Context-based simulator for sensor networks in domestic environments

Markus Huebscher
Imperial College London
Outline

- Introduction
  - Smart homes
  - Types of sensors
- Wireless Sensor Networks
  - UC Berkeley motes and TinyOS
- Simulating sensors
  - Based on high-level description of scenarios
  - Implementation example: TinyOS motes
Outline

- Introduction
  - Smart homes
  - Types of sensors
- Wireless Sensor Networks
- Simulating sensors
  - Based on high-level description of scenarios
  - Implementation example: TinyOS motes
Smart Homes

- Augment home with sensors
  - Support individuals with health problems
    - Elderly living alone at home
    - Patients with heart condition
  - Improve lifestyle
    - Smart fridge keeps track of items in fridge
    - Smart phones ring only in room where addressee is present
Types of sensors

- Video cameras
- Location
- Light
- Sound
- Temperature, humidity
- Pressure
- Motion detection
- Acceleration
Types of sensors

- Video cameras
- Location
- Light
- Sound
- Temperature, humidity
- Pressure
- Motion detection
- Acceleration

Wireless sensor nodes
Wireless sensor nodes

- Resource-constrained devices with
  - Processor
  - RAM, EEPROM
  - Radio transmitter/receiver
  - ADC and DIO ports for sensors
  - Battery with limited power
Wireless sensor nodes: Example

- UC Berkeley mote (MICA2)
  - Processor: 8bit 20MHz microcontroller
  - Program flash memory: 128KB
  - RAM: 512KB
  - EEPROM: 4KB
  - Radio: ISM Bands, ~40Kbps
  - 10 bit ADC ports + DIO
  - Battery life depends mainly on communication
Using sensor data in context-aware applications

- Adaptation
  - e.g. check that pulse rate low.

- Context
  - e.g. user relaxing and reading a book.
  - e.g. user in living room sitting on sofa, room bright, no noise, etc.

- Interpret
  - e.g. user at location (x,y), light and sound values, etc.

- Collect
- Aggregate

- Sensors
  - Simulate
Why simulate?

- Controlled environment with reproducible sensor data.
- Physical limitations of available space and sensors (cost).
- Testing context-aware application with real-life sensor data is tedious.
Goals of simulation

- Produce “virtual” sensor data such that application logic can be adequately tested.
- Simulator invisible to application.
  - Application can run on simulator without modifications.
- Simulator configuration input are high-level.
  - People’s activities (context).
Simulating data from context

Adaptation

Context

Interpret

Collect

Aggregate

Sensors
Simulating data from context

Adaptation → Context → Interpret → Collect → Aggregate → Sensors

Context → Interpret → Distribute → Abstract sensors → Map to ADC values → Sensors

e.g. user relaxing and reading a book.

e.g. user in living room sitting on sofa, room bright, no noise, etc.

e.g. pressure on sofa pressure sensors in Newton, quiet noise level

e.g. 10-bit values on ADC ports of motes
How simulate

- Separate home topology from activity scenarios

John Doe
- Sleep
- Get up
- Brush teeth
- Get Dressed
- …

Allows people’s activities scenarios to be executed on different home topologies
Home topology

- 2D map with
  - Walls/doors
    - Determine valid paths of people
    - Affect detection range of sensors
  - Sensors
    - Detect people, e.g. motion detection, sound, pressure sensors in chairs/bed.
  - Environment
  - Rooms
    - Areas with a context, e.g. bedroom, bathroom
  - Hotspots
    - Locations where activities take place.
Activities description

- Defines people
- Defines their activities
  - Each activity takes place at a Hotspot
    - Hotspot is only a well-defined location given an instance of a home topology
    - An activity usually has a default hotspot, e.g. “Brushing teeth” takes place at wash basin in bathroom
  - Executing an activity causes the person to move from the previous Hotspot to current Hotspot
  - Activity has duration
  - Activity determines sensor data produced
Sensor data

- Sensors produce data knowing
  - Complete home topology
  - Complete schedule of activities of people
  - Current location of people in home
    - A location also has a context, e.g. bedroom
  - Current activity of people
    - Hotspot location
    - Duration of activity
Implementation example

- UC Berkeley motes
  - Support most types of sensors
  - Run TinyOS (component-based application programming environment)
  - TinyOS Simulator: TOSSIM
TOSSIM

- Simulates a network of motes
- Bit-level simulation of motes’ operation and communication
  - Too low-level to implement activities scenarios

✓ TOSSIM execution can be “steered” by an external application
Motes

Application

serial link

gateway mote

Sensor Network
TOSSIM

Application

serial link
gateway mote

Sensor Network

Application

serial link

TOSSIM

Virtual Sensor Network
Context-level simulation

Application

Serial link

Gateway mote

Sensor Network

Application

Serial link

TOSSIM

Virtual Sensor Network

TCP

Events

Socket

Commands

Socket

Context-Aware Simulator
TOSSIM ports

- Event port
  - Notifies external app of execution events

- Command port
  - Set sensor data read on motes
  - Change location of sensors
  - Activate/Deactivate motes (simulate failures/upgrades)
Conclusions

- Need to simulate sensor data involved in smart home scenarios
- Existing simulators too low-level
- Our simulation framework allows configuration at activity context level
- TinyOS example: simulator invisible to application
The End