 2004). He published in journals and conference proceedings more than 150 research papers related to design, modeling, and performance evaluation of computer-network architectures and protocols. He served as TPC chair of IFIP-TC6 Conferences “Networking2002” and ”PWC2003”, and as TPC co-chair of ACM WoWMoM 2002, IFIP-TC6 WONS 2004, WiOpt '04, and the 6th IEEE WoWMoM 2005 Symposium. He is the TPC chair of IEEE PerCom 2006 conference, TPC co-chair of the ACM MobiHoc 2006

Symposium, and general co-chair of IEEE WoWMoM 2006 Symposium. He is Associate Editor of Pervasive and Mobile Computing Journal, and he is on the editorial board of: IEEE Transactions on Mobile Computing, Ad Hoc Networks journal and Wireless Ad Hoc and Sensor Networks: An International Journal. He served as guest editor, among others, for IEEE Transactions on Computers, ACM MONET, ACM WINET, and Performance Evaluation. He is member of ACM, IEEE, and IFIP WGs 6.2, 6.3 and 6.8.

Enrico Gregori received the Laurea in electronic engineering from the University of Pisa in 1980. In 1981 he joined the Italian National Research Council (CNR) where he is currently a CNR research director. He is currently the deputy director of the CNR institute for Informatics and Telematics (IIT). In 1986 he held a visiting position in the IBM research center in Zurich working on network software engineering and on heterogeneous networking. He has contributed to several national and international projects on computer networking. He has authored more than 100 papers in the area of computer networks and has published in international journals and conference proceedings and is co-author of the book "Metropolitan Area Networks" (Springer, London 1997). He was the General Chair of the IFIP TC6 conferences: Networking2002 and PWC2003 (Personal Wireless Communications), and of the 4th IEEE PerCom conference (PerCom 2006). He served as guest editor for the Networking2002 journal special issues on: Performance Evaluation, Cluster Computing and ACM/Kluwer Wireless Networks Journals. He is on the editorial board of Cluster Computing, Computer Networks and e Wireless Networks. His current research interests include: Ad hoc networks, Sensor networks, Wireless LANs, Quality of service in packet-switching networks, Evolution of TCP/IP protocols.

Andrea Passarella is a Researcher at the Institute for Informatics and Telematics (IIT) of the National Research Council (CNR), Italy. Before joining IIT he was a Research Associate at the Computer Laboratory of the University of Cambridge, UK. He also held a Visiting Student position at the Rutgers University, NJ, USA. He received a PhD and MS Degree in Computer Engineering, both from the University of Pisa, Italy, in 2005 and 2001, respectively. His current research is mostly on ad hoc and sensor networks. Specifically he is currently working on energy-efficient systems and MAC protocols for sensor networks, p2p systems, multicasting, transport protocols, and energy-efficient protocols for ad hoc networks. His research interests also include opportunistic and delay-tolerant networking, and wireless access to the Internet. In 2002 he won the Vodafone Prize for the Best Italian MS Thesis of the year in the field of Technology Innovation. He served and is currently serving in the TPC of several international conferences and workshops, including PerCom 2006 and WoWMoM 2006. He was TPC Vice-Chair for REALMAN 2005, and is currently TPC Vice-Chair for both MDC 2006 and REALMAN 2006. He is an Associate Technical Editor for IEEE Communications Magazine.