Ad Hoc Grids: Communication and Computing in a Power Constrained Environment

Dan C. Marinescu, Gabriela M. Marinescu, Yongchang Ji, and Ladislau Bölöni

School of Electrical and Computer Engineering
University of Central Florida
Orlando, Florida, 32816, USA
E-mail: [dcm,magda,yji,lboloni]@cs.ucf.edu

Howard Jay Siegel

Department of Electrical and Computer Engineering and Department of Computer Science
Colorado State University
Fort Collins, Colorado, 80523-1373, USA
E-mail: hj@colostate.edu

Abstract

In this paper we introduce ad hoc grids as a hierarchy of mobile devices with different computing and communication capabilities. An ad hoc grid allows a group of individuals to accomplish a mission, often in a hostile environment; examples of applications of ad hoc grids are disaster management, wild-fire prevention, and peacekeeping operations. We are concerned with the interplay between computing and communication in the power-constrained environment of an ad hoc grid.

1 Introduction

Computational, service, and data grids are computing and communication infrastructures allowing a large user population to share a rich set of resources and to conduct collaborative activities. Concurrency, heterogeneity, hierarchical design, composition and decomposition, and fault-tolerance dominate the rather long list of desirable features/attributes of all grid software systems. The models of computational, service, and data grids are based on several implicit assumptions:

1. Virtually all components of the grid are stationary. The grid access devices may be mobile, but physical mobility is an exception rather than the norm. The topology of the grid changes over relatively long periods of time.

2. The grid environment is relatively stable. The individual characteristics of the resources made available to the grid by autonomous entities change slowly over time.

3. Heterogeneity is a major concern in a grid environment. However, resources can be grouped into a large number of equivalence classes based upon common properties. The number of entities in each class is substantial and relatively stable.

4. A grid is expected to provide a resource-rich environment and the failure of a small number of nodes or communication links is expected to be hardly noticeable. Fault tolerance is an important characteristic of a grid.

5. Power consumption is not an issue because virtually all grid resources are stationary.

6. There are three forms of virtual mobility: (i) code on demand, when a host loads code from a remote repository and links it to an application; (ii) remote evaluation, when both data and code are sent from one host to another; and (iii) mobile agents, when the code, the data, and the state of a running process are sent to another host. All three forms of virtual mobility are encouraged and supported, but virtual mobility is not a paramount concern.

7. There is a clear separation of societal services (e.g., look-up services) from goal-directed activities, (e.g., computation of an FFT by an application). Societal services are supported by dedicated nodes.

8. There is no single dominant activity in a computational, service, or data grid. Resources are shared among many groups, each with its own agenda. Collaboration among the members of a group is supported, but groups compete with each other for shared resources.

Recent technological advancement in low-power microprocessors and micro-controllers, sensors, embedded devices, and wireless communication, coupled with important practical applications allow us to introduce a new type of grid. An *ad hoc grid* is a heterogeneous system without a fixed infrastructure; all its components are mobile. It consists of a hierarchy of systems with different hardware, software, and communication capabilities.

In an ad hoc grid, (a) the topology changes constantly; (b) there are only a few identifiable classes of systems with common characteristics and a small number of systems in each class; (c) there is little redundancy in each class, a node has a well-defined mission; (d) power consumption is a major concern; (e) virtual mobility is important; and (f) in addition to goal-directed activities, grid nodes perform societal services. There is a dominant activity and all individuals using the grid share a common objective at any given time. The nodes need some degree of autonomy and intelligence to ensure that they can operate in isolation when needed. These differences from traditional grids motivate our interest in research specific to ad hoc grids.

The contributions of this paper are a communication architecture and models of power consumption of different entities involved in the grid.

2 Related work

The hardware components of an ad hoc grid need to be low-power and high-performance to ensure high power efficiency and, implicitly, a long operational life of the system while meeting its performance requirements. Studies on energy optimization in designing as a part of hardware or software were started in earnest in the 1990s, e.g., [6, 9, 11, 13, 16, 20]. The compilers were targeted for power consumption optimization towards the end of the 1990s, e.g., [12, 17, 18]. A comprehensive review of system-level power optimization is presented in [1]. The work has been extended recently to wireless sensor networks with sensing, computing, and wireless communication capabilities integrated into battery-driven sensing nodes.

Embedded systems are reactive systems; they are expected to monitor changes in the environment using sensors and to react to a pre-defined set of events. Embedded systems are pervasive; according to a market study 98% of the 4.4 billion microprocessors and micro-control units produced in 1997 were used for embedded applications [2].

Communication networks supporting an ad hoc grid can be broadly classified as mobile ad hoc networks (MANETs) and sensor nets. There is a substantial body of research results regarding MANETs and some results covering the so-called sensor nets, but there are considerations specific to ad hoc grids, which are not addressed in conjunction with either MANETS or sensor nets.

Coordination is the cornerstone of the computing infrastructure supported by any grid; grid users require composition of multiple services to carry out their tasks, e.g., [3, 4, 5]. In an ad hoc grid, individual tasks are related to a common objective and coordination services are subject to common constraints regarding information assurance and power management. At the same time, individual nodes must exhibit some form of intelligence and be able to operate autonomously for extended periods of time.

3 Architecture and applications

In Figure 1, we introduce a four-level hierarchical model of ad hoc grid systems. We identify four classes of systems: L1, L2, L3, and L4. The processor speed, amount of main memory, amount of secondary storage, speed of communication devices, and sophistication of the software support increase as we move from L4 down the pyramid.

Powerful laptops or regular desktop computers with significant amounts of secondary storage, a variety of I/O devices, high speed network access, and long life batteries, which are mounted on all-terrain vehicles, airplanes, or ships, are called L1 systems. Robots (terrestrial or airborne) and laptops with infrared, wireless, and/or satellite communication are examples of L2 systems. Wearable computers, PDAs with little or no secondary storage, and crossover devices (e.g., portable phones with PDA) are included in the L3 systems category. These devices communicate using infrared and/or wireless communication channels. An L4 system is a low-cost system such as a smart sensor, e.g., a video camera, an infrared detector, a source of light, a generator of acoustic signals, or another type of sensor or actuator. The device is coupled with a

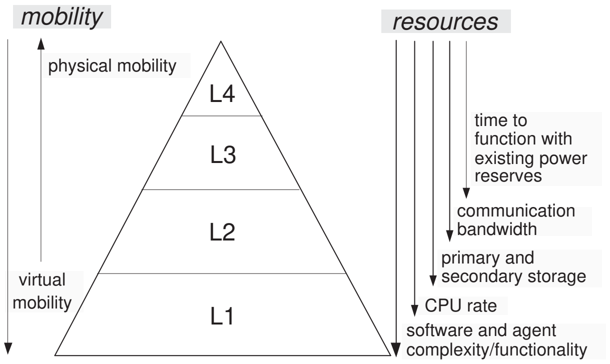

Figure 1: An ad hoc grid is a heterogeneous system without a fixed infrastructure; all its components are mobile. We illustrate a four-level hierarchy of L1, L2, L3, and L4 systems and their properties.

low-end microprocessor without any secondary storage, with little memory, an infrared port, a sonar system, or a very low-speed wireless communication channel, and batteries with varying life-time. Most of the systems are expected to have global positioning system (GPS) capabilities.

There are many potential applications of ad hoc grids for wild fire prevention and control, disaster management, peace keeping operations in a remote part of the world, sporting events, discovery expeditions, natural resource exploration, and battlefield management. Olympic Games, Tour de France, the Trans-Africa car rally, the dog sled race in Alaska, archeological excavations, an expedition to K9, and underwater oil exploration are just a few practical cases when a grid-like environment is necessary to reliably support the coordinated effort of a group of individuals working under extreme conditions.

An ad hoc gird is configured prior to a mission by installing a set of societal services and repositories with mission-specific information on a number of L1 and L2 systems and configuring L3 and L4 systems. Societal services are generic, e.g., authentication, look-up, persistent storage. Other services are mission-specific, e.g., cartographic services, weather services, fire tracking services, population evacuation service, and so on, would probably be part of a wild fire control mission. Some of the task descriptions are generated outside the grid; individuals carrying out the mission create others. For example, the task description report on the fire situation in area X translates internally into a number of actions: locate area X on the map, identify the grid nodes in the vicinity of area X, request reports from all the nodes identified in the previous step, establish the quality of each report based upon a set of criteria (attributes of the reporting node, its proximity to the target area, past history, etc.), analyze the reports given different weights depending upon the quality of each report, construct a synthetic representation of the data and create a final report, identify the nodes where the report should be distributed, and so on.

An ad hoc grid is an open system; new hardware and services may join or leave the system. For example, a set of sensors may be dropped highly in an area of interest, a surveillance aircraft may provide a surveillance service for a brief period of time, and a weather monitoring service may be installed on an L1 system. Communication and computation are deeply intertwined in an ad hoc grid and any optimization must consider both types of activities together rather than separately. An additional requirement is that nodes should exhibit some degree of autonomy and intelligence so they are capable of acting independently when silence periods are imposed or individuals are isolated from the group; thus the need for intelligent agent support. Agents play an important role in intrusion detection and power management. The level of intelligence and autonomy of various agents is related to their mission. For example, a robot and an unmanned surveillance aircraft are expected to be highly intelligent and autonomous while a sensor is not.

4 Communication architecture

The communication architecture of ad hoc grids is illustrated in Figure 2. A subset of nodes forms a virtual backbone (VB). A VB is constructed dynamically; initially it consists of L1 nodes and grows by adding L2 nodes and even L3 nodes, as needed. Thus, each node in the VB set builds-up a virtual micronet (VMN) consisting of lower-power nodes located in its proximity and acts as the VMN head. Each host in the network is either a VB node and a VMN head, or a member node of some VMN. To increase the level of fault tolerance of the system, a member node belongs to a secondary VMN. The VB nodes exchange among themselves the primary and secondary VMN tables during the set-up phase and any time the configuration of VMNs changes. The VB set-up and VMN configuration protocols are based on authenticated and encrypted messages.

Two VB nodes communicate with each other using an asymmetric power constrained optimized link state routing (APC-OLSR) protocol, which we are currently developing. OLSR [7] is a proactive protocol for ad hoc networks; it is an optimization of a pure link state (LS)

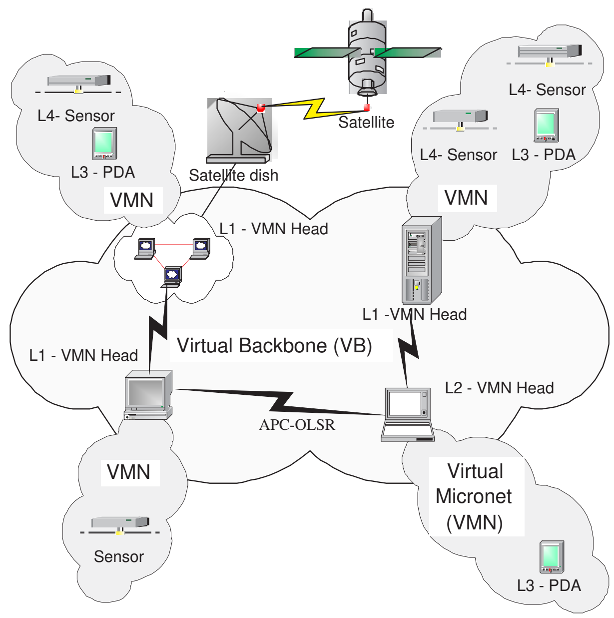

Figure 2: The communication architecture proposed in this paper for an ad hoc grid.

protocol that reduces the size of the control messages and the number of message retransmissions. APC-OLSR is a modified version of the OLSR protocol taking into account the asymmetry of the communication links and the power constraints.

Two VMN member nodes communicate with one another as follows. The sender delivers the message to its primary VMN head. This node searches the set of primary VMN tables to locate the destination primary VMN head, the one managing the communication in the VMN where the destination member node is located. The message is then delivered to the destination primary VMN head node, which in turn forwards the message to the destination member node.

A member node communicates with the head of its VMN in two modes: a base station mode (BSS) or an ad hoc mode (AH). In the BSS mode, the network interface of the VMN member node is in the sleep state most of the time and wakes up periodically to poll the primary VMN head for traffic directed to the node. The polling frequency is established at set-up time and may be changed during the life time of a mission, based upon the function of the satellite and the environment. Even in the case of an empty message queue, the VMN head acknowledges every k-th poll from the member node. If the member node detects a long silence period of its primary VMN head, then it attempts to connect to the secondary VMN head; if this fails, it switches to the ad hoc mode. In the ad hoc mode, the network interface is in the inactive state and continuously listens to the wireless communication channel. BSS is an energy saving mode suitable when the VMN head is directly reachable from the VMN member node. The communication architecture borrows elements from the Bluetooth and IEEE 802.11 standards.

There is an asymmetry of coverage areas among the four classes of nodes; an L4 or L3 node may be capable of listening to an L1 node, but incapable of reaching it for security reasons. The security in a power constrained environment is the most important consideration affecting the communication architecture as well as the MAC (media access control), link layer, and network layer protocols.

When designing energy-aware protocols, we attempt to minimize system cost; including power consumption. Some of the more important observations regarding the design of energy efficient protocols are as follows. (a) Energy consumption and bandwidth utilization are not synonymous. In addition to the cost of transmitting a packet, we need to consider the cost of receiving and possibly discarding a packet. Small packets have high energy costs because of the channel acquisition overhead. (b) Ad hoc operation incurs a much higher cost than base station operation. (c) It is desirable to avoid retransmissions and use some form of forward error correction.

Following Bluetooth, we will most likely define two types of links for the VMN communication. The first, SCO (synchronous connection-oriented), is a symmetric point-to-point link between the VMN head and a member node; The head maintains the SCO link by using reserved slots at regular intervals. The second, ACL (asynchronous connection-less), is a point-to-multipoint link between the VMN head and all the members of the VMN; the head may establish an ACL link on a per slot basis with any member node.

Several logical channels for different types of information could be defined: channel control (CC), link management (LM), and user information channels (UIC). The state machine for a connection will include several states: active, sniff, hold, and sleep. A connection setup procedure must also be defined.

There are two classes of routing protocols for mobile ad hoc networks: proactive and reactive. In proactive routing, each node has one or more tables containing the latest information regarding the routes to any node of the network, very much like traditional routing protocols operate. In reactive, or on-demand protocols, each node does not maintain or constantly update its routing tables with the latest topology and cost information. Instead, the nodes attempt to discover routes on demand by flooding the network with query requests, when a route is needed. The latency of finding a route increases, but the overhead of the protocol is lower.

Dynamic source routing (DSR), ad hoc on-demand distance vector (AODV), the temporary ordered routing algorithm (TORA), and the highly dynamic destination-sequenced distance-vector routing (DSDV) are examples of reactive routing protocols. The optimized link state routing protocol (OLSR) is another proactive protocol proposed recently by a group at INRIA. The key concept of OLSR is that of multipoint relays (MPR) nodes, a selected group of nodes that forward broadcast messages during the flooding process, and only the nodes in the MPR set retransmit a message.

Virtually all reactive protocols for ad hoc networks assume symmetric links and are not power-aware. A power aware routing algorithm attempts to minimize the amount of power expanded for getting a packet from a source to a destination; this objective can be formulated as:

$Minimize \sum_i P(i, i+1)$

with $P(i, i+1)$ the power expanded for transmission between two consecutive nodes i and $i+1$ along a path from the source to the destination.

5 Power management

5.1 Power awareness in an ad hoc grid

Figure 3 illustrates eight different layers of an ad hoc grid: applications and services, operating systems (OS), middleware, transport, network, logical link control (LLC), medium access control (MAC), and the physical layer, and presents some of the most promising areas of research related to power awareness at each layer.

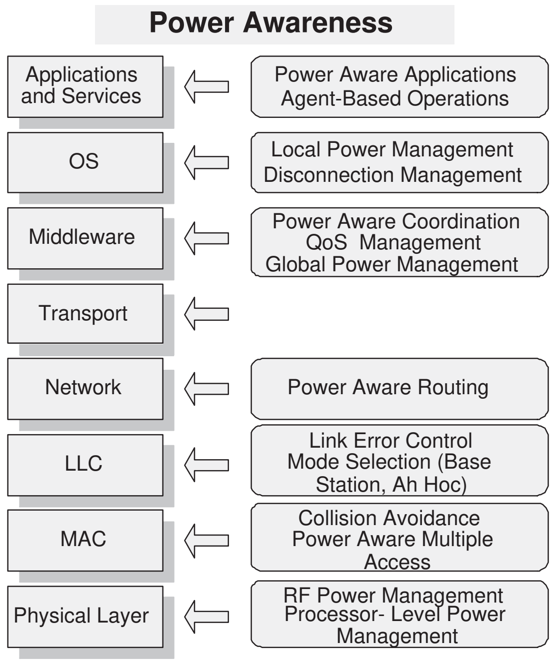

Figure 3: Power awareness in an ad hoc grid.

The applications and services should be designed to adjust their operation to the level of power, e.g., to lower the quality of the solution under low power condition. For a variety of reasons, including the need to reduce power consumption, an application should be able to operate in a autonomous manner under the control of a local agent.

The operating system is responsible for the local power management in a node and for disconnection management. For local power management the operating system relays on models of power consumption of microprocessors and communication hardware, and uses state information and quality of service (QoS) requirements to control them accordingly. Disconnection management is another technique to reduce power consumption by temporarily disconnecting the system; then the OS instructs various communication system components to enter a sleep state characterized by very low power consumption.

The middleware should provide societal services supporting information assurance, such as threat evaluation services and authentication services. The coordination services provided by the middleware should attempt to reconcile local and global power reduction while ensuring end-to-end QoS requirements. Intrusion resilient transport and network protocols offer additional security, but their development should be balanced against the need to use off-the-shelf technology and to support interoperability with other systems. Power aware routing in ad hoc networks has been investigated for some time. Link-specific encryption should recognize the fact that some communication links are more vulnerable and more sensitive than others. To minimize retransmissions, using some form of forward error correction could also help reduce the power used by the communication subsystems.

Power awareness is focused on error control schemes to minimize retransmissions and on attempting to keep low power nodes operating in a base station mode, when the network interfaces are in the sleep state for most of the time, as opposed to the less frugal operation in ad hoc mode. At the MAC layer, power aware multiple access techniques should reduce collisions. At the physical layer, the focus is on using the interfaces exposed by the hardware to select the most power efficient execution mode. Examples of techniques for processor level power management are: frequency scaling – the processor clock is reduced, clock throttling – the clock speed is maintained but the clock signal is gated or disabled for some number of cycles at regular intervals, dynamic voltage scaling – reducing the operating voltage.

By power management, we mean a complex of measures to (a) minimize the power consumption during a mission and (b) extend the operational life time of nodes involved in critical activities. Prior to a mission, we examine the physical components and select only those with power and performance characteristics suitable for the objectives, conditions, and duration of the mission. For each type of physical device, we need to have a model of power consumption in different states and an interface allowing us to control its mode of operation at run time. During the mission, we rely on the power awareness of applications, OS, and middleware to monitor the power reserves of individual nodes and use the models of power consumption to predict the power requirements for different tasks.

A centralized approach to power management on ad hoc grids is not feasible. The amount of communication needed to transfer state information to the central locations would consume significant power. Moreover, the connectivity of the ad hoc network cannot be guaranteed. Some of the power management decisions need to be made autonomously.

On the other hand, the requirements of the ad hoc grid are often given as expressions over a number of nodes. For example, for sensor networks the requirement is a minimum density of active sensors in a specific area. For computational nodes, the requirement might be minimum cumulative computational power available in active nodes. These requirements cannot be satisfied with a local power management approach.

We propose a power management approach based on intelligent agents, integrated with the middleware. The agents use a set of rules to enforce energy minimization policies, while observing QoS requirements. The power management agents can communicate to accomplish non-local requirements. The agents participating in power management can be *observer agents*, *local controller agents*, or *global controller agents*. Controller agents use models of power consumption in conjunction with knowledge bases providing device specific power consumption information as well as QoS task requirements.

Based on their complexity, the power management agents can be:

- simple reactive agents
- agents based on more complex rule based systems
- agents with the abilities to infer new facts, plan a course of action, and learn.

Components of the ad hoc grid deploy different types of agents depending on their position on the mobility/resource hierarchy presented in Figure 1. L4 type devices can deploy only simple reactive agents. While these agents do not have the power to implement formal models of communication, they can still communicate using very simple messages and relying on the emerging properties of the system to achieve a higher level goal. For example, an actively transmitting sensor can inhibit the transmission of nearby sensors, therefore saving their power and preventing them from transmitting duplicate data.

L3 systems can host agents with rules based behavior and formal models of knowledge and communication. These agents can make more elaborate decisions about the power management of the device, and also communicate with other agents. Nevertheless, the power of these devices is not enough to run computationally intensive tasks, such as scheduling and planning based on complex algorithms such as genetic algorithms. At the same time, L3 devices are suffering from the asymmetry of communication channels, where the outgoing communication is more expensive than incoming messages. Therefore, the agents on these devices are not suitable for the role of a central controller agent.

Agents on L1 an L2 devices can implement complex behavior and computation intensive tasks. At the same time, these agents can participate in frequent communication without affecting the power management of the device. While the power management in L1 and L2 type devices is less critical, the power management agents on these devices have important roles for (a) scheduling and planning the active, standby and sleep periods of power critical devices, (b) acting as a communication facilitator for other agents, and (c) monitoring the power consumption through the system.

Communication and computational tasks are tightly coupled together in an ad hoc grid. Our objective is to minimize the overall power consumption, rather than to examine in isolation power-aware routing protocols and energy consumption for various computational activities. For example, if a group of nodes located in the proximity of each other needs access to a service, then the service will be replicated in the vicinity of nodes. This is in contrast to an ad hoc network, where a power aware routing protocol would try only to find optimal routes.

5.2 Power consumption

We discuss the power consumption of several subsystems of an hoc grid node: (i) power supply - battery and dc-dc converter; (ii) sensing - sensors, transducers, and actuators; (iii) computing - microprocessor or microcontroller; (iv) communication - RF communication elements, network interfaces, and signal processing components; and (v) encryption devices.

(i) The operation of a battery depends on factors such as: battery dimensions, type of electrode, electrolyte materials used, diffusion rate of the active materials in the electrolyte, discharge rate, and relaxation effect. The discharge rate is the most important factor that affects the life time of a given battery; when the discharge rate is higher than the rated current capacity specified by the manufacturer, the life time of the battery is significantly reduced.

(ii) There are a large variety of sensors and a wide range of power consumption figures. Passive sensors, such as temperature or seismic sensors, consume negligible power relative to other components of a node. Active sensors, such as sonar rangers, array sensors (e.g., imagers), and narrow field-of-view sensors that require repositioning (e.g., cameras with pan-zoom-tilt), are large consumers of power.

The sources of power consumption in a sensor are:

(a) Sensing devices. Their main functions are: signal sampling and conversion of physical signals to electrical ones, signal conditioning, and analog-to-digital conversion.

(b) The microcontroller unit. It is the intelligent component of the system. It controls the sensing devices and it is responsible for the execution of the communication protocols and the processing of algorithms on collected sensor data. The microcontroller must ensure the performance requirements of the sensor and, therefore, it has a high impact on the power dissipation characteristics of the sensing node.

ii.a. Advances in integrated circuits (CMOS) and MEMS technology make possible the deployment of low power sensing devices. Nowadays, sensing, signal processing, and radio elements can be integrated on the same node. For example, the Rockwells WINS nodes [8] are based on an open modular design and use commercial-of-the-shelf (COTS) technology. They are equipped with the powerful StrongARM SA-1100 processor from Intel, their radio module uses the RDSSS9M Digital Cordless Telephone (DCT) chip-set from Conexant Systems, and several sensor modules (seismic, acoustic, accelerometer, and magnetometer). The nodes can run continuously for 15 hours on two 9 V batteries. A node consumes a peak of 1W. The power consumption of individual components is:

- processor - 300 mW,
- RF - 600 mW in transmit mode and 300 mW in receive mode, and
- the sensors and transducers - less than 100 mW.

The node lifetime can be extended by lowering the duty cycle of data collection and communication. The node components can be put into idle or sleep modes under low-level software control to increase the system operational lifetime. For example, networks of WINS nodes installed to monitor machinery with a duty cycle of 1 second of data collection, a few seconds for processing the data and transmitting the information, and 10 hours of low-power sleep had their battery life extended to several months of operation.

ii.b. Ultra low-power microprocessor and microcontrollers have been developed based on advances in low-power circuit and system design, see Table 1. Studies of power consumption and performance characteristics of specific microprocessors have proposed several techniques to estimate the power consumption of embedded systems [14, 18].

AES 128 bit key, 128 bit data	CMOS implementation	FPGA implementation	Software implementation
Throughput (Mbps)	1,280	640	645
Power(W)	0.065	1.63	41.4
Figure of Merit (Gb/sec/W)	22.8 (100%)	0.39 (1.7 %)	0.0015 (0.06%)

Table 1: The CMOS implementation of a Rijndael processor for an advanced encryption standard (AES) done at UCLA compared with a field programmable gate arrays (FPGA) implementation and with a software implementation using a 1.13 GHz Pentium III processor.

The StrongARM embedded microprocessors from Intel are used by many designers because of their high performance, aggressive power efficiency, and low cost. For example, the latest microprocessor of this series, SA-1110, is optimized for meeting wireless portable and embedded application requirements [19]. The 206 MHz SA-1110 core runs at a supply voltage of 1.75 V and is able of delivering up to 235 Drystone 2.1 MIPS. Its power dissipation is:

- 400 mW in run mode,
- 50 mW in idle mode, and
- 0.16 mW in sleep mode.

Transitioning between operating modes involves power and latency overhead, see Figures 4 and 5.

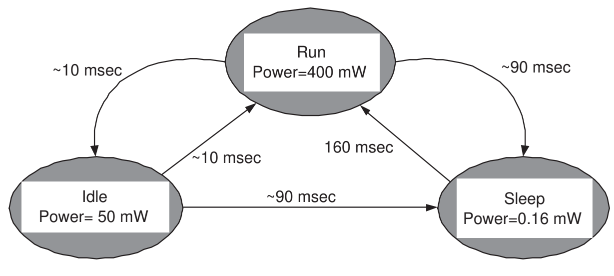

Figure 4: The power dissipation and the transition times between the run, idle, and sleep states of the StrongARM microprocessor.

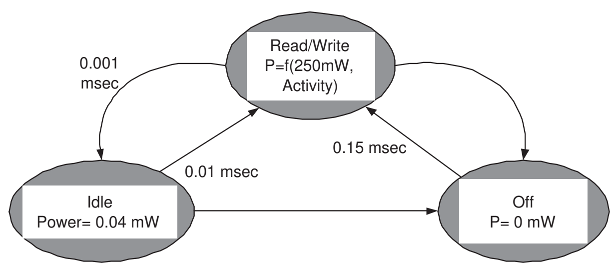

Figure 5: The power dissipation and the transition times between the read/write, idle, and off states of a memory component. The transitions from the idle or the off states to the read/write state are triggered by read or write requests. In the active state, the power consumption is a function of memory activity.

Studies on the experimental Medusa nodes [15] show that a very low power processor such as Atmels 4 MHz AVR 8535 consumes:

- 9 mW in active mode,
- 6 mW in sleep mode and
- only 3 microW in power down mode.

But the processor has modest performance capabilities and cannot perform other tasks besides node localization. Another popular microcontroller, the Atmels Atmega103L AVR, used in SmartDust nodes, consumes around 16 mW in normal mode, but provides even lower performance.

Power reduction techniques applied to high performance CPUs include:

- Voltage scaling. Because power is proportional to the square of the supply voltage, the reduction of the supply voltage is the most effective way to reduce power.
- Clock gating. The clock network is partitioned and only those potions that are needed on each cycle are allowed to toggle; reducing the switched capacitance on the clocks has the most impact on total power consumed.
- System power management reduces the power wasted by the CPU in waiting for inputs from peripherals. The CPU is provided with a hierarchy of power states; each state defines a certain level of activity on the CPU and a certain time penalty for going back into the fully active state. Similar power states are often defined for the memory and for the I/O devices. The system power management mechanism monitors the system activity and enforces the transition of the system components between different power states.

The estimation of the software power consumption is very important for embedded applications. Measurement performed on StrongARM SA-1100 and Hitachi SH-4 microprocessors show that the current consumption of all the instructions in their instruction set is rather uniform. On the average, arithmetic and logical instructions consume 0.178 A, multiplications consume 0.196 A, load instructions 0.196 A, store instructions 0.229 A, while the other instructions use about 0.170 A in the case of Intels StrongARM SA-1100. The variation in average instruction current is +/- 38%. For the a Hitachi SH-4 microprocessor operating at 2.0 V core power supply, the average instruction current is 0.29 A +/- 38% [14]. The current consumption depends only on the operating voltage and frequency of the processor and it is independent to a first order of the code, the addressing modes, or the operands.

(iii) In a laptop, the significant consumers of power are the microprocessor (CPU), liquid crystal display

(LCD), hard disk, system memory (DRAM), keyboard/mouse, CDROM drive, floppy drive, I/O system, and the wireless network card. For example, according to [10], in the case of a Toshiba 410CDT the power is consumed by:

- the display (36%),
- the CPU/memory (21%),
- the wireless interface (18%), and
- the hard drive (18%).

The idle power consumption of a high-end laptop such as NEC Versa is 13.3 W, while for a Toshiba Libretto 60 it is 7 W. Idle power consumption of personal digital assistants (PDAs) is reported to be in the 100-200 mW range.

Studies of power consumers in a high performance CPU using detailed switch-level power simulation show that the clock is by far the largest power consuming component, followed by the datapath, the memory, the control, and I/Os.

(iv) We concentrate our attention on the power consumption of RF units, network interfaces, and MAC-layer protocols including IEEE 802.11 and Bluetooth operating in either base station (BSS) or ad hoc (AH) modes. The power consumed by the RF unit depends upon the transmission distance and the modulation parameters. The electronics component represents the power consumed by the circuits performing frequency synthesis, filtering, up-conversion, and so on.

The RF units are part of the node localization system for an ad hoc grid with infrastructure either pre-installed or not. In the case of an AG with numerous L1 sensor nodes, the use of GPS as a localization system on each individual node has several drawbacks: (i) GPS cannot work indoors or in the presence of obstacles that block the line-of-sight from the GPS to satellites; (ii) the power consumption of GPS will reduce the battery life on the sensor nodes thus reducing the effective life time of the grid; (iii) the size of GPS and its antenna increases the form factor of an otherwise small and unobtrusive sensor node; and (iv) the production cost of GPS can become an issue when large numbers of nodes are produced.

Power consumption tests were performed on the radio used in the Medusa nodes [15]. The DR300 radio module from RF Monolithics supports two data rates (2.4 and 19.2 kbps) and two modulation schemes (ASK and OOK), and its six transmission power levels range between 0.74 and 0.1 mW. The power consumption of the RF module was measured at different operational modes.

- In transmit mode the power consumption varied between 14.88 mW (2.4 kbps, OOK modulation) and 17.76 mW (19.2 kbps, ASK) for transmission power set at 0.74 mW and between 9.54 mW (2.4 kbps, OOK) and 12.36 mW (19.2 kbps, ASK) for transmission power set at 0.098 mW.
- In receive mode - 12.50 mW,
- In idle mode - 12.36 mW, while
- In sleep mode - 0.016 mW.

The power management solutions used for microprocessors and micro controllers, namely to shut them off during periods of inactivity, are not applicable to RF units because their start-up power requirements are very high.

The questions of interest for the link layer and MAC protocols are: the relative costs of sending, receiving, and discarding traffic, and the merits of BSS versus AH mode and of broadcast versus point-to-point traffic. The network interface can be in one of several states: sleep, idle, transmitting, and receiving. Idle mode power consumption incurs a high cost, while the power consumption in the sleep state is low. For example, for the Lucent IEEE802.11 WaveLAN PC card running at 11 Mbps, with a 5 Volt power supply the current intensity in different states is:

- 10 mA - sleep,
- 156 mA - idle,
- 284 mA - transmit.

Thus, energy-conserving link-layer protocols are targeted towards BSS operations, because it allows the network interface to be in the sleep state most of the time and there is no packet forwarding. The number of hosts, which receive and discard traffic, multiplies the cost of receiving and discarding packets. MAC layer protocols should attempt to avoid collisions as well.

(v) Encryption units. The advantages of application specific processors versus other hardware and/or software solution for encryption are illustrated by Table 1.

6 Summary

Ad-hoc grids are a hierarchy of mobile devices with different computing and communication capabilities. An ad hoc grid allows a group of individuals to accomplish a mission, often in a hostile environment. In this paper, we are concerned with the interplay between computing and communication in the power-constrained environment of an ad hoc grid. We describe the generic architecture of an ad hoc grid (Figure 1), and outline the communication architecture (Figure 2). Then we discuss briefly power consumption models for different types of activities (Figure 3), and sketch an agent-based power management system for an ad hoc grid. In a different paper, we will address the problem of coordination of computational tasks.

Acknowledgments

The research reported in this paper was partially supported by National Science Foundation grants MCB9527131, DBI0296035, ACI0296035, and EIA0296179, and by the Colorado State University George T. Abell Endowment.

References

[1] L. Benini, G. de Micheli, "System-level power optimization: Techniques and tools," *ACM Transactions on Design Automation and Electronic Systems*, Vol. 5, No. 2, pp. 115-192, 2000.

[2] K. H. Hodges, "Embedded Micro-component Market Study," *DARPA*.

[3] S. Czerwinski, B. Y. Zhao, T. Hodes, A. D. Joseph, and R. Katz, "An architecture for a secure service discovery service," *Proc. of the 5-th Annual ACM/IEEE International Conference on Mobile Computing and Networking (MobiCom-9)*, ACM Press, pp. 24-35, 1999.

[4] J.C. Fabre, and T. Perennou,"A meta-object architecture for fault-tolerant distributed systems," *IEEE Transactions on Computers*, Vol. 47, No. 1, pp. 78-95, 1998.

[5] T. Kidd et al., "An overview of MSHN: The management system for heterogeneous networks," *Proc. of the Heterogeneous Computing Workshop (HCW99)*, IEEE Press, pp. 78-87, 1999.

[6] C. T. Hsieh, Q. Wu, C. S. Ding, and M. Pedram, "Statistical sampling and regression analysis for RT-level power evaluation," *Proc. of International Conference on Computer Aided Design (ICCAD96)*, pp. 583-588, 1996.

[7] http://menetou.inria.fr/olsr/

[8] http://wins.rsc.rockwell.com/WST_Content.html

[9] C. X. Huang, B. Zhang, A.-C. Deng, and B. Swirski, "The design and implementation of PowerMill," *Proc. of 1995 Symposium on Low Power Design*, pp. 105-108, 1995.

[10] C. E. Jones, K. M. Sivalingam, P. Agarwal, and J. C. Chen, "A survey of energy efficient protocols for wireless networks," *Wireless Networks*, Vol. 7, No. 4, pp. 343-358, 2001.

[11] M. T.-C. Lee, V. Tiwari, S. Malik, and M. Fujita, "Power analysis and minimization techniques for embedded DSP software," *IEEE Transactions on Very Large Scale Integration Systems*, Vol. 5, No. 1, pp. 123-135, 1997.

[12] H. Mehta, R. M. Owens, M. J. Irvin, R. Chen, and D. Ghosh, "Techniques for low energy software," *Proc. of The International Symposium on Low Power Electronics and Design (ISLPED97)*, pp. 72-75, 1997.

[13] F. N. Najm, "Transition density: A new measure of activity in digital circuits," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, Vol. 14, No. 1, pp. 310-323, 1993.

[14] A. Sinha, and A. P. Chandrakasan, "JouleTrack: A web tool for software energy profiling," *Proc. of Design and Automation Conference 2001*, pp. 220-225, 2001.

[15] A. Savvides, C.-C. Han, and M. B. Strivastava, "Dynamic fine-grained localization in ad-hoc networks of sensors," *Proc. of ACM SIGMOBILE 2001*, pp. 166-179, 2001.

[16] T. Sato, Y. Ootugaro, and H. Tago, "Evaluation of architecture-level power estimation for CMOS-RISC processors," *Proc. of The International Symposium on Low Power Electronics and Design (ISLPE95)*, pp. 44-45, 1995.

[17] V. Tiwari, S. Malik, and A. Wolfe, "Power analysis of embedded software: A first step towards power minimization," *IEEE Transactions on Very Large Scale Integration Systems*, Vol. 2, No. 4, pp. 437-445, 1994.

[18] V. Tiwari, S. Malik, A. Wolfe, and M. T.-C. Lee, "Instruction level power analysis and optimization of software," *J. VLSI signal Processing Systems*, Vol. 13, No. 2/3, pp. 223-238, 1996.

[19] http://developer.intel.com/design/pca/applications/index.htm

[20] E. Vittoz, "Low power microelectronics: Ways to approach the limits," *Proc. of International Conference on Solid-State Circuits 1994*, pp. 14-18, 1994.