CS 525W: Mobile *Ubiquitous* Computing and Wireless Networking

Emmanuel Agu
A Little about me

• Faculty in WPI Computer Science
• **Research interests:** graphics, mobile computing/wireless and mobile graphics
• How did I get into mobile computing + wireless?
  – 3 years in wireless LAN lab (*pre 802.11*)
  – Designed, simulated, implemented wireless protocols
  – Group built working wireless LAN testbed (*pre 802.11*)
• Computer Systems/Electrical/Computer Science background
  • Hardware + software
About this class (Administrivia)

• **Class goal:** provide overview, insight into hot topics, ideas and issues in mobile ubiquitous computing and wireless networking
• Full course name: *Mobile Ubiquitous Computing and Wireless Networking*
• Meet for 14 weeks, break on March 8 (term break)
• Seminar style: I will present, **YOU** will present papers
• See big picture through focussed discussions
• Check for papers on course website: [http://web.cs.wpi.edu/~emmanuel/courses/cs525m/S11/](http://web.cs.wpi.edu/~emmanuel/courses/cs525m/S11/)
• **Projects:** 1 or 2 assigned, 1 big final project
• This area combines lots of other areas: (networking, OS, software, machine learning, etc): Most people don’t have all the background!!
  • Projects: Make sure your team has requisite skills
Administrivia: Papers

- **Weeks 1 and 2**: I will present
- **Weeks 2 – 12**: You will present + I will present
  - I will present background material on the week’s topic
  - 3 student presentations from Required Papers for the week
- Student presentations: ~30 mins + ~10 mins discussion
- 15-min break halfway through each day
**Formal Requirements**

- **What do you have to do to get a grade?**
- Seminar: Come to class + Discuss!! Discuss!! Discuss!!
- Present 2 or 3 papers
- Email me 1-page summaries (in ASCII text) for weekly papers
- Do assigned project(s)
- Do term project: 5-phases
  - Pick partner + decide project area
  - Submit intro + related work
  - Propose project plan
  - Build, evaluate, experiment, analyze results
  - Present results + submit final paper (in week 14)
- Grading policy: Presentation(s): 30%, Class participation: 10%, Final project: 50%, Summaries: 10%.
Written Summaries

- Email to me before class in ASCII text. No Word, Latex, etc.
- Summarize key points of all 3 papers for week
  - Main contributions
  - Limitations of the work
  - What you like/not like about paper
  - Any project ideas?
- 20 sentences max per paper
- Summary is quick refresh in even 1 year’s time
  - Include main ideas/algorithms, results, etc.
- See handout for more details
Students: Please Introduce Yourselves!

• Name
• Status: grad/undergrad, year
• Relevant background: e.g. coal miner 😊
• Relevant courses taken:
  • *Systems*: Networks, OS,
  • *Advanced*: machine learning, advanced networks, etc
• What you would like to get out of this class?
  – Understanding a hot field
  – Just a class for masters degree/PhD
  – Compliments your research interests/publications
  – My spouse told me to 😊
Next… Overview

- Brief overview of topics/issues
- Define/motivate area, excite (or discourage) you
- Provoke thinking: More questions, problems than solutions
- Sample of topics to be covered in class
- ALL topics covered in more detail later
- Students may only understand part of topics in today’s overview
Mobile computing

- Mark Weiser, Xerox PARC CTO
- 1991, articulated vision (and issues) for ubiquitous mobile computing
- **Weiser’s Vision:**
  
  “Environment saturated with computing and communication capabilities, with humans gracefully integrated”

- **Core idea:** Invisible hardware/software that assist human
  - **Hardware:** smart phones, sensors, tablets, wearable devices, etc
  - **Software:** Voice recognition, Mobile OS, Networking/communication software, protocols, etc
- Weiser’s vision ahead of its time, available hardware and software
- Example: voice recognition was not available then
- Today, envisioned hardware and software is available
Mobile vs Ubiquitous Computing

• Mobile computing
  • deals mostly with *passive* network components
  • Human simply provided universal, seamless network connectivity
  • Human does all the work, initiates all activity, network traffic!!
  • Example: Using *foursquare.com* on smart phone

• Ubiquitous computing
  • introduces collection of specialized assistants to assist human in tasks (reminders, personal assistant, staying healthy, school, etc)
  • Networked array of active elements, sensors, software agents, artificial intelligence
  • Builds on *distributed systems* and *mobile computing* (more later)
Ubicomp Sensing

• Sense what?
  – *Human*: motion, mood, identity, gesture
  – *Environmental*: temperature, sound, humidity, location
  – *Ubicomp example*:
    • *Assistant senses*: Temperature outside is 10F (environment sensing) + Human plans to go work (schedule)
    • *Assistant advise*: Dress warm!
  • Sensed environment + Human + Computer resources = **Context**
  • **Context-Aware** applications adapt their behavior to **context**
Sensing the Human

- Environmental sensing is relatively straight-forward to integrate
- Human sensing is a little harder (ranked easy to hard problems)
  - **Where**: location (easiest):
  - **Who**: Identification
  - **How**: (Mood) happy, sad, bored (gesture recognition)
  - **What**: eating, cooking (meta task)
  - **Why**: reason for actions (extremely hard!)
- Human sensing (gesture, mood, etc) easier with cameras than sensors
- Research in ubiquitous smart environments (office, kindergarten) integrates location sensing, user identification, emotion sensing, gesture recognition, activity sensing, user intent
Mobile Devices

- Smart phones (Blackberry, iPhone, Android, etc)
- Personal Digital Assistants (PDAs)
- Tablets (iPad, etc)
- Laptops
Mobile Devices: Droid

- This class: Google Droid as main mobile device
- Google donated Motorola Droid smart phones
- One assigned project and final project based on Droid
  - Connects to Verizon network, WLAN or Bluetooth
  - Google Android OS
  - 5 MegaPixel camera
  - Streaming video: mpeg, H.264
  - GPS, google maps, etc
  - Sensors: accelerometer, proximity
eCompass, ambient light
Sensor Node

- Sensor? Think of automatic doors
- Automatic door sensor has single purpose: detect human
- New multi-functional sensors, programmable for various tasks (intrusion detection, temperature, humidity, pressure, etc)
- Low cost ($1 per sensor), 1000’s per room, attach to objects
- Capabilities: Sense, process data, communicate with sink node
- Constraints: Small CPU, OS, programmable

(courtesy of MANTIS project, U. of Colorado)   RFID tags   Tiny Mote Sensor, UC Berkeley
Wireless Sensors for Environment Monitoring

- Embedded in room/environment
- Many sensors cooperate/communicate to perform task
- Monitors conditions (temperature, humidity, etc)
- User can query sensor (What is temp at sensor location?)
remote patient monitoring

Jovanov et. al, “Stress Monitoring...”
IEEE Engineering in Medicine and Biology Mag, May/June 2003
Explosion of Devices

- **Recent Nokia quote**: More cell phones than tooth brushes
- Many more sensors envisaged
- **Ubiquitous computing**: Many computers per person

The Major Trends in Computing
Worldwide cellular subscriber growth

Global Subscriber Base (in million)

Source: Informa
**Definitions: Portable, mobile & ubiquitous computing**

- **Distributed computing:** system is physically distributed. User can access system/network from various points. E.g. Unix, WWW. (huge 70’s revolution)

- **Portable (nomadic) computing:** user intermittently changes point of attachment, disrupts or shuts down network activities

- **Mobile computing:** continuous access, automatic reconnection

- **Ubiquitous (or pervasive) computing:** computing environment including sensors, cameras and integrated active elements that cooperate to help user

- Class concerned mostly with last 2 (mobile and ubiquitous)
Distributed Computing

- **Distributed computing example**: You, logging in and web surfing from different terminals on campus. Each web page consists of hypertext, pictures, movies and elements anywhere on the internet.
- **Note**: network is fixed, YOU move
- **Issues**:
  - Remote communication (RPC),
  - Fault tolerance,
  - Availability (mirrored servers, etc)
  - Caching (for performance)
  - Distributed file systems (e.g. Network File System (NFS))
  - Security (Password control, authentication, encryption)
Nomadic computing

- Nomadic computing… Nomads… ?
Nomadic Computing

- **Portable (nomadic) computing example:** I own a laptop. Plugs into my home network, sit on couch, surf web while watching TV. In the morning, wake up, un-plug, shut down, bring laptop to school, plug into WPI network, start up!
- **Note:** Network is fixed, device moves and changes point of attachment.
- **Issues:**
  - File/data pre-fetching
  - Caching (to simulate availability)
  - Update policies
  - Re-integration and consistency models
  - Operation queuing (e.g. emails while disconnected)
  - Resource discovery (closest printer while at home is not closest printer while at WPI)
- **Note:** much of the adaptation in “middleware” layer
Mobile Computing Example

• **Mobile computing**: Sarah owns SPRINT PCS phone with web access, voice, SMS messaging and can run apps like facebook and foursquare. She remains connected while she drives from Worcester, Massachusetts to Compton, California.

• **Note**: Network topology changes, because sarah and mobile users move. Network deals with changing node location.

• **Issues**
  – Mobile networking (mobile IP, TCP performance)
  – Mobile information access (bandwidth adaptive)
  – System-level energy savings (variable CPU speed, hard disk spin-down, voltage scaling)
  – Adaptive applications: (transcoding proxies, adaptive resource management)
  – Location sensing
  – Resource discovery (e.g. print to closest printer)
Ubiquitous Computing Example

• **Ubiquitous computing:** John is leaving home to go and meet his friends. While passing the fridge, the fridge sends a message to his shoe that milk is almost finished. When John is passing grocery store, shoe sends message to glasses which displays “BUY milk” message. John buys milk, goes home.

• **Core idea:** ubiquitous computing assistants that help John

• **Issues:**
  – Sensor design (miniaturization, low cost)
  – Smart spaces
  – Invisibility (room million sensors, minimal user distraction)
  – Localized scalability (more distant, less communication)
  – Uneven conditioning
  – Context-awareness (assist user based on her current situation)
  – Cyber-foraging (servers augment mobile device)
  – Self-configuring networks
Summary/Relationships

• Systems perspective: nomadic and mobile are reactive, ubiquitous is proactive

• Distributed systems + mobile computing research issues = mobile computing

• Mobile computing + pervasive computing issues = pervasive computing

• In this class, first part will be mobile/nomadic computing, then ubiquitous computing part
Typical of Ubicomp App

Generic:

Gather sensor data → Process sensor data (Intelligence) → Assist User (Output)
Location-aware mobile computing apps

- Focus mostly on mobile and ubiquitous computing apps that use Smart Phone and Internet connectivity.

- Example: Location-aware mobile computing apps. Issues:
  - **Entropy**: Inferring how close two Facebook friends are based on locations mutually visited
  - May not want all Facebook friends to know exactly where I am
  - Automatically **anonymize location info**
    - **Fact**: User is at Starbucks, 180 Main St, Worcester, MA
    - Status update to friend A: Emmanuel is at “coffee shop”
    - Status update friend B: Emmanuel is at “Starbucks, 180 Main St, Worcester”
    - Algorithms to automatically generate status update (based on closeness)
The Internet as a data source for Location-aware apps

[Identifying the Activities Supported by Locations with Community-Authored Content, Dearman and Truong, Univ. of Toronto]

- User at location X would like to make location-based queries
  - What activities can I do here?
  - What’s a good close place to do X activity (e.g. soccer)
- Solution: Yelp is a community-authored reviewer website for restaurants, activities, etc
- Yelp has: activities + location + goodness of venues
- Scrape + mine yelp: augment with location as searchable tag
Location-Aware Apps

- Easier location check-in
  - Ubicomp 2010 video p395
Context-Aware Search

[ Hapori: Context-based Local Search for Mobile Phones using Community Behavioral Modeling and Similarity, Nicholas D. Lane, Dartmouth College]

- **Goal**: Improves Internet search results using context, such as weather, age, profile of user, time, location and profile of other users to improve search.
- Example: a teenager gets a completely different set of recommendations from an elder.
Mobile Social Networking

- Partipatory sensing: Many people cooperating on a task
- **Classic example:** Comparative shopping
- At CVS, ready to buy toothpaste. Is CVS price the best locally?
- Phone has software to query other members of my network
- People at other local stores (Walmart, Walgreens, etc) respond with prices
UCLA Participatory Sensing Video

- Demo from UCLA
Mobile Social Networking

• Smart phones have many sensors, cameras, etc
• Imagine ability to access other people’s phones: *Phone Sensing*
• Like a telescopic lens into different locations: *Microblogging*
Sensing Human Behavior

[Social Sensing for Epidemiological Behavior Change, Anmol Madan et al, MIT Media Lab]

- **Examples**: Changes in call rates or visiting low entropy places more could mean person is sick
- Statistics of number of calls, co-location, proximity, WLAN and bluetooth entropy found to be good predictors of illness.
- Findings could be used as an early warning tool.
- If strong inference, then nurse could call the person
Energy Efficiency

• Most resources increasing exponentially except battery energy (ref. Starner, IEEE Pervasive Computing, Dec 2003)

• Strategies:
  • Energy harvesting: Energy from vibrations, moving humans
  • Scale down: Reduce image, video resolutions to save energy
  • Better user interface: Estimate and inform user how long each potential task will take
    • E.g: At current battery level, you can either type your paper for 45 mins, watch video for 20 mins, etc
Developed countries (e.g. US, UK) have 4 main wide area telecommunications networks (or backbones)
  – Internet
  – Telephone
  – Cable television
  – Cellular phone
Most are hierarchical: divided into **backbone** and **local loop**
Only some of these wide area networks in developing nations?
Internet is main computing backbone
• **Characteristics of Web Content** by Timmins et al
  - Formats, sizes, etc of mobile web pages
• **Haggle: Seamless Networking for Mobile Applications** by Su et al
  - Framework that manages various available networks, speeds, etc for user
• **A First Look at Traffic on Smartphones** Hossein Falaki et al
  - Analysis of measured smart phone traffic

• Goal: Activity detection around the home
• Many new appliances have a “soft switch”
• Proposed a sensor for homes, plugged into **single point**:
  • Train first: captures electric signature of each appliance in home
  • Can then detect device when appliance turned on in future
• Appliance signature was unique and usable at different time home
  E.g: iMac signature is unique. Capture once, use many times
Energy efficiency

• Smart home: energy efficiency
  • Ubicomp 2010 video p361
Wireless Networks Types

- **Cellular Network:** Wide area wireless network operated by Sprint, Verizon, AT&T, etc. 1G (analog), 2G today’s network, 3G coming, 4G (in some labs)

- **WLANs:**
  - **Infrastructure networks:** wired backbone (Internet), wireless last hop. E.g WPI wireless LAN, **New:** mesh networks
  - **Ad hoc networks:** all wireless, no backbone, no order known in advance. E.g. few deployed examples.. .futuristic

- **Bluetooth:** Short range communications, printers, headsets, etc
- **WiMax:** Wide area high bandwidth

- **Sensor networks:** self-organizing network of large numbers of cooperating sensors deployed inside phenomenon. E.g. even more futuristic. Many research projects
Wireless systems: evolution

1981: NMT 450
1986: NMT 900
1982: AMPS
1983: Inmarsat-A
1988: Inmarsat-C
1992: CDMA
1991: D-AMPS
1991: PDC
1992: Inmarsat-B
1992: Inmarsat-M
1998: Iridium
2000: GPRS
2001: IMT-2000

analogue

digital

cellular phones

1980: CT0
1984: CT1
1987: CT1+
1989: CT 2
1991: DECT

cordless phones

1991: D-AMPS
1991: CDMA
1993: PDC
1997: 802.11, Bluetooth
1999: 802.11a
199x: proprietary
1997: IEEE 802.11
2000: IEEE 802.11a

wireless LAN

1980: CT0

satellites

1984: Inmarsat-C
1988: Inmarsat-C
1991: DECT

4G – fourth generation: when and how?

Ref: Mobile Communications, 2nd edition
Wireless Networking Challenges

- Wireless networking issues
  - Wireless spectrum scarcity (regulated)
    - Low bandwidth, asymmetric, heterogeneous
  - Higher error rates ($10^{-3}$):
    - multipath fading, noise (engines, microwaves), echos...
    - Note: indoor channel is different from outdoor
  - Higher delays, higher jitter
    - Connection time: secs for GSM, > 0.1s other wireless
  - Moving users:
    - Uncontrolled cell population, variable link quality
    - Different points of attachment to network
    - Frequent network disconnections (cell phone)
Wireless Networking Challenges

• Wireless networking issues (contd)
  – Less secure and less robust
    • (e.g. signal leakage)
    • More easily stolen, tampered with (drunk employees)
  – Shared medium
    • Who’s turn to transmit, etc
  – Tough to guarantee Quality of Service (QoS)
Wireless Measurement

- Previous versions of class covered wireless protocols, standards
- This version: brief coverage on wireless
  - Usage: measurement studies of wireless LANs and mobile web, wireless mesh networks, etc
  - Programmer perspectives: How to program Android apps for wireless (WLAN, bluetooth, cellular) connectivity
  - Novel wireless frameworks for ubicomp, seamless communications during roaming
Wireless Security

- Wireless signals leak beyond building confines
- Mobile devices designed to be carried around => more prone to theft or misplacement
- Mobility: tracking perpetuators is hard
- Security standards like Wireless Encryption Protocol (WEP) have significant demonstrated flaws
- Anderson: over 90% of security breaches caused by lapses in physical security:
  - **Example:** drunk employee at bar with laptop
WLAN Vulnerabilities

- Protocol (e.g. 802.11) vulnerabilities:
  - **Rogue APs:** Attacker inserts access point, hijacks mobile nodes
  - **Jamming:** ISM bands prone to that, microwaves, etc
  - **Induce congestions, collisions:** Induce collisions, congestion, disobey protocol. Delay bad for multimedia
  - **Exhaustion:** Keep sending packets to wireless node, prevent sleep modes, drain battery, DoS
  - **Packet header manipulation:** e.g. sequence/ACK Nos.
Wi-Fi Privacy Ticker

[Sunny Consolvo et al., Intel Labs Seattle, University of Washington]

• Many wireless security/privacy breaches occur
• Many open problems. Some too hard to solve for now
• Examples:
  – website A may send your information to website B without your knowledge
  – New google search sends typed characters BEFORE you hit enter
• **Solution:** Alert to user when info is being transmitted unsecurely
• Ticker streams violations of user's pre-defined breaches
• “Breach“ identified and importance customizable
• Wi-Fi Ticker increased user awareness about security
• Even highly techno-savvy learned about breaches
Final Words

- This is a **special topics** graduate class
- **Special Topics**: I have picked selected topics that are hot.
- Coverage is not complete
- Graduate class so graduate level work/effort is expected
- Seminar style classes: You get out what you put into them
Homework

• Today: Sign up for papers to present
  • Procedure: Sign up sheet passed around, simply sign
• Summaries of week 2 papers (Smart homes and healthcare): due before next class
• Two weeks: decide project area and partners (if any)
  • Project? Never too early to start thinking about project, talking to me.