

# Design of a Wireless Assisted Pedestrian Dead Reckoning System—The NavMote Experience

Lei Fang, Panos J. Antsaklis, *Fellow, IEEE*, Luis A. Montestruque, *Member, IEEE*, M. Brett McMickell, Michael Lemmon, Yashan Sun, Hui Fang, Ioannis Koutroulis, Martin Haenggi, *Senior Member, IEEE*, Min Xie, and Xiaojuan Xie

**Abstract**—In this paper, we combine inertial sensing and sensor network technology to create a pedestrian dead reckoning system. The core of the system is a lightweight sensor-and-wireless-embedded device called *NavMote* that is carried by a pedestrian. The *NavMote* gathers information about pedestrian motion from an integrated magnetic compass and accelerometers. When the *NavMote* comes within range of a sensor network (composed of *NetMotes*), it downloads the compressed data to the network. The network relays the data via a *RelayMote* to an information center where the data are processed into an estimate of the pedestrian trajectory based on a dead reckoning algorithm. System details including the *NavMote* hardware/software, sensor network middleware services, and the dead reckoning algorithm are provided. In particular, simple but effective step detection and step length estimation methods are implemented in order to reduce computation, memory, and communication requirements on the *Motes*. Static and dynamic calibrations of the compass data are crucial to compensate the heading errors. The dead reckoning performance is further enhanced by wireless telemetry and map matching. Extensive testing results show that satisfactory tracking performance with relatively long operational time is achieved. The paper also serves as a brief survey on pedestrian navigation systems, sensors, and techniques.

**Index Terms**—Dead reckoning, pedestrian navigation system, wireless sensor network.

## I. INTRODUCTION

THE ABILITY to locate the position of the user is an essential part of many applications: electronic travel aids (ETAs) for the blind or visually impaired [31], context-aware guidance systems for exhibition touring [13], spatially based applications for a wearable computer [88], public safety services such as E911 [21], and integrated navigation systems for the dismounted infantry soldier [22], [65].

The problem of positioning in an open environment is relatively easy—use Global Positioning System (GPS). In this case, ordinary GPS receivers typically provide 10 to 30 meters depending on the number of satellites visible. However, GPS is limited as a navigation aid by its inability to provide static

heading and its lack of availability when used around obstructions (terrain or man-made) or in the presence of jamming. Therefore, it is necessary to develop a positioning system which can complement GPS in GPS-compromised areas. The difficulties for pedestrian navigation systems (PNS) include the resource and design constraints (size, weight, power) due to its mobile and man-portable nature, accuracy and availability in a wide range of operational conditions, and the restricted choice of technologies due to environment constraints. For instance, the frequent use of a GPS receiver may not be allowed since this significantly decreases the operational time of the battery-powered pedestrian navigation devices.

With advances in computation, communication, and sensing capabilities, large-scale sensor-based distributed environments are emerging as a predominant mobile computing infrastructure [3]. Hundreds or thousands of small, inexpensive and low-power sensors, such as Berkeley Motes [33], can be quickly deployed to monitor a vast field. In this paper, we combine this mobile infrastructure with inertial sensing to build a PNS that is capable of working either indoors or outdoors. Fig. 1 illustrates the system concept. A typical scenario involves the reconstruction of the space/time path a subject has taken from an initial to a final location. Our approach relies on the custom-built device, the *NavMote*, that includes sensors, memory, a processor, and RF communication capabilities. While the *NavMote* carried by the subject is out of range of the network, it uses the flash memory to store the accelerometer and compass data. Custom algorithms provide compensation for adverse sensor orientation and calibration, and for collection with possible compression of data. When the subject reenters again into the radio range of the network, the data captured by the *NavMote* are extracted automatically via wireless communication and deposited in the network nodes (*NetMotes*). These data that are distributed within the network are then via multihop transmissions sent to a designated special “relay” or “destination” *Mote* (*RelayMote*) for further processing. The dead reckoning (DR) algorithm detects the occurrence of steps and calculates the distance traveled, and uses direction of movement and distance traveled to generate trajectory points, which are then appropriately displayed. Trajectory correction by a manual operation is provided through a graphical user interface (GUI), so that recognition and judgment by human vision based on map matching can be used.

There are two challenges in our system design: On the *NavMote* side, the resource-poor *Mote* prevents using any computationally expensive algorithms, while size, weight, and cost constraints need to be met. On the network side, a large amount

Manuscript received January 9, 2004; revised June 27, 2005. The work is supported in part by the Defense Advanced Research Projects Agency/IXO-NEST Program under AF-F30602-01-2-0526) and in part by the National Science Foundation under CCR-02-08537 and ECS-02-25265.

L. Fang, P. J. Antsaklis, M. Lemmon, Y. Sun, H. Fang, I. Koutroulis, M. Haenggi, M. Xie, and X. Xie are with Department of Electrical Engineering, University of Notre Dame, Notre Dame, IN 46556 USA (e-mail: lfang@nd.edu; antsaklis.1@nd.edu).

L. A. Montestruque is with EmNet, LLC, Granger, IN 46530 USA.  
M. B. McMickell is with Honeywell DSES, Glendale, AZ 85308 USA.  
Digital Object Identifier 10.1109/TIM.2005.858557

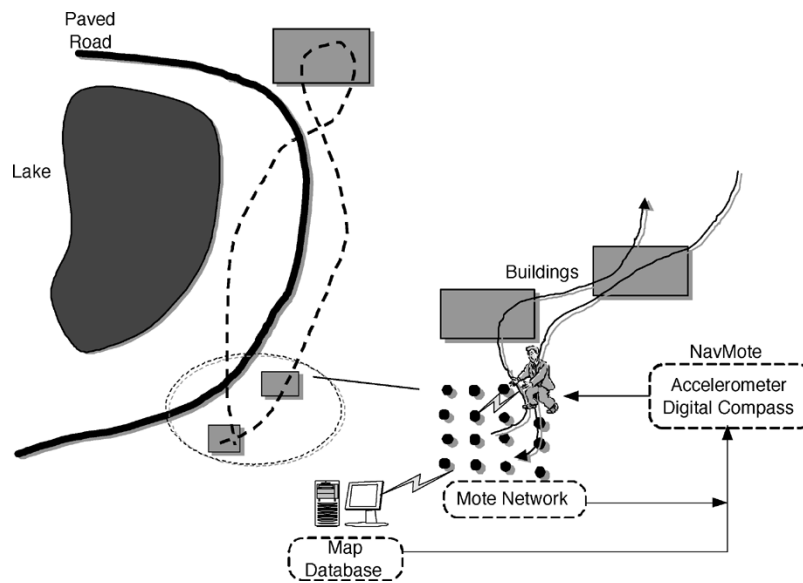


Fig. 1. Pedestrian navigation system concept.

of data need to be downloaded from the NavMote to one or more of (unreliable) NetMotes and forwarded to RelayMote with error-prone wireless channels.

The remainder of this paper is organized as follows. Section II serves as a brief survey on PNS. Techniques, commonly used sensors, and various existing systems are reviewed. We also describe advantages and limitations of various techniques, and discuss how their limitations can be overcome by combining or improving techniques. In particular, the recent advent of wireless networks opens new perspectives for navigation applications. Based on these observations, our PNS is developed in Section III. System details including NavMote hardware/software, sensor network middleware services, and dead reckoning algorithm are provided. Experimental results and navigation performance are presented in Section IV. Finally, Section V concludes the paper.

## II. PNS—TECHNIQUES, SENSORS, AND SYSTEMS

Pedestrian navigation systems should be able to tell users their current position, either absolute within some reference coordinate system or relative to landmarks known to the system. For example, GPS uses an absolute measurement-based approach. On the other hand, dead reckoning (or odometry) is a relative measurement approach, whose fundamental idea is to integrate incremental motion information over time. There are many competing technologies, which vary greatly as to their range, physical characteristics, and how their spatial and temporal accuracy is affected by properties of the environments in which they are used. The reader is also referred to a complementary survey [30]. Common design criteria for PNS dictate that the system should be portable, inexpensive, and of little hindrance to the natural walking pattern.

### A. Systems With Smart Environment

One approach to acquire location information is to make the environment smart, so that it can supply location information

to users by using a special infrastructure or by enhancing the existing communications infrastructure. Examples of such systems include infrared-based systems [87], [96], ultrasonic-based systems [11], various computer vision systems [45], physical contact systems [67], and radio frequency (RF)-based systems [69], [72]. On a large scale, the wide-area cellular network infrastructure [20], [94], or even on a global scale the GPS system [41] establish smart environments that provide location information to users. Triangulation, proximity measuring, and scene analysis are three major techniques to determine a given location [32]. Furthermore, some location information can be derived from communication among users, for instance collocation, or from any adjacency relationships between them with *a priori* knowledge of user's behavior [46], [47]. Table I summarizes the properties of the above location technologies with representative systems. All these systems somehow place the burden on the infrastructure and, unfortunately, cannot operate independent of external assets.

### B. Systems With Unstructured Environment

We cannot always rely on modifying environments to fit the needs of PNS. For example, in a military application, it is not possible to prepare the battlefield beforehand. Alternatively, local sensor-based methods working in their own right can support tetherless positioning in unstructured environment. This is in contrast to smart environment as it rather smartens up mobile devices than their usage environments. In this context, the positioning task is often performed by exploiting a set of wearable sensors which include sensors such as accelerometers, gyroscopes, electronic compasses, barometers, fluorescent light detectors, cameras, temperature sensors, etc. In the following, commonly used wearable sensors will be discussed in detail to give a clear view of their roles in various systems. In each discussion, emphasis is placed on each sensor's functionality and error characteristics. Care should also be taken of the sensors' placement on the subject.

TABLE I  
SOME POSITIONING TRACKING SYSTEMS BASED ON SMART ENVIRONMENTS

Trial/Product	Technique	Accuracy	Weakness	Strength
Active Badge [96]	Diffuse infrared cellular proximity	Room size	Sunlight and fluorescent light interfere with infrared, dead spots, potential compromises to user privacy	Ease of deployment
Active Bat [11]	Ultrasound time-of-flight lateration	3cm in 3D	Required ceiling sensor grid, a tightly controlled and centralized architecture, hard to maintain	Target orientation information, simultaneously usable by a number of different applications
EasyLiving [45]	Vision, triangulation	Variable	Expensive camera, processing demanding, line-of-sight problem	Unencumbered user, intelligent environment (cataloging room contents, detecting unbadged occupants)
Smart Floor [67]	Physical contact proximity	Spacing of pressure sensors	Does not scale, high incremental cost	Unencumbered user
RADAR [72]	802.11 RF scene analysis and triangulation	3-4.3m	Performance degradation due to noisy license-free band and multi-path fading, wireless NICs required	Uses an existing 802.11 network
TurePosition [94]	Uplink Time Difference of Arrival (U-TDOA)	100m-300m	Expensive infrastructure, multi-path problem, mobile phone needed	Consistent performance in various environments
GPS Pathfinder Pocket [71]	Radio time-of-flight lateration	2-5m after differential correction	Not indoors or urban canyon, high power consumption, expensive infrastructure	Absolute accuracy

1) *Accelerometer*: Accelerometers and gyroscopes, also called inertial sensors, are used to measure the rate of acceleration and rotation, respectively. A survey of commercially available accelerometers and gyroscopes, together with innovative applications, is presented in [95]. The Field Robotics Center at Carnegie Mellon University also provides a list of commercial sensors [26]. Measurements are integrated twice (or once, for gyroscopes) to yield position. Inertial sensors, thus, have the advantage that they are self-contained, that is, they do not need external references. Moreover, the output rate of inertial sensors could be much faster than GPS. The principal application shortfall of inertia sensors is that the integration of the accelerometer's (or gyroscope's) drift and noise results in errors that would accumulate quickly [55]. A theoretical study of the errors caused by the noise in accelerometer is given in [93]. Recommendations are made regarding the calibration of accelerometers prior to their use. Methods such as Kalman filtering are employed to reduce errors due to the random bias drift [10], [25].

DR relying on inertial navigation has become the *de facto* backup method to generate position in the absence of GPS signals. Together with gyroscopes which are an intrinsic part of such inertial navigation systems (INS), accelerometers are now being considered for the function of measuring velocity, one of the requirements of DR. Recent progress in the development of integrated microelectro mechanical systems (MEMS) using surface micromachining technology are now enabling new form factors and price points for this function [85]. Accelerometers find a wide spectrum of applications in mobile robot or land-vehicle navigation [1], [10], [97]. Test results for mobile robot navigation, however, have been generally poor due to extensive drift and the poor signal-to-noise ratio at lower accelerations [14]. To obtain acceptable performance, accelerometers with bias ac-

curacies several orders of magnitude better than what is commonly available today are required. Besides, accelerometers are sensitive to uneven ground because any disturbance from a perfectly horizontal position will cause the sensor to detect a component of the gravitational acceleration. The situation is a lot worse for distance measurement in pedestrian navigation when an accelerometer is attached to the subject's body. As the axis of the accelerometer pitches, rolls, and yaws during walk, the measurement acceleration are heavily affected by the gravitational acceleration.

One way to mitigate the drift problem is to take advantage of the fact of *zero velocity updates* with each footfall between leg swings [22], [79]. The integration of inertial measurement is only performed during the swing of legs and the velocity errors can be reset with each step since the true velocity must be zero if the INS is known to be stationary. To obtain maximum benefit from the zero velocity updates the inertial measurements should be as close to the ground as possible, perhaps in the sole or the heel of the boot (where the velocity is not influenced by body sway). Local ground noise and coupling of foot motion to the inertial sensors must be minimized and time at zero state should also be maximized. The experimental results show that very accurate distance estimation can be obtained, at the cost of extensive computation, careful system calibration, and extra hardware such as the long wiring between the foot unit and the module carried elsewhere on the body.

To calculate the walked distance, an alternative approach is to totally avoid integration of measurements, but rather make use of walk dynamics contained in the accelerometer signals such as frequency, maximum/minimum amplitude, etc. In this case, the inertial sensors may be located in a waist pack. When people walk, there is vertical movement of the body with each step. An intuitive way to measure distance walked is to use

vertical axis acceleration signal to determine how many steps have been taken, and then multiply the number of steps taken by the average step length. A common algorithm for step counting uses some manner of peak detection. Unfortunately using a fixed value for stride length will always result in a low-accuracy system. One important feature of a human's walking pattern is a good correlation between the step length and the step frequency. Specifically, a linear relationship between step size and walking speed is apparent, although this relationship breaks down as a person transitions from walking to jogging or running [60]. By simple algebraic manipulation, one can express step size in terms of step frequency, which is easily computed from the step detections. A number of step models [48], [51], [52] are based on this biological characteristics. For a detailed account of step length estimation models and error analysis of the step length estimate, see [56]. Particularly, walking speed strongly influences the amplitudes of the acceleration signal. Therefore, the step length can be approximated with minimum latency by using a simple formula [2]:

$$\text{step length} \approx \sqrt[4]{A_{\max} - A_{\min}} \times K \quad (1)$$

where  $A_{\max}$  (or  $A_{\min}$ ) is the maximum (or minimum) vertical acceleration in a single stride and  $K$  is a constant. The assumption is that the leg is a lever of fixed length while the foot is on the ground. In this model, a differential acceleration measurement results in reduced sensitivity to drift. More importantly, it is much easier to implement online calibration algorithms for a one-parameter model than for a multiparameter model, cf. [48], [51], [52], [56]. Although the parameter  $K$  is user specific, our test results show that it does not vary too much from person to person. In the case of the accelerometer attached on a subject's ankle, a formula similar to (1) is derived in [43].

In addition to being used for velocity/distance measurement, accelerometers also measure the degree of tilt or inclination. The acceleration sensors have dc response and, therefore, can accurately measure the gravitational G-force. When the sensor is static or quasistatic, the accelerometers will indicate the roll and pitch of the device. This information is useful in azimuth calculation using electronic compass readings as illustrated below.

2) *Gyroscope*: Micromachined gyroscopes measure yaw or angular rate by picking up a signal generated by an electromechanical oscillating mass as it deviates from its plane of oscillation under the Coriolis force effect when submitted to a rotation about an axis perpendicular to the plane of oscillation. Gyroscopes are sourceless and relatively immune to environmental disturbances and have been widely accepted in automotive, aerospace, defense, medical, and industrial applications [59]. The foremost problem with gyroscopes is errors mainly caused by bias drift and noise; in low-cost gyroscopes, measurement errors integrate into heading errors at a rate of 15–100°/h. Powerful signal processing and filtering are necessary to cancel drift and artifacts in the signal. Excellent discussions of the nature of error sources can be found in [1]. A thorough treatment of all types of gyroscopes is given in [23], while theoretical and experimental evaluation of gyroscopes for mobile robots are provided in [10], [15]. Compared with accelerometers, gyroscopes have relatively higher cost, larger size, and higher sensitivity to

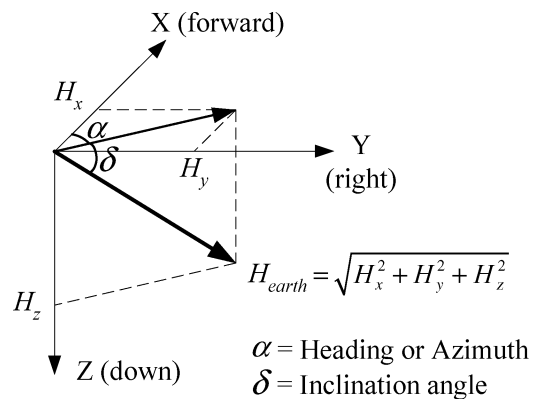


Fig. 2. Earth's field in  $(X, Y, Z)$  coordinates.

temperature and shock, making them less favorable for pedestrian navigation.

Few systems use a gyroscope directly to calculate the distance traveled by the subject. Miyazaki integrated a 0.5-Hz high-pass filtered piezoelectric rate gyroscope to the leg and determined hip flexion extension angle and step length by using a simple step model [63]. However, this method considers thigh and shank as a unique segment and the error reaches  $\pm 15\%$  with subject-dependent calibration. A method based on wavelet transforms is proposed in [6] to compute the values of step parameters from the angular velocity of lower limbs. The estimation error for velocity and step length is around 7%, but the method is computationally expensive and hard to implement on a microprocessor. In [79], a gyroscope attached to a toe is used to compensate for the rotation of the accelerometer: the estimated angular velocity of the foot is integrated to estimate its pitch angle. More often, the gyroscope is located at a subject's waist and used to determine the body orientation. Since the gyroscope's operation principle is completely different from that of a compass, it plays a major role in the filtering of the azimuth. In case of a magnetic disturbance and no turn, the gyroscope remains still while the compass indicates a turn. If both sensors, comparing the respective azimuth rate of change, do not indicate a turn at the same time, then no turn is considered, and a magnetic disturbance is detected. At this stage, only the gyroscope output is used to compute the azimuth of displacement. If both data are coherent, they can be merged through a Kalman filter [43], [49]. Zero velocity updates are also needed from time to time in order to avoid any divergence caused by integration of gyroscope's measurements.

3) *Electronic Compass*: Heading in DR navigation solution is normally obtained from a magnetic compass. Using the earth's magnetic field, electronic compasses based on magnetic sensors provide heading information without requiring any modification of the environment. The earth's magnetic field has a component parallel to the earth's surface that always points toward magnetic north. It is this component that is used to determine compass direction. The angle of the magnetic field to the surface of the earth is called *inclination angle* (see Fig. 2). Since the direction of true north and magnetic north can deviate significantly from each other, the proper *declination angle* needs to be added or subtracted to correct for true north. Declination is defined as angle from true north

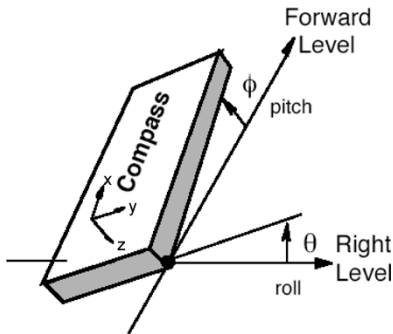


Fig. 3. Compass tilt referenced to the Earth's horizontal plane (figure taken from [18]).

to magnetic north. Declination data for locations world wide can be found at NGDC [66]. Therefore, finding a compass heading, or azimuth, is a two step process: First, determine the  $H_x$  and  $H_y$  horizontal components of the earth's magnetic field; and second, correct for true north using declination data. Compasses are generally implemented with a magnetoresistive (MR) sensor, a Hall-effect sensor, or a set of orthogonal coils referred to as a "fluxgate." Compared to fluxgate sensors, MR technology offers a much more cost-effective solution. Due to their higher sensitivity, MR sensors are also superior to Hall elements in pedestrian navigation.

The determination of the horizontal component of the Earth magnetic field requires implicitly the knowledge of the horizontal or vertical plane. This is commonly done by sensing the gravity vector at rest using accelerometers. To compute then the azimuth of walk, one has to constantly compute the attitude of the sensor in order to correct the measured magnetic values, a scheme often called *tilt compensation* (see Fig. 3). The  $X$ ,  $Y$ , and  $Z$  magnetic readings can be transformed to the horizontal plane ( $H_x, H_y$ ) by applying the rotation equations

$$H_x = X \cos \phi + Y \sin \theta \sin \phi - Z \cos \theta \sin \phi \quad (2)$$

$$H_y = Y \cos \theta + Z \sin \theta. \quad (3)$$

It turns out that the azimuth error depends on the inclination angle, the pitch angle, and the azimuth itself, which can be written as [49]

$$\Delta\alpha = -\Delta\theta \cdot \tan \delta \cdot \cos \alpha - \Delta\phi \cdot \tan \delta \cdot \sin \alpha \quad (4)$$

where  $\Delta\alpha$  is the azimuth error,  $\delta$  the inclination angle, and  $\phi$  the pitch angle. This relation explains that the error in determining the attitude angles affects directly the azimuth and its effect strongly depends on the azimuth itself. For mid-latitude ( $\tan \delta \approx 2$ ), a  $1^\circ$  error in tilt can cause nearly  $2^\circ$  of heading error. A more complete error model is provided in [57].

Independently to these errors, the earth field at the compass may be superimposed by other magnetic fields or distorted by nearby ferrous materials. An efficient compensation of such effects is required in order to achieve reliable azimuth readings [90]. Only errors caused by deterministic interference sources can be compensated. Basically, the deterministic interference can be categorized as two types—*hard iron* and *soft iron* effects.

Hard iron effects are caused by magnetized objects, which are at a fixed position with respect to the compass. For example, some type of batteries are strongly magnetized; therefore, their closeness to the compass should be avoided. Soft iron effects occur due to distortion of the earth field by ferrous materials. Even the presence of a human body distorts the magnetic field around the compass. To compensate these effects, a rigorous approach would require the determination of 12 parameters at known elevations, not feasible for pedestrian navigation application. A simplified approach consisting of determining only the corrections in the horizontal plane is more convenient [18]. The four parameters are two scale factors ( $X_{sf}, Y_{sf}$ ) and two offset values ( $X_{off}, Y_{off}$ ). The ( $\bar{H}_x, \bar{H}_y$ ) values used to compute the azimuth are

$$\bar{H}_x = X_{sf} \cdot H_x + X_{off} \quad (5)$$

$$\bar{H}_y = Y_{sf} \cdot H_y + Y_{off} \quad (6)$$

$$\text{Azimuth} = \text{atan2}(\bar{H}_y, \bar{H}_x) \quad (6)$$

where ( $H_x, H_y$ ) are tilt-corrected magnetic values. The  $\text{atan2}()$  function computes the principal value of the arc tangent of  $\bar{H}_y/\bar{H}_x$ , using the signs of both arguments to determine the quadrant of the return value.

For nondeterministic error signals, package material that has a good shield effectiveness for all magnetic interferences, but permeable to the earth magnetic field is desirable. The fact that the lower frequencies are harder to shield means that, in the end, the earth field gets through the shield. The performance of various materials with different frequencies and source locations are summarized in the shielding effectiveness tables [68], [98]. Steel was found to be the appropriate material since it has a good behavior over a wide range of frequencies. For example, for 6-mm-thick steel and an outside magnetic field of 30 Hz, the magnetic interference inside the shield is 149 times lower [58]. On the other hand, basic prefiltering technique proves to be a very effective compensation scheme for random disturbances [74]. Other compensation methods depend on measurements from other sensors or systems, such as gyroscope [35], [49], [89] or GPS [36], [38].

4) *Barometer*: The continuous knowledge of the altitude opens possibilities for three-dimensional (3-D) positioning. In pedestrian navigation, altitude changes when taking slopes, stairs, or elevators with changes in the air pressure, which can be measured by the barometer attached to the human body. Determination of altitude changes using barometers, however, is complicated due to the following facts. First, the natural air pressure changes with the weather or building conditions. Second, as the air pressure changes slightly during the vertical movement ( $0.1 \text{ mb} \approx 1 \text{ m}$ ), the output of the barometer needs to be amplified exceedingly. Thus, the sensor resolution or electrical noise may cause oscillations between two altitudes, even on flat ground. Third, the vibration of the human body could also change the output of the barometer.

In [79], a bandpass filter (BPF) is constructed to reject slowly varying components and noisy components in the barometer signal. The vertical movement is derived by integrating the output of the BPF and calibrating it using a mean value of seven subjects. The estimation error is within 11%. The improvement

of the sensor resolution is possible by detecting the situations (slopes, stairs, or elevators), followed by an appropriate pressure treatment [49]. If a barometric reference station is available, differential barometry is also viable. Several tests have shown that precision better than 50 cm in absolute altitude can be maintained up to several kilometers from the reference station, even in changing weather conditions [50].

5) *Vision Sensor*: Vision sensors (cameras) have been successfully used in the past to provide motion estimates between consecutive image frames in mobile robot and aeronautical applications, see, e.g., [4], [75], and [76]. Some of common computer vision algorithms include mesh-based modeling, neuro-fuzzy classification, simple shape fitting, feature extraction-based tracking, and shape-volume approximation. Most of these algorithms are computationally demanding and are, thus, unable to deliver high-quality motion capture data in real-time using current processing power. Vision algorithms also suffer from a lack of robustness. Camera rotations or image noise can cause vision tracking failures or instabilities.

The operation principle for a *visual odometer* (incremental stereo ego-motion), which uses stereo vision system to correlate common features in two images, is briefly described next. Features of images obtained from left and right cameras are detected and matched. With the known geometry between cameras, the positions of the features relative to the cameras can be calculated. Hence, the displacement (distance and direction) of the cameras can be estimated from the difference in position of the tracked features in successive frames. For pedestrian navigation, the major shortcoming of this technology is that when the cameras are attached to the walking subject, rather than a wheeled robot or a helicopter, the shaking and rotations of the cameras can considerably degrade the performance of feature tracking [37].

In addition, cameras find a wide spectrum of applications in augmented reality, context awareness, and navigation systems for the blind. They have been used to track orientation [100]; to recognize locations [7], [77]; to detect obstacles [64], curbs, and steps [80], and zebra crossings [40], [81]. Images can also be transmitted back to the human operator at the navigation service center [28].

6) *Other Sensors*: Newer inertial devices with compact, lower precision sensors have become available in recent years. This group of instruments, called motion sensors, is six to eight times less costly than a standard INS. Given their weak stand-alone accuracy and poor run-to-run stability, such devices are not usable as sole navigation systems. Even the integration of a motion sensor into a navigation system as a supporting device requires the development of nontraditional approaches and algorithms [17].

Other sensors might be useful in pedestrian navigation applications including light sensors, microphones, humidity and temperature sensors, pressure sensors, chemical sensors, ultraviolet sensors, etc. Fluorescent light sensors that work by extracting the 60-Hz component of the signal from a photodiode aimed at ceiling are useful in indoor environments. While utility workers find that chemical sensors could save lives, the blind appreciate road information provided by microphones. Users under different situations, thus, find a set of sensors that best meets their needs.

### C. No Sole Solution

From the above discussion, it is clear that that no single tracking technology by itself appears to offer a complete solution to pedestrian navigation. Given the wide range of strengths and weaknesses that different navigation technologies have in different circumstances, one approach has been to combine a set of complementary technologies in ways that the advantage of one technology compensates the drawback of the other to provide acceptable performance. The most common practice is to integrate GPS with DR using a Kalman filter to control navigation error growth [19], [27], [36], [48], [73], [84]. The optimal setting of the filter parameters is not trivial and often requires intensive testing and investigations of the system preferably in predefined surroundings, i.e., reference trajectories for the pedestrian are needed. Intelligent methods to decide when to fuse data would also be useful.

Various other methods have been attempted to improve the navigation performance beyond that of DR. Unusual motions such as sidestepping are handled in [50], [84] by analyzing the pattern of the acceleration signals. In addition, mechanics of walking completely changes once the slope is becoming greater than 10%. The step model which is good for level work is no longer valid when walking on a slope or a stairway. The ability to distinguish various activities of a subject (sitting, level/down-hill/uphill walking, jogging, etc.) [5], [51], [82], [91] helps to develop adaptive step length estimation methods. Knowledge about the environment and the constraints that it imposes on navigation can serve as an important source of information to correct for inaccuracies in the system. This concept is known as *map matching*. The difficulty here is how to reconcile inaccurate locational data with an inaccurate map. See [99] and references therein. Moreover, the position information must be transmitted via a properly working communication channel, especially in health care -related applications [53], [86]. An appropriate communications function is, thus, desirable for PNS. The ultimate goal is for PNS to support robust and accurate tracking in arbitrary environments and conditions: indoors, outdoors, anywhere the user wants to go.

In the end, it is worth mentioning another type of positioning systems with different considerations: electronic travel aids (ETAs) systems. ETAs are not concerned with globally directing the user toward a desired target but rather with local obstacle avoidance. In this case, commonly used technologies include, inter alia, optical triangulation [12], ultrasonic ranging [78], auditory localization [42], mobile robotics technology [83], and active beacon positioning [61].

### D. New Perspectives

With the advancement of MEMS technologies, wireless networks consisting of tiny sensor devices hold the promise of revolutionizing sensing in a wide range of application domains because of their flexibility, low cost, and ease of deployment [3]. Among the most common such devices today are Berkeley Motes (manufactured by Crossbow), which integrate computation, communication, and sensing into a single small device [33]. Their peer-to-peer networks can scale to thousands of devices, from which can emerge advanced behavior. The Motes

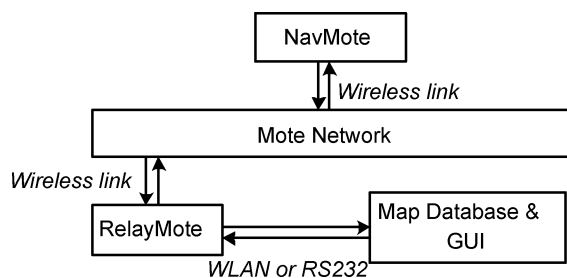


Fig. 4. The overall system architecture.

run an operating system called *TinyOS* [92], which fosters an event-based execution model and supports flexible component modules. The constrained computation power, battery power, storage capacity, and communication bandwidth of the tiny devices, however, pose challenging problems in the design and deployment of such systems.

One exciting application provided by wireless sensor networks is distributed detection and tracking of moving objects. The central idea is to utilize decentralized processing within the network to build a better picture of what is going on (locally), before reporting the results (centrally). This notion, which relies on collaboration among sensors, reduces false alarms and avoids sending all the sensor measurements back to a centralized processing unit [62]. In [34], accurate personnel and vehicle tracking has been achieved using wireless MSTAR sensors. Magnetometer sensors and micropower impulse radar (MIR) sensors are used to detect, classify, and track various types of objects in a variety of applications. In [16], emphasis is placed on location-aware data routing to conserve system resources, such as energy and bandwidth. In our project, the wireless sensor network is used in a unique way for pedestrian navigation, mainly as a communication network.

It is envisioned that, in the future, wireless sensor networks will be an integral part of our life, more so than the present-day personal computers [3]. In addition, the accurate position of users will be known via PNS. These characteristics of future mobile environments will enable the development of location-aware and, more generally, context-sensitive applications.

### III. NAVMOTE-SYSTEM DESCRIPTION

#### A. Overview and Objective

The system involves a dead reckoning module (DR module), self-organizing wireless ad hoc networks (sensor Mote network) and an information center with map database. The DR module consists of the Leica Geosystems DMC-SX three-axes accelerometer and magnetic compass (Leica Vectronix AG [54]) married to a generic wireless controller board, with the radio, the processing, and the power storage all integrated. The controller board for the DR module and network nodes both use the open-source Berkeley Motes with the *TinyOS* (operating system) [92]. The NetMote network plays a critical role in NavMote calibration, trajectory data collection, and exfiltration. The information center is responsible for trajectory generating and displaying, map matching, and other purposes.

The specific objective is to demonstrate the effectiveness of architecture shown in Fig. 4 and the feasibility of building a

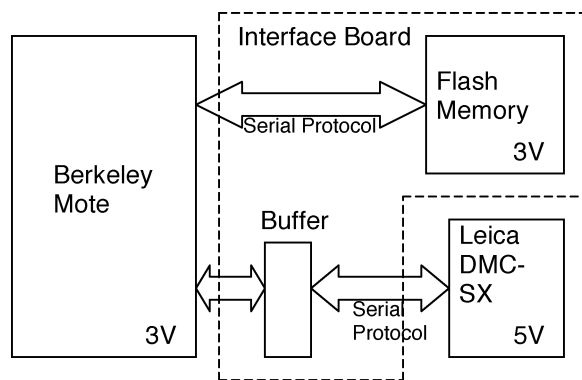


Fig. 5. NavMote hardware.

NavMote (hardware/software) that cooperates with the sensor network using Mote network middleware. The power of the network comes in the ability of NavMote to communicate with any NetMote that may come within range. If the network became commercially popular, it could grow organically. It is this vision shared by the system design where we bring wireless sensors into our everyday life. Note that the present application differentiates itself from the traditional sensor network applications in several ways. See [24] for a detailed discussion.

A side remark is that the system is mainly used to track or monitor the user's position with potential security applications rather than to provide high-level interactive travel aids (For such systems, interested reader is referred to [8], [29].) In particular, the NavMote does not maintain any kind of rough position estimate while is out of contact with the network. Once the information center computes the accurate pedestrian trajectory, the information will not be relayed back to the NavMote. If necessary, the NavMote can be reset for a new tracking task via wireless telemetry starting from a new initial position known to the information center. This design is by no means restrictive since positioning is the core functionality of any navigation systems [53] and the system can be augmented so that the user can know its own position.

#### B. Technical Approach

1) *NavMote Hardware/Software*: NavMote uses a special mobile sensor Mote to collect data and process these data which are automatically downloaded to the Mote network when in proximity. Fig. 5 shows three building blocks of NavMote: Leica Digital Magnetic Compass, interface board, and a Berkeley Mote. Stereolithography is used to manufacture the housing case. Separated from these components is a 9-V battery which is placed in a pocket of the subject. The resulting design allows substantial cost saving as well as the flexibility to change components without considerable redesign efforts. The NavMote is compact as shown in Fig. 6. With a weight of 100 g (not including battery), the NavMote is unobtrusive when fixed to the middle of the waist of the subject. In the sequel,  $X$ ,  $Y$ , and  $Z$  axes denote forward, right, and down directions of the NavMote, respectively.

The core of NavMote is the Leica Digital Magnetic Compass (DMC-SX), which consists of three MEMS accelerometers and three magnetometers (MR), producing an azimuth ac-

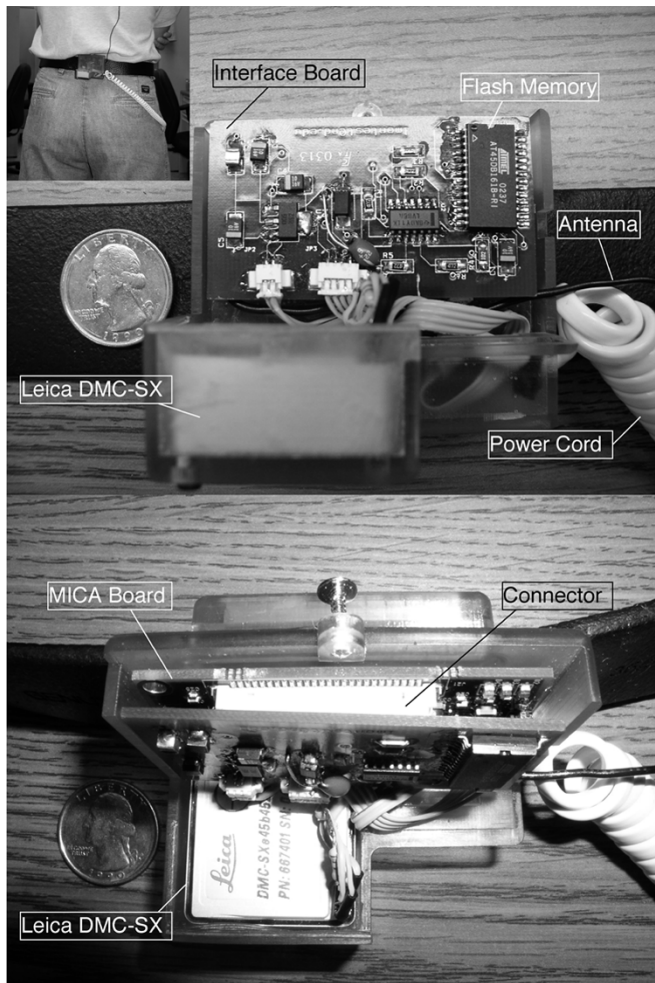


Fig. 6. Inside view of the NavMote.

curacy of  $0.5^\circ$  (2 sigma); 3-D-accelerometer range of 2 g; 3-D-magnetometer range  $\pm 100 \mu\text{T}$ , typical resolution  $0.01 \mu\text{T}$ . The DMC weighs less than 28 g, and measures  $31 \times 33 \times 13.5$  mm. It incorporates amplification, signal conditioning, temperature compensation, and internal voltage regulation, all in a small package. The DMC outputs accelerations and magnetic field in three directions, to compute the azimuth, bank, and elevation. The accelerations are also used to detect step events. The DMC functions at 5 V and the Mote at 3 V. The devices' operating period depends on sensor use, power-saving mode, and so on. At present, the NavMote works with a Lithium 9-V battery for about 6 h continuously.

The NavMote incorporates a Berkeley Mote to provide processing power, temporary storage, and communications to the network [33]. Its central processor is an Atmel ATMEGA128L 8-bit processor running at 7.3728 MHz. This microcontroller has 128 kB of programmable flash memory and 512 kB of external flash memory. The Mote also provides radio communication at 433 MHz. The radio has eight channels and is capable of transmitting data at a rate of 38.4 kBaud with a range of 1000 ft (the radio transmission range may be affected by ground absorption effects). The interface board has a 4-MB flash memory, a circular buffer, and other miscellaneous components. A 51-pin connector is used to interface with the Mote. Sensor data (tri-

axial magnetic and triaxial acceleration) are stored in 4-MB flash memory which allows an operation time of 1.7 h with a sampling frequency of 30 Hz. A circular buffer structures and task driven write and read processes minimize blocking of other interrupt driven processes.

The NavMote works on its own when away from the NetMotes. If any NetMote is within range, the NavMote can exchange information with it. For example, (preprocessed) sensor data can download from the NavMote to a NetMote. This NetMote, on the other hand, can transmit calibration data to NavMote. Preliminary data processing is performed onboard the NavMote just before sending data to the network. The first in, first out (FIFO) buffer stores the last eight samples as data come out from the Leica DMC-SX. At every sampling, the data of the buffer are analyzed for detection of extremum points (maximum or minimum detection) of the Z-axis acceleration signal. If an extremum is detected, then the data over the buffer are averaged and transmitted (for all eight measurements.) Transmission of data associated with extremum points greatly reduces the amount of data that needs to be downloaded to the network. This also results in great energy saving since the energy cost of transmitting a single bit is in the order of three magnitudes more expensive than executing a single instruction [70]. A detailed description of the trajectory reconstruction algorithm is given in Section III-B.3.

2) *NetMote Network Middleware Services*: The NavMote downloads its trajectory data to a network of stationary NetMotes where they are logged in each NetMote's electronically erasable programmable read-only memory (EEPROM). The NetMotes then stream those logged data to the RelayMote where they can be accessed by the user to reconstruct the subject's trajectory. The software supporting the application's use of the communication network is called *middleware*. Essentially, middleware is software that lies between the application and operating system services, providing a high-level abstraction of network communication that can be easily used by the application services.

The software supporting the pedestrian navigation application is written using TinyOS/NesC. The software is organized into a stack (see Fig. 7) consisting of three layers; the application, middleware and operating system or OS layers. There are three primary *application services*. These services are itemized here.

- Coordinator: This service manages the interaction of the other application services.
- Dump: This service downloads trajectory data from the NavMote to the NetMote's EEPROM.
- Exfil: This service streams data in a NetMote's EEPROM to the network's RelayMote (base station).

There are five primary *middleware services*. These services are itemized here.

- Ping: NavMote uses this service to determine if it is in the vicinity of a NetMote.
- Backbone: This service builds and maintains a robust minimum hop spanning tree from all NetMotes to the RelayMote.



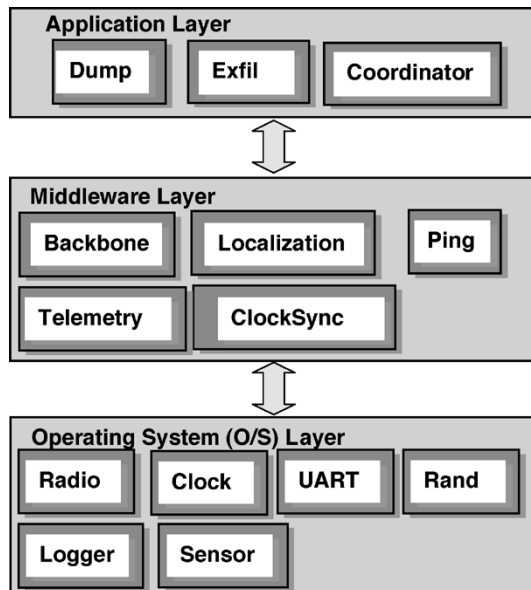


Fig. 7. Software stack consists of three layers.

- **Clock Synchronization:** This service maintains a global clock variable across all nodes in the network.
- **Localization:** This service initializes and maintains a variable representing the NetMote's physical position.
- **Telemetry:** This service periodically sends packets down the network's backbone to the RelayMote. The telemetry packets contain information about the network's current configuration. This service is used by the RelayMote to build a picture (e.g., routes and connectivity) of the entire network.

The *OS services* are software components interfacing directly to the Mote's physical resources such as the universal asynchronous receiver-transmitter (UART), radio, sensors, clock, random number generator, and EEPROM (logger).

The services in the middleware layer are responsible for setting up and maintaining the network infrastructure required to recover data from the NavMote in a flexible and reliable fashion. As soon as a NetMote is reset, it starts up these middleware services in a specific order. The *Clock Synchronization* and *Localization* services are started first in order to initialize the Mote's clock and location variables. Once these services have stabilized, a signal is issued which starts the *Backbone* service. This service automatically builds a minimum hop spanning tree from all NetMotes to the RelayMote. The service is designed to detect changes in link quality that would adversely effect the network connectivity. Upon detecting such changes, the backbone service reconfigures its spanning tree to restore network connectivity. Finally, once the backbone service has stabilized, it issues a signal which starts the *telemetry* service. The telemetry service simply sends packets to the RelayMote that provide the user with a global view of the network's connectivity.

The NavMote uses the *Ping* service to find the network. Upon finding the network, it uses the application layer's *Dump* service to download the trajectory data to the network. The dump service works in an opportunistic manner, dumping as much data as possible to the nearest available NetMote's EEPROM. The download then switches to another download should the original

dump stream be interrupted. Having multiple NetMotes store the NavMotes data is desirable since a single NetMote may not have enough EEPROM to store all of the trajectory data and the subject may be moving from being close to one NetMote to being closer to a different NetMote.

The *Exfil* service is used to transmit data logged in the NetMote's EEPROM to the RelayMote. The *coordinator* on the NetMote ensures that the *Dump* and *Exfil* service operate in a mutually exclusive manner. In this way, the Exfil service does not begin until the download from the NavMote has stopped. Upon starting, the Exfil service begins reading data stored in EEPROM and storing it in a 128-line buffer. Once this buffer is full, the service issues a request-to-send (RTS) to the next-hop on the backbone's spanning tree. All nodes (except the sender and next-hop) hearing the RTS are immediately set to a dormant state, thereby establishing a *basic service cell* (BSC) in which the sender and next-hop can communicate with little interference from other NetMotes. Upon receiving a clear-to-send (CTS) from the next-hop, the Exfil service begins transmitting the buffered data, line by line, to the next-hop. Each transmitted line is explicitly acknowledged (ACK) by the next-hop. Should the sender miss an ACK or the next-hop miss an expected message, then they issue a clear (CLR) message that resets the basic service cell so it is ready for the next RTS message.

The Exfil service essentially streams data from all NetMotes with logged data to the RelayMote. Observed transmission rates using the MICA2 Motes have been about 400 lines (16 data bytes per line) per minute per link. The trajectory data transmitted by the Exfil service follows the same spanning tree used by the *Telemetry* service packets. The final destination for both message streams is the RelayMote. Upon being caught by the RelayMote, the packets are forwarded over the RelayMote's UART to a PC that is running a Java graphical user interface (GUI) called *NetConsole*. The GUI displays telemetry packets that are forwarded to the RelayMote from the network. The data contained in these packets allow the GUI to display the NetMote and NavMote position and neighborhoods. The data also allows the GUI to display the routes and connectivity between NetMotes. Finally, the NetConsole dumps the decoded data packets to a file for subsequent postprocessing.

To help monitor operations of the whole network, a *listener* is also necessary. The listener does not interfere with network operations, but rather eavesdrops on data packets that are sent from NavMote to NetMotes and dumps the decoded packets to a file. The file contains raw data that the NavMote downloaded to the network and can be compared with the file received at RelayMote for diagnostic purpose. The listener also displays the NetMote location (coded in SerialID) and provides controls to restart the NetMotes and to adjust their radio power.

Although the sensor Mote network is used mainly as a communication network in our project, the framework allows an easy extension. The network middleware also provides a method for *in situ* calibration of the Berkeley sensors. Sensor Mote network localization services can be used to provide "absolute" position fixes that can be transmitted to NavMote. These absolute position fixes can then be smoothed with raw data to provide autocalibration of the NavMote while it is in the sensor field.

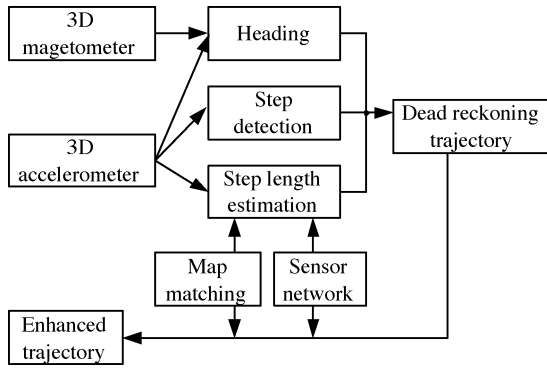


Fig. 8. Signal flow of the wireless assisted dead reckoning algorithm.

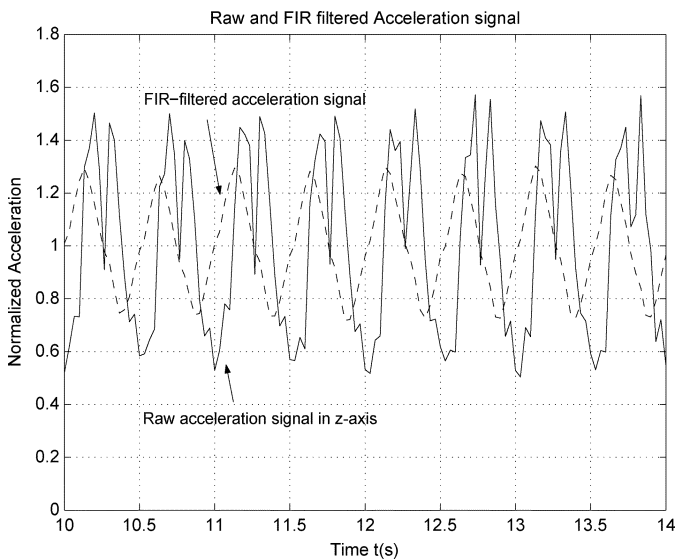


Fig. 9. Raw acceleration signal (solid) versus finite-impulse response (FIR)-filtered acceleration signal (dashed).

3) *Wireless Assisted Dead Reckoning Algorithm*: Our approach to DR is to use the acceleration signal pattern to detect the step occurrences and use the magnetic compass to provide continuous azimuth information. Based on a simplified kinematic model of a person's gait, the walked distance is provided by summing up the size of each step over the step count. Static and dynamic calibrations are essential for compensating azimuth error caused by body offset and walking. DR is further combined with wireless telemetry and map matching to improve navigation performance. Currently, the sensor network only provides initial position fix to NavMote and map matching is not automated but executed by a human operator. Fig. 8 illustrates the signal flow of the algorithm.

Both the step count and the step length estimation require an accurate and robust step detection method. In the following, we show that the step can be reliably detected using only  $a_z$ , the acceleration signal in  $z$ -axis. Due to complicated walking dynamics, the raw signal  $a_z$  has double or triple peaks in every step as shown in Fig. 9. Since the normal walking has a natural frequency no more than 3 Hz, it turns out that a lowpass FIR

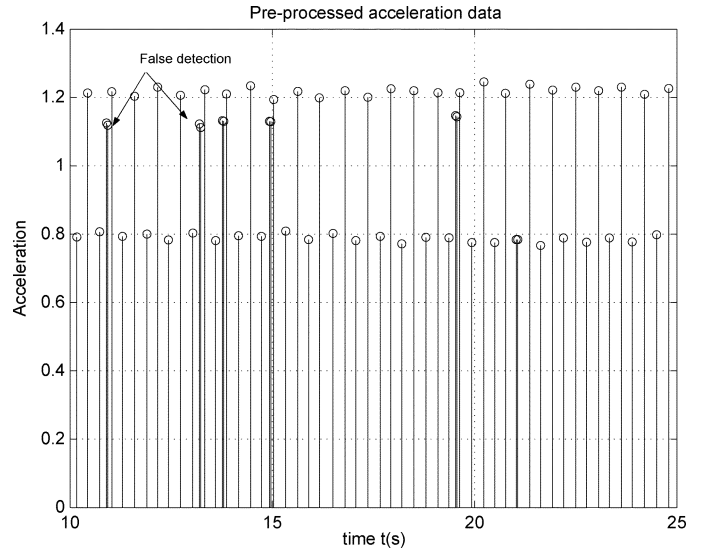


Fig. 10. False detections by simply checking the sign of first-order (backward) difference of  $\bar{a}_z$ .

filter (or simply, a moving average) can be applied to smooth  $a_z$ . The filter output  $\bar{a}_z$  is calculated as

$$\bar{a}_z(t) = \frac{1}{8} \sum_{i=t-7}^t a_z(i) \quad (7)$$

where  $t$  is the sampling time. (In general, it is recommended that the user limit bandwidth to the lowest frequency needed by the application to maximize the resolution and dynamic range of the accelerometer.) Fig. 9 demonstrates that “pseudo” peaks are almost removed from  $a_z$ , resulting in a much smoother signal  $\bar{a}_z$ . Now, the extreme values of  $\bar{a}_z$  can be found by using a conventional extremum detection algorithm. Fig. 10 shows extremum points detected by simply checking the sign of the first-order (backward) difference of the signal  $\bar{a}_z$ , i.e.,  $\text{sgn}\{\bar{a}_z(t) - \bar{a}_z(t-1)\}$ . The detection rule is that if the sign changes from positive to negative,  $\bar{a}_z$  has a local maximum; if the sign changes from negative to positive,  $\bar{a}_z$  has a local minimum. This rule is chosen to facilitate real-time extremum detection. Only those extremum data points are recorded, resulting in a factor of seven of data reduction. (The sampling frequency is 30 Hz but only four extremum points are recorded in every second assuming a step frequency of 2 Hz.) False detections are often observed based on this rule, but they can be avoided by checking additional conditions such as the following:

- whether the maximum/minimum acceleration are above/below the threshold values;
- whether the time since the last maximum/minimum detection is greater than some minimum period;
- whether the maximum and minimum detection alternate.

If the above conditions are true, then the system increases the step count and records maximum and minimum acceleration amplitudes.

After the extrema are detected, the step length is calculated using (1). This technique has been shown to measure distance walked to within  $\pm 3\%$  for the same subject and  $\pm 8\%$  across

TABLE II  
CONSTANT FACTOR  $K$  FOR THREE DIFFERENT SUBJECTS

Subject	A	B	C
Leg Length	0.94	1.06	1.08
$K$ (Nominal Value)	0.50	0.55	0.57

TABLE III  
USING THE SAME  $K$  TO ESTIMATE THE DISTANCE WITH DIFFERENT WALKING PATTERNS (75-m STRAIGHT WALKING ON THE CEMENT SIDEWALK)

Subject	Normal	Slow	Fast
A ( $K = 0.50$ )	75.3	74.2	73.5
B ( $K = 0.55$ )	75.5	74.6	73.3
C ( $K = 0.57$ )	74.9	76.0	72.8

TABLE IV  
DEAD RECKONING PERFORMANCE (THE MEAN AND THE STANDARD DEVIATION ARE CALCULATED BASED ON 20 TESTS OF 100-m NORMAL WALK AT AN OLYMPIC STADIUM)

	Est. Dist.	Err. 1 (no calib.)	Err. 2 (w/ tilt calib.)	Err. 3 (w/ total calib.)
Mean	102.2m	6.8°	2.1°	1.1°
STD	0.7m	2.1°	1.0°	0.5°

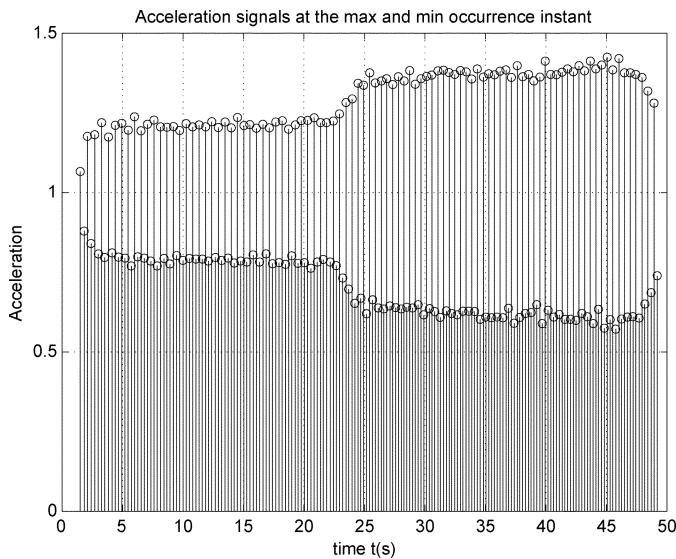


Fig. 11. Amplitudes of  $\bar{a}_z$  change as the subject changes his pace.

a variety of subjects of different leg lengths (see Tables II, III, and IV). Close coupling of the accelerometer to the body is important to maintain accuracy. Note that the biased distance estimation in Table IV is caused by the artificial track surface, which generally results in 5% higher acceleration amplitude than walking on the asphalt surface.

The experimental results demonstrate that the factor  $K$  can be set to be constant for the same subject without a large accuracy penalty. GPS or wireless sensor networks with the localization capability can be combined with (1) to obtain more accurate step length estimation. The same  $K$  can also be used for normal, slow and fast walking on different surfaces. Fig. 11 reports how the amplitudes of filtered acceleration signal  $\bar{a}_z$  change as the subject changes his pace from slow walking to fast walking.

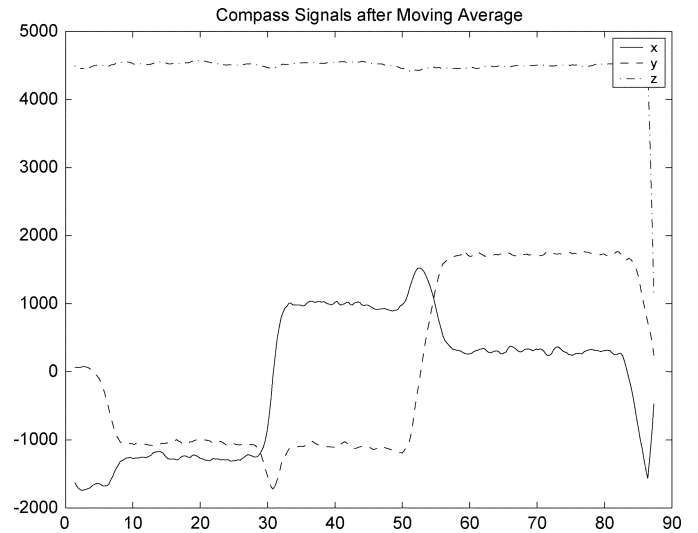


Fig. 12. Compass data after lowpassing filtering.

As expected, the step length differs on different paces. Generally, a shorter distance is obtained for fast walking and a larger variance is obtained on soft surfaces such as grassland. The errors mainly result from the violation of the assumption that the leg has fixed length. An adaptive algorithm that “learns” the user’s walking characteristics could improve the accuracy significantly.

The main source of error in DR comes from the errors in azimuth determination. Lowpass filtering of compass data proves to be very effective in removing random disturbances, as shown in Fig. 12. In addition, dynamic (tilt) and static calibrations are needed to compensate for compass errors.

- Dynamic (tilt) calibration: Dynamic calibration is necessary to compensate the individual errors caused by walking. Lowpass filtering the acceleration signals eliminates the typical oscillations in walking. The filtered acceleration signals are also used to estimate the pitch and roll angle of the NavMote. The pitch and roll angles provide the attitude values to find the horizontal projection plane. The steps to compute tilt-compensated magnetic values ( $H_x, H_y$ ) are as follows:

- 1) moving-average raw compass readings ( $X, Y, Z$ ) to get  $(\bar{X}, \bar{Y}, \bar{Z})$ ;
- 2) moving-average raw acceleration signals ( $a_x, a_y, a_z$ ) to get  $\bar{a} = (\bar{a}_x, \bar{a}_y, \bar{a}_z)$ ;
- 3) compute  $w_x = \bar{a}_x/|\bar{a}|$ ,  $w_y = \bar{a}_y/|\bar{a}|$ ,  $w_z = \bar{a}_z/|\bar{a}|$ ;
- 4) compute

$$\begin{aligned} H_z &= w_x \cdot \bar{X} + w_y \cdot \bar{Y} + w_z \cdot \bar{Z} \\ H_y &= w_y \cdot \bar{Z} - w_z \cdot \bar{Y} \\ H_x &= \bar{X} - w_x \cdot H_z \end{aligned}$$

- Static calibration: When a two-axis magnetic sensor is rotated in the horizontal plane with no disturbances, the output plot of  $H_x$  versus  $H_y$  will form a circle centered at the origin (0,0). However, even the human body can distort the circle, resulting in, e.g., an ellipsoid not centered at the origin as plotted in Fig. 13. The plot data are collected while the subject is walking in a small circle. The method

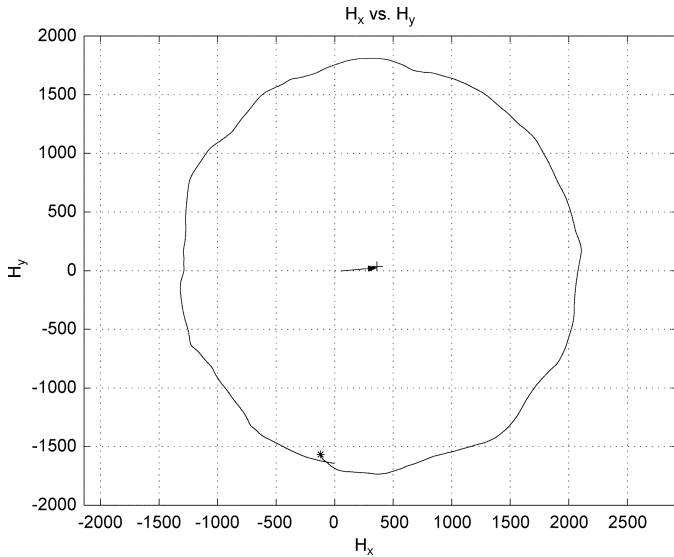


Fig. 13. Body effect on magnetic values.

proposed in [18] is used with slight modification to determine the offset and scale factor values in (5)

$$X_{sf} = \max \left\{ 1, \frac{(H_{y,95} - H_{y,05})}{(H_{x,95} - H_{x,05})} \right\}$$

$$Y_{sf} = \max \left\{ 1, \frac{(H_{x,95} - H_{x,05})}{(H_{y,95} - H_{y,05})} \right\}$$

$$X_{off} = 0.8 \cdot \left[ \frac{(H_{x,95} - H_{x,05})}{2} - H_{x,05} \right] \cdot X_{sf}$$

$$Y_{off} = 0.8 \cdot \left[ \frac{(H_{y,95} - H_{y,05})}{2} - H_{y,05} \right] \cdot Y_{sf}$$

where  $H_{x,95}$  and  $H_{x,05}$  ( $H_{y,95}$  and  $H_{y,05}$ ) are the 95th percentile and the 5th percentile of  $H_x$  ( $H_y$ ) obtained in calibration experiments, respectively. The factor 0.8 is introduced to avoid over-compensation. Finally, the azimuth is calculated according to (6). The declination angle should also be applied to adjust magnetic north to true north.

The effect of total calibration (dynamic and static calibration) has been shown in Table IV. A heading error of  $1.1^\circ$  is obtained after total calibration, compared to a heading error of  $6.8^\circ$  without any calibration. Fig. 14 depicts the effects of the different calibration phases.

With step length and heading estimation in hands, the entire trajectory is reconstructed by piecing the segments together and then displayed against a “map” of the region. In the implementation, step detection is done on NavMote and other computations are done at the workstation connected to the RelayMote. At this stage, the map-matching is not automated but done by a human operator using prior information to match pieces of estimated trajectory to certain features in the maps, such as wall boundaries and building entrances for the indoor environment, and satellite photos in the outdoor environment. The adjustable parameters are starting positions, the step size factor  $K$ , and the compass offset values ( $X_{off}$ ,  $Y_{off}$ ). A detailed discussion of map matching algorithms is out of the scope of this paper. We refer the reader to [99].

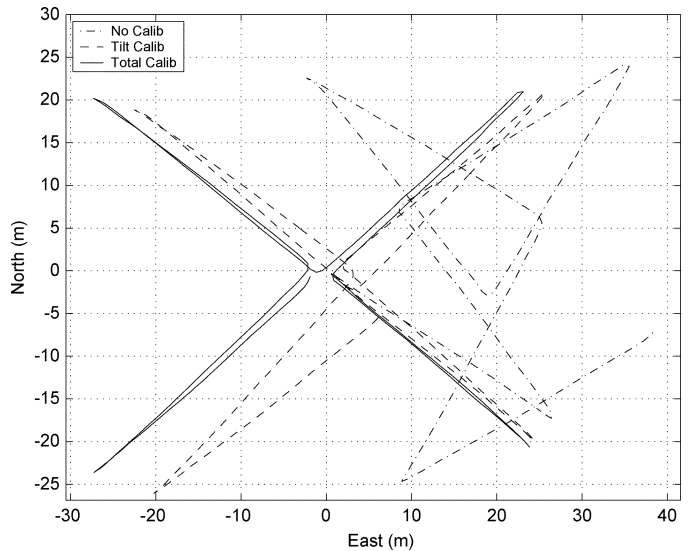


Fig. 14. Effects of the azimuth calibration.

#### IV. NAVMOTE-FIELD EXPERIMENTS

The system was examined in both indoor and outdoor environments. Experimental results show that the navigation performance is satisfactory provided that the indoor environment is not hostile to the NavMote in terms of magnetic disturbances. Fig. 15 shows a trajectory with an “odd” hook: the subject went inside a room in Debartolo Hall (University of Notre Dame). One of the attractive features of the gait model (1) is that it works for frequently changing walking patterns. Fig. 16 shows that a subject went through three doors in a building. Three notches, generated by stopping to open doors, can be easily identified from the figure. The last two events separate themselves by less than 5 s. It will be difficult for step frequency-based step length models, e.g., the one in [39], [48], to correctly estimate step length between such two events since accurate step frequency cannot be obtained at this time.

One experiment took place in an olympic stadium. A subject circled the 400-m track three times. As shown in Fig. 17, the end-to-end (starting point to ending point) error is around 10 m for a trajectory of 1200 m (0.8% of the total distance). The operation time for this experiment is 14 minutes, demonstrating the reproducibility of the tracking error for a short operation time.

The system was also tested in swamp terrain. The subjects took 5- and 3-min walks through the test course. When the subjects returned in the vicinity of the sensor Mote network, the data were automatically exfiltrated by the network. Data received by the RelayMote were processed to generate the subject’s trajectory. The trajectory is plotted against an aerial view of the area, as shown in Fig. 18. The offset and scale factor boxes were provided via a GUI, so that a human operator could manipulate these values to get enhanced tracking performance using human knowledge on the environment.

#### V. CONCLUSION

In this paper, we describe the development of a pedestrian navigation system that uses a three-axes accelerometer and

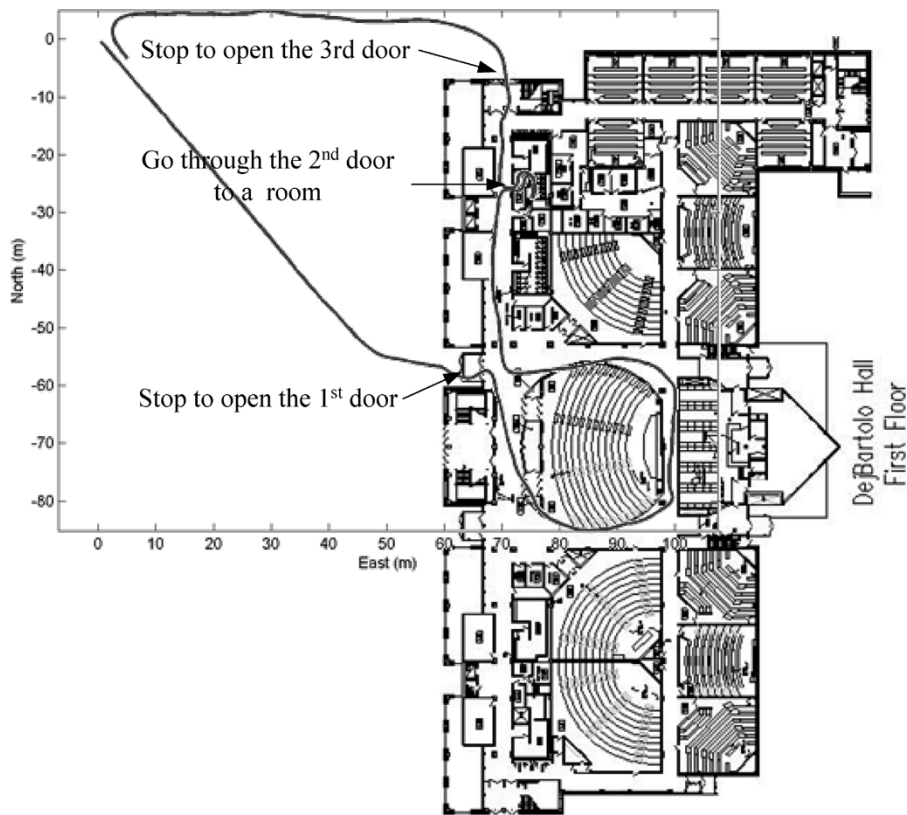


Fig. 15. Test in outdoor and indoor environments.

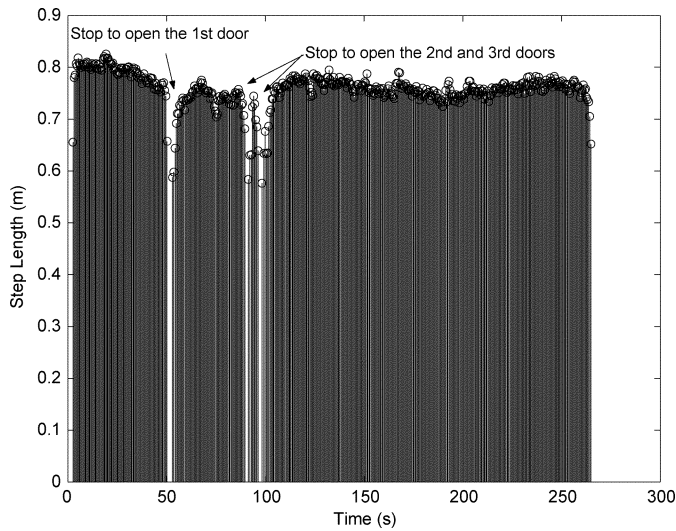


Fig. 16. Subject went through three doors in a building. Three notches (stepsize = 0) is generated by stopping to open doors.

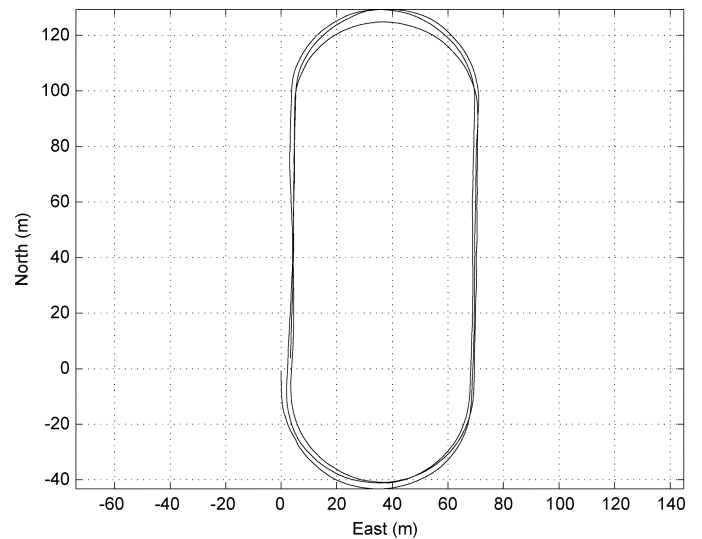


Fig. 17. Repetitions of the same trajectory in an olympic stadium.

magnetic compass (Leica Geosystems) and the Berkeley Motes. Implementing such an innovative commercial-off-the-shelf (COTS) approach is a direct alternative to traditional methods of project development. In the following, we summarize the system from a functional point of view.

NavMote Software implements the following:

- compensation for adverse sensor orientation, calibration, and filtering;
- collection of data from sensors and storage on a non-volatile memory;

- retrieval of data from memory, processing, and transmission;
- detection of the sensor Mote network and data download;
- preprocessing of raw sensor data before data download.

Sensor Mote network services provide the following:

- synchronization: establishes global time reference;
- backbone generation: establishes min-hop routes to the RelayMote;
- streaming service: streams large blocks of data from the NetMotes to RelayMote;

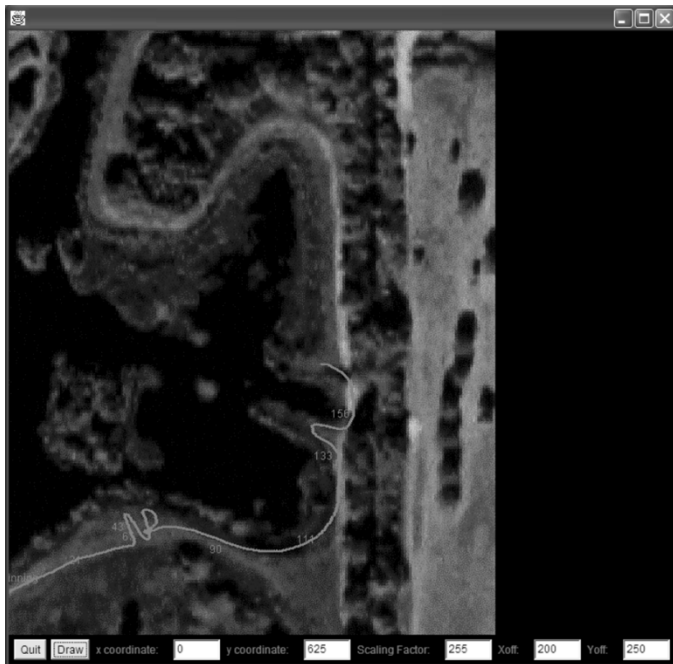


Fig. 18. Three-minute walk in swamp terrain: the letters ND are clearly seen.

- download service: streams large blocks of data from the NavMote to the NetMotes.

Trajectory postprocessing implements the following:

- Java programs on PC connected to RelayMote;
- reassembly of data packet;
- compensation for missing data;
- reconstruction of complete trajectory;
- display of trajectory over area map.

Satisfactory system performance is obtained from the following:

- distance accuracy:  $\pm 3\%$  for same subject;
- heading accuracy:  $1^\circ$ ;
- download time (NavMote to NetMotes): 3-min data file = 3 min download time;
- exfiltration time (NetMotes to RelayMote): 3-min data file = 1 min Exfil time;
- size: currently limited by the sensor and battery sizes.

Although the system performed very well, there is plenty of room for improvement. Whereas outdoor environments typically pose only a few problems to the developed DR algorithm, indoor passages may be more critical. This is mainly due to the method of heading determination relying on magnetic azimuth measurement. One way to solve the problem is to integrate GPS/gyroscope with the magnetic compass. In addition, the download and exfiltration time need to be reduced via improvements in NavMote hardware and software, and in network middleware. In the latest dual-processor design, a slave processor is added to perform the sensor acquisition algorithm, the memory management, and data preprocessing tasks (e.g., differential data compression). The speed of download and exfiltration is greatly reduced. For a 3-min walk, 45 s of download time from the NavMote to network and 25 s of exfiltration time to the RelayMote are typical.

A number of practical challenges remain, including network security, privacy, authentication, mobility management, and provisioning of key services [9]. We believe, however, that our development offers a vision of future mobile environments that may emerge once ubiquitous wireless coverage becomes available.

## REFERENCES

- [1] E. Abbott and D. Powell, "Land-vehicle navigation using GPS," *Proc. IEEE*, vol. 87, no. 1, pp. 145–162, Jan. 1999.
- [2] *Using the ADXL202 in Pedometer and Personal Navigation Applications*, Application Notes AN-602, Analog Devices, [Online]. Available: [http://www.analog.com/UploadedFiles/Application\\_Notes/513772624AN602.pdf](http://www.analog.com/UploadedFiles/Application_Notes/513772624AN602.pdf).
- [3] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, "Wireless sensor networks: a survey," *Comput. Netw.*, vol. 38, pp. 393–422, 2002.
- [4] O. Amidi, T. Kanade, and K. Fujita, "A visual odometer for autonomous helicopter flight," *Robotics Autonomous Syst.*, vol. 28, no. 2/3, pp. 185–193, 1999.
- [5] K. Aminian, P. Robert, E. Jequier, and Y. Schutz, "Level, downhill and uphill walking identification using neural networks," *Electron. Lett.*, vol. 29, no. 17, pp. 1563–1565, 1993.
- [6] K. Aminian, B. Najafi, C. Bula, P.-F. Leyvraz, and P. Robert, "Spatio-temporal parameters of gait measured by an ambulatory system using miniature gyroscopes," *J. Biomech.*, vol. 35, no. 5, pp. 689–699, 2002.
- [7] H. Aoki, "Real-time personal positioning system for wearable computers," *Trans. Info. Process. Soc. Japan*, vol. 41, no. 9, pp. 2404–2412, 2000.
- [8] I. Aslan and A. Krüger, "The Bum Bag Navigator (BBN): an advanced pedestrian navigation system," in *Proc. AIMS 2004*, Nottingham, U.K., 2004. [Online]. Available: <http://w5.cs.uni-sb.de/~baus/aims04/cameraready/P3.pdf>.
- [9] P. Bahl, W. Russell, Y. Wang, A. Balachandran, G. M. Voelker, and A. Miu, "PAWNS: satisfying the need for ubiquitous secure connectivity and location services," *IEEE Trans. Wireless Commun.*, vol. 9, no. 1, pp. 40–48, Jan. 2002.
- [10] B. Barshan and H. F. Durrant-Whyte, "Inertial navigation systems for mobile robots," *IEEE Trans. R&A*, vol. 11, no. 3, pp. 328–342, 1995.
- [11] *The Bat Ultrasonic Location System*, [Online]. Available: <http://www.uk.research.att.com/bat/>, 2002.
- [12] J. M. Benjamin, "Laser cane," *Bull. Prosthetics Res.*, pp. 443–450, 1974.
- [13] G. Bieber and M. Giersich, "Personal mobile navigation systems—design considerations and experiences," *Comput. Graphics*, vol. 25, no. 4, pp. 563–570, 2001.
- [14] J. Borenstein, H. R. Everett, L. Feng, and D. Wehe, "Mobile robot positioning: sensors and techniques," *J. Robotic Syst.*, vol. 14, no. 4, pp. 231–249, 1997.
- [15] J. Borenstein, "Experimental evaluation of a fiber optics gyroscope for improving dead-reckoning accuracy in mobile robots," in *Proc. IEEE Int. Conf. Robotics and Automation*, Leuven, Belgium, 1998, pp. 3456–3461.
- [16] R. R. Brooks, P. Ramanathan, and A. M. Sayeed, "Distributed target classification and tracking in sensor networks," *Proc. IEEE*, vol. 91, no. 8, pp. 1163–1171, Aug. 2003.
- [17] M. E. Cannon, R. Nayak, G. Lachapelle, O. S. Salychev, and V. V. Voronov, "Low-cost INS/GPS integration: concepts and testing," *J. Navigation*, vol. 54, no. 1, pp. 119–134, 2001.
- [18] M. J. Caruso, "Application of magnetoresistive sensors in navigation systems," *Sens. Actuators*, pp. 15–21, 1997.
- [19] S. Y. Cho and C. G. Park, "Modeling and error compensation of walking navigation system," *Trans. Korean Inst. Elect. Eng.*, vol. 51, no. 6, pp. 221–227, 2002.
- [20] C. Drane, M. Macnaughtan, and C. Scott, "Positioning GSM telephones," *IEEE Comm. Mag.*, vol. 36, no. 4, pp. 46–54, Apr. 1998.
- [21] J. Warrior, E. Mchenry, and K. Mcgee, "They know where you are," *IEEE Spectrum*, vol. 40, no. 7, pp. 20–25, Jul. 2003. See also: <http://www.fcc.gov/911/enhanced/>.
- [22] J. Elwell, "Inertial navigation for the urban warrior," *Proc. SPIE*, vol. 3709, pp. 196–204, 1999.
- [23] H. R. Everett, *Sensors for Mobile Robots: Theory and Application*. Wellesley, MA: A. K. Peters, 1995.

- [24] L. Fang, P. Antsaklis, L. Montestruque, M. B. McMickell, M. Lemmon, Y. Sun, H. Fang, I. Koutroulis, M. Haenggi, M. Xie, and X. Xie, "A wireless dead reckoning pedestrian tracking system," in *Proc. WAMES*, Boston, MA, 2004. [Online]. Available: [http://icawwww.epfl.ch/luo/WAMES%202004\\_files/wames\\_WirelessPTS.pdf](http://icawwww.epfl.ch/luo/WAMES%202004_files/wames_WirelessPTS.pdf).
- [25] J. A. Farrell and M. Broth, *The Global Positioning System and Inertial Navigation*. New York: McGraw-Hill, 1999.
- [26] *List of Commercial Sensors, CMU*, [Online]. Available: <http://www.frc.ri.cmu.edu/robotics-faq/10.html#10.1.2>, 1997.
- [27] V. Gabaglio, "Centralized Kalman filter for augmented GPS pedestrian navigation," in *Proc. ION GPS*, Salt Lake City, UT, 2001, pp. 312–318.
- [28] V. Garaj, F. Cecelja, and W. Balachandran, "The brunel navigation system for blind," in *Proc. ION GPS*, Salt Lake City, UT, 2000, pp. 1729–1734.
- [29] G. Gartner, "Location-based mobile pedestrian navigation services—the role of multimedia cartography," in *Proc. UPIMap*, Tokyo, Japan, 2004. [Online]. Available: <http://www.ubimap.net/upimap2004/html/papers/UPIMap04-B-03-Gartner.pdf>.
- [30] G. Gartner, V. Radoczky, and G. Retscher, "Location technologies for pedestrian navigation," *GIS Development*.
- [31] H. Hashimoto, K. Magatani, and K. Yanashima, "The development of the navigation system for visually impaired persons," in *Proc. 23rd Annu. EMBS Int. Conf.*, Istanbul, Turkey, 2001, pp. 1481–1483.
- [32] J. Hightower and G. Borriello, "Location systems for ubiquitous computing," *Computer*, vol. 34, pp. 57–66, 2001.
- [33] J. Hill and D. Culler, "Mica: A wireless platform for deeply embedded networks," *IEEE Micro*, vol. 22, no. 6, pp. 12–24, Nov./Dec. 2002.
- [34] M. Horton, A. Broad, M. Grimmer, K. Pister, S. Sastry, T. Rosenbury, and N. A. Whitaker, "Deployment ready multimode micropower wireless sensor networks for intrusion detection, classification, and tracking," *Proc. SPIE*, vol. 4743, pp. 307–312, 2002.
- [35] M. Hoshino, Y. Gunji, S. Oho, and K. Takano, "A Kalman filter to estimate direction for automotive navigation," in *Proc. IEEE/SICE/RSJ Int. Conf. Multisensor Fusion and Integration for Intelligent Systems*, Washington, DC, 1996, pp. 145–150.
- [36] R. Jirawimut, P. Ptasinski, V. Garaj, F. Cecelja, and W. Balachandran, "A method for dead reckoning parameter correction in pedestrian navigation system," *IEEE Trans. Instrum. Meas.*, vol. 52, no. 1, pp. 209–215, Feb. 2003.
- [37] R. Jirawimut, S. Prakoonwit, F. Cecelja, and W. Balachandran, "Visual odometer for pedestrian navigation," in *IEEE Instrum. Meas. Technology Conf.*, Anchorage, AK, 2002, pp. 43–48.
- [38] C. T. Judd, "Personal dead reckoning module," in *ION GPS*, Kansas City, MO, 1997, pp. 47–51.
- [39] J. Käppi, J. Syrjärinne, and J. Saarinen, "MEMS-IMU based pedestrian navigator for handheld devices," in *ION GPS*, Salt Lake City, UT, 2001, pp. 1369–1373.
- [40] S. Kaluwahandi and Y. Tadokoro, "Portable traveling support system using image processing for the visually impaired," in *Proc. Int. Conf. Image Processing*, Thessaloniki, Greece, 2001, pp. 337–340.
- [41] E. D. Kaplan, Ed., *Understanding GPS: Principles and Applications*. Norwood, MA: Artech House, 1996.
- [42] L. Kay, "A sonar aid to enhance spatial perception of the blind: engineering design and evaluation," *Radio. Electron. Eng.*, vol. 44, pp. 605–627, 1974.
- [43] J. W. Kim, H. J. Jang, D.-H. Hwang, and C. Park, "A step, stride and heading determination for the pedestrian navigation system," *J. Global Positioning Syst.*, vol. 3, no. 1–2, pp. 273–279, 2004.
- [44] Y. Konishi and R. Shibasaki, "Development of an autonomous personal positioning system," in *Proc. Asia GIS*, 2001.
- [45] J. Krumm, S. Harris, B. Meyers, B. Brumitt, M. Hale, and S. Shafer, "Multi-camera multi-person tracking for easy living," in *Proc. IEEE 3rd Int. Workshop on Visual Surveillance*, Dublin, Ireland, 2000, pp. 3–10.
- [46] J. Krumm, L. Williams, and G. Smith, "SmartMoveX on a graph—an inexpensive active badge tracker," in *Proc. UbiComp*, Göteborg, Sweden, 2002, pp. 299–307.
- [47] J. Krumm, "Probabilistic inferencing for location," in *Proc. 2003 UbiComp Workshop on Location-Aware Computing*, Seattle, WA, 2003, pp. 25–27.
- [48] Q. Ladetto, "On foot navigation: continuous step calibration using both complementary recursive prediction and adaptive Kalman filtering," in *Proc. ION GPS*, Salt Lake City, UT, 2000, pp. 1735–1740.
- [49] Q. Ladetto, J. van Seeters, S. Sokolowski, Z. Sagan, and B. Merminod, "Digital magnetic compass and gyroscope for dismounted soldier position and navigation," in *Proc. NATO-RTO Meetings*, Istanbul, Turkey, 2002.
- [50] Q. Ladetto and B. Merminod, "In step with INS," in *GPS World*, Oct. 2002, pp. 30–38.
- [51] S.-W. Lee and K. Mase, "Recognition of walking behaviors for pedestrian navigation," in *Proc. IEEE Conf. Contr. Applicat.*, Mexico City, Sep. 2001, pp. 1152–1155.
- [52] H. S. Lee, K. Mase, T. Adachi, T. Oosawa, K. Nakano, M. Sengoku, H. Hidaka, N. Shinagawa, and T. Kobayashi, "Pedestrian tracking using GPS, pedometer and magnetic compass," *Trans. IEICE-B*, vol. 84-B, no. 12, pp. 2254–2263, 2001.
- [53] K. Legat and W. Lechner, "Navigation systems for pedestrians—a basis for various value-added service," in *ION GPS*, Salt Lake City, UT, 2000, pp. 1713–1720.
- [54] *DMC-SX Digital Magnetic Compass—Operator Manual*, Leica Vectorix AG, Switzerland.
- [55] H. Leppäkoski, J. Syrjärinne, and J. Saarinen, "Personal positioning using wireless assisted GPS with low-cost INS," in *Proc. ION GPS*, Salt Lake City, UT, 2001, pp. 1361–1368.
- [56] H. Leppäkoski, J. Käppi, J. Syrjärinne, and J. Takala, "Error analysis of step length estimation in pedestrian dead reckoning," in *Proc. ION GPS*, Portland, OR, 2002, pp. 1136–1142.
- [57] S. Liu, Z. Zhao, and J. C. Hung, "A high accuracy magnetic heading system composed of fluxgate magnetometers and a microcomputer," in *Proc. IEEE National Aerospace Electronics Conf.*, Dayton, OH, 1989, pp. 148–152.
- [58] J. Lobo, P. Lucas, J. Dias, and A. T. de Almeida, "Inertia navigation system for mobile land vehicles," in *IEEE Int. Symp. Industrial Electronics*, Athens, Greece, 1995, pp. 843–848.
- [59] A. M. Madni, L. A. Wan, and S. Hammons, "A microelectromechanical quartz rotational rate sensor for inertial applications," in *Proc. 1996 IEEE Aerospace Applications Conf.*, Aspen, CO, 1996, pp. 315–332.
- [60] R. Margaria, *Biomechanics and Energetics of Muscular Exercise*. Oxford, U.K.: Clarendon, 1976.
- [61] K. Magatani, K. Sawa, and K. Yanashima, "Developing of the navigation system for the visually impaired by using optical beacons," in *Proc. 23rd Annu. EMBS Int. Conf.*, 2001, pp. 1488–1490.
- [62] D. McErlean and S. Narayanan, "Distributed detection and tracking in sensor networks," in *Proc. Asilomar Conf. Signals, Systems, Computers.*, Pacific Grove, CA, 2002, pp. 1174–1178.
- [63] S. Miyazaki, "Long-term unrestricted measurement of stride length and walking velocity utilizing a piezoelectric gyroscope," *IEEE Trans. Biomed. Eng.*, vol. 44, no. 8, pp. 753–759, Aug. 1997.
- [64] N. Molton, S. Se, J. M. Brady, D. Lee, and P. Probert, "Stereo vision-based aid for the visually impaired," *Image and Vision Computing*, vol. 16, no. 4, pp. 251–263, 1998.
- [65] J. Murray, "Wearable computers in battle: recent advances in the land warrior system," in *Proc. 4th Int. Symp. Wearable Computers.*, Atlanta, GA, 2000, pp. 169–170.
- [66] *Magnetic Declination*, [Online]. Available: <http://www.ngdc.noaa.gov/seg/potfld/declination.shtml>. National Geophysical Data Center.
- [67] R. J. Orr and G. D. Abowd, "The smart floor: a mechanism for natural user identification and tracking," in *Proc. Conf. Human Factors in Computing Systems*, The Hague, The Netherlands, 2000, pp. 1–6.
- [68] H. W. Ott, *Noise Reduction Techniques in Electronic Systems*, 2nd ed. New York: Wiley, 1988, pp. 159–201.
- [69] K. Pahlavan, X. Li, and J.-P. Mäkelä, "Indoor geolocation science and technology," *IEEE Commun. Mag.*, vol. 40, no. 2, pp. 112–118, Feb. 2002.
- [70] S. Park, I. Locher, A. Savvides, M. B. Srivastava, A. Chen, R. Muntz, and S. Yuen, "Design of a wearable sensor badge for smart kindergarten," in *Proc. 6th Int. Symp. Wearable Computers*, Seattle, WA, 2002, pp. 231–238.
- [71] *GPS Pathfinder Pocket Receiver*, [Online]. Available: <http://www.trimble.com/pathfinderpocket.html>, 2003.
- [72] P. Bahl and V. N. Padmanabhan, "RADAR: an in-building RF-based user location and tracking system," in *Proc. IEEE INFOCOM*, 2000, pp. 775–784.
- [73] G. Retscher, "Multisensor systems for pedestrian navigation," in *Proc. ION GNSS*, Long Beach, CA, 2004, pp. 1076–1087.
- [74] R. M. Rogers, "Land vehicle navigation filtering for a GPS/dead-reckoning system," in *Proc. National Technical Meeting, Institute of Navigation*, Santa Monica, CA, 1997, pp. 703–708.
- [75] C. F. Olson, L. H. Matthies, M. Schoppers, and M. W. Maimone, "Stereo ego-motion improvements for robust rover navigation," in *Proc. ICRA*, Seoul, Korea, 2001, pp. 1099–1104.
- [76] S. I. Roumeliotis, A. E. Johnson, and J. F. Montgomery, "Augmenting inertial navigation with image-based motion estimation," in *Proc. ICRA*, Washington, DC, 2002, pp. 4326–4333.

- [77] W. Rungtanyotin and T. Starner, "Finding location using omnidirectional video on a wearable computing platform," in *4th Int. Symp. Wearable Computers*, Atlanta, GA, 2000, pp. 61–68.
- [78] L. Russell, "Travel path sounder," in *Proc. Rotterdam Mobility Res. Conf.*, New York, 1965. Amer. Foundation for the Blind.
- [79] K. Sagawa, H. Inooka, and Y. Satoh, "Non-restrictive measurement of walking distance," in *IEEE Int. Conf. SMC*, 2000, pp. 1847–1852.
- [80] S. Se and M. Brady, "Vision-based detection of kerbs and steps," in *Proc. 8th British Machine Vision Conf.*, Colchester, U.K., 1997, pp. 410–419.
- [81] S. Se, "Zebra-crossing detection for the partially sighted," in *Proc. IEEE Conf. Computer Vision and Pattern Recognition*, Hilton Head Island, SC, 2000, pp. 2211–2217.
- [82] M. Sekine, T. Tamura, T. Togawa, and Y. Fukui, "Classification of waist-acceleration signals in a continuous walking record," *Med. Eng. Phys.*, vol. 22, no. 4, pp. 285–291, 2000.
- [83] S. Shoval, J. Borenstein, and Y. Koren, "The Navbelt—a computerized travel aid for the blind based on mobile robotics technology," *IEEE Trans. Biomed. Eng.*, vol. 45, no. 11, pp. 1376–1386, Nov. 1998.
- [84] W. Soehren and C. Keyes, "Human-motion based-navigation algorithm development," in *Proc. IEEE Position Location and Navig. Symp.*, San Diego, CA, 2000.
- [85] C. Song, B. Ha, and S. Lee, "Micromachined inertial sensors," in *Proc. IEEE/RSJ IROS*, Kyongju, South Korea, 1999, pp. 1049–1056.
- [86] V. Stanford, "Using pervasive computing to deliver elder care," *IEEE Pervasive Computing*, vol. 1, no. 1, pp. 10–13, Jan./Mar. 2002.
- [87] T. Starner, D. Kirsch, and S. Assefa, "The locust swarm: an environmentally-powered, networkless location and messaging system," in *First Int. Symp. Wearable Computers*, 1997, pp. 169–170.
- [88] T. E. Starner, "Wearable Computing and Contextual Awareness," Ph.D. thesis, School Architecture and Planning, Massachusetts Inst. Technol., MA, 1999.
- [89] D. J. Stilwell, C. E. Wick, and B. E. Bishop, "Small inertial sensors for a miniature autonomous underwater vehicle," in *Proc. IEEE Int. Conf. Control Applications*, Mexico City, Mexico, 2001, pp. 841–846.
- [90] *Electronic Compass Design Using KMZ51 and KMZ52*, [Online]. Available: [http://www.semiconductors.philips.com/acrobat\\_download/applicationnotes/AN00022\\_COMPASS.pdf](http://www.semiconductors.philips.com/acrobat_download/applicationnotes/AN00022_COMPASS.pdf).
- [91] T. Takahashi, N. Yamanaka, H. Osawa, K. Sagawa, and H. Inooka, "A long-term ambulatory monitoring system for walking behaviors of a human," *Proc. SPIE*, vol. 2101, pp. 36–41, 1993.
- [92] *TinyOS—A Component-Based OS for Networked Sensor Regime*, [Online]. Available: <http://webs.cs.berkeley.edu/tos/>.
- [93] Y. K. Thong, M. S. Woolfson, J. A. Crowe, B. R. Hayes-Gill, and R. E. Challis, "Dependence of inertial measurements of distance on accelerometer noise," *Meas. Sci. Technol.*, vol. 13, no. 8, pp. 1163–1172, 2002.
- [94] *AnyPhone, AnyWhere—Wireless Location Technology*, [Online]. Available: <http://www.trueposition.com>, 2003.
- [95] C. Verplaetse, "Inertial proprioceptive devices: self-motion-sensing toys and tools," *IBM Syst. J.*, vol. 35, no. 3/4, pp. 647–649, 1996.
- [96] R. Want, A. Hopper, V. Falcao, and J. Gibbons, "The active badge location system," *ACM Trans. Info. Syst.*, vol. 10, no. 1, pp. 91–102, 1992.
- [97] J. L. Weston and D. H. Titterton, "Modern inertial navigation technology and its application," *Electron. Commun. Eng. J.*, vol. 12, no. 2, pp. 49–64, 2000.
- [98] D. R. J. White, *Electromagnetic Shielding Materials and Performance*, 2nd ed. Warrenton, VA: Don White Consultants, 1980.
- [99] C. E. White, D. Bernstein, and A. L. Kornhauser, "Some map matching algorithms for personal navigation assistants," *Transport. Res. Part C*, pp. 91–108, 2000.
- [100] S. You, U. Neumann, and R. Azuma, "Hybrid inertial and vision tracking for augmented reality registration," in *Proc. IEEE Virtual Reality*, Houston, TX, 1999, pp. 260–267.



**Lei Fang** received the B.S. degree in electrical engineering from Zhejiang University, Hangzhou, China, in 1996, and the M.S. degree in electrical engineering from the University of Macau, Macau, in 1999. He is currently working toward the Ph.D. degree in the Department of Electrical Engineering, University of Notre Dame, Notre Dame, IN.

His research interests include decentralized control of multiagent systems under information flow constraints, sensor networks, and optimization theory and algorithms and their applications to

control and manufacturing systems.



**Panos J. Antsaklis** (S'74–M'76–SM'86–F'91) is the Brosey Professor of Electrical Engineering and Director of the Center for Applied Mathematics at Notre Dame. His work includes analysis of behavior and design of control strategies for complex autonomous, intelligent systems. His recent research focuses on networked embedded systems and addresses problems in the interdisciplinary research area of control, computing, and communication networks, and on hybrid and discrete-event dynamical systems. He has authored a number of publications in journals, conference proceedings, and books, and he has edited several books on intelligent autonomous control and on hybrid systems. In addition, he has coauthored the research monograph "Supervisory control of discrete event systems using petri nets" (with J. Moody) and the graduate textbook "Linear Systems" (with A.N. Michel).

Prof. Antsaklis has been a Guest Editor of special issues in IEEE TRANSACTIONS ON AUTOMATED CONTROL and the PROCEEDINGS OF THE IEEE. He has served as Chair of major systems and control conferences and he was the 1997 President of the IEEE Control Systems Society (CSS). He is a recipient of the IEEE Distinguished Member Award of CSS, and Chair of the Awards committee of CSS and as AEAL of the IEEE TAC.



**Luis A. Montestruque** (M'04) received the B.S. degree in electrical engineering from the Pontifical Catholic University of Peru, Lima, Peru, in 1997, the M.S. degree in electrical engineering from the University of Notre Dame, Notre Dame, IN, in 2001, and the Ph.D. degree in 2005.

He is currently the President of EmNet, LLC, Granger, IN, a company specializing in the design and implementation of wireless networked control systems. Since 1995, he has worked in the design and implementation of industrial, environmental, and military-oriented networked embedded systems. His research interests include networked control and sensing systems and real-time embedded systems.



**M. Brett McMickell** received the B.S. degree in mechanical engineering from the University of Utah, Salt Lake City, UT, and the M.S. and Ph.D. degrees from the University of Notre Dame, Notre Dame, IN.

He is currently with Honeywell DSES, Glendale, AZ. His current interests are in vibration control systems, attitude control systems, and symmetry analysis.



**Michael Lemmon** received the Ph.D. degree in electrical and computer engineering from Carnegie Mellon University, Pittsburgh, PA, in 1990.

He is a Professor of electrical engineering at the University of Notre Dame, Notre Dame, IN. His primary research interests are in the area of control theory, including hybrid control systems and supervisory control of discrete-event systems. He is currently studying decentralized control, detection, and estimation on ad hoc sensor-actuator networks.

Dr. Lemmon is a former Associate Editor for the

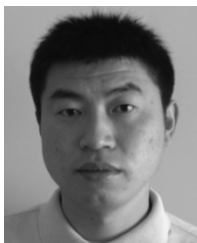
IEEE TRANSACTIONS ON CONTROL SYSTEMS TECHNOLOGY.





**Yashan Sun** received the B.S. and M.S. degree in electrical engineering from China. She is currently working toward the Ph.D. degree in electrical engineering at the University of Notre Dame, Notre Dame, IN.

Her research interests include estimation and formation control problems for multiple robotic vehicle systems.



**Hui Fang** was born in Fuyang, Anhui Province, China, in 1978. He received the B.S. degree from the Department of Automation, University of Science and Technology of China, Hefei, China, in 2001, and the M.S. degree in electrical engineering from University of Notre Dame, Notre Dame, IN, where he is now working toward the Ph.D. degree in the area of distributed control systems and networked control system under communication constraints.



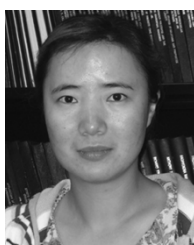
**Ioannis Koutroulis** received the B.S. degree in electrical and computer engineering from University of Patras, Patras, Greece, in 2002, and the M.S. degree in electrical engineering from the University of Notre Dame, Notre Dame, IN, in 2004. He is currently on a leave of absence from the University of Notre Dame. His research and professional interests include networked control combined with wireless communications and web technologies.



**Martin Haenggi** (S'95–M'99–SM'04) received the Dipl.Ing. (M.Sc.) degree in electrical engineering, the Dipl.NDS ETH (post-diploma) degree in information technology, and the Ph.D. degree in analysis, design, and optimization of cellular neural networks from the Swiss Federal Institute of Technology (ETHZ), Zurich, Switzerland, in 1995, 1996, and 1999, respectively.

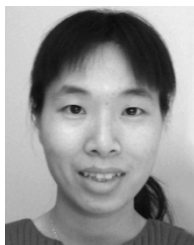
In 1995, he joined the Signal and Information Processing Laboratory at ETHZ as a Teaching and Research Assistant. After a postdoctoral year at the Electronics Research Laboratory, University of California, Berkeley, he joined the faculty of the Electrical Engineering Department, University of Notre Dame, Notre Dame, IN, as an Assistant Professor in January 2001. His scientific interests include networking and wireless communications, with an emphasis on ad hoc and sensor networks.

Prof. Haenggi was awarded the ETH medal for both his M.Sc. and his Ph.D. theses, and he received an NSF CAREER Award in 2005. He is a Distinguished Lecturer of the IEEE Circuits and Systems Society in 2005/06.



**Min Xie** received the B.S. and M.S. degrees in electrical engineering from the Xidian University, Xi'an, China, in 1996 and 1999, respectively, and M.S. degree in electrical engineering from the National University of Singapore, Singapore, in 2001. She is currently working toward the Ph.D. degree in electrical engineering at the University of Notre Dame, Notre Dame, IN.

Her research interests include the QoS guarantees and performance analysis of wireless networks.



**Xiaojuan Xie** was born in Hunan, China, in 1975. She received the M.S. degree in electrical engineering from the Chinese Academy of Science, Beijing, China, in 2000. She is currently working toward the Ph.D. degree in electrical engineering at the University of Notre Dame, Notre Dame, IN.

From 2000 to 2003, she worked as a Software Engineer at Huawei Technologies, Shenzhen, China. Her current research interest is in the area of networked control systems.