



---

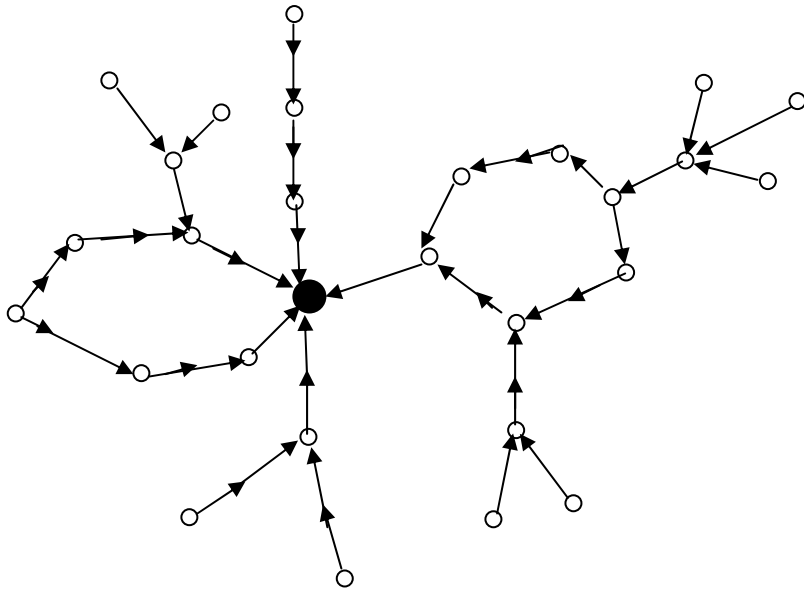
# **A PROBLEM OF SENSING IN A NETWORK ENVIRONMENT**

**Anthony Ephremides**  
**University of Maryland**

**Netted Sensors Community Workshop**

**The Mitre Corporation**  
**Mclean, Virginia**  
**October 25, 2005**

# The Problem



“it’s not an insect”

**Objective**: deploy for the purpose of deciding presence of an object of interest and providing this functionality for as long as possible

**Translation**: Design and Operate this Sensor Network for:

- (i) Maximum Probability of Correct Detection
- (ii) Maximum Lifetime



# Discussion

---

- Application-Specific Overriding Criterion (ASOC)
- Energy Efficiency

➔ These characteristics are shared by any wireless sensor network ←

- Hence: Cross-Layer Design  
with
  - (i) Application-specific character
  - (ii) Holistic Layer-space (i.e - from the application layer down to below the physical layer, namely the hardware)



# Network Issues

---

- Neighbor Discovery
- Measurement specification and processing
- MAC/interference control
- Routing

(always so as to max P [correct detection] and min Energy)



# Routing

---

- Need Link Metric
- How to map Performance Measures to a Link Metric
  - Energy
    - transmission power
    - residual batter level
  - Detection Probability?
    - akin to blocking probability in circuit switched networks
    - recent result:
      - distance (the farther the better!)
      - intuitively sensible
      - Markovian Signal Model (with L. Tong, et. al.)



# Energy Trade-Off

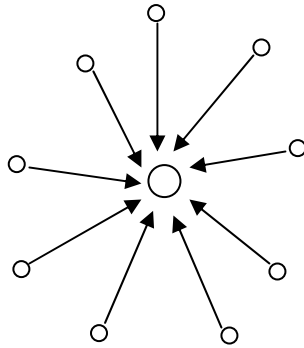
---

- Transmission vs. Processing
  - Lore (as usual) inaccurate
- Processing includes:
  - Being “ON”
  - “Receiving”
  - Local Manipulation
    - memory management
    - compressing
    - Etc.

Hence: HARDWARE becomes important

- HPA, antennas, etc
- Embedded Processor

# Some Past Work (with L. Yu)



- For simplified case (no routing):
  - Sequential Detection requires (on average) significantly fewer measurements than fixed detection schemes
  - Comparison between extremes
    - single-bit transmissions
    - full data transmissions
    - intermediate quantized data transmission

- Energy vs. Detection Trade-off depends crucially on Processor and Communication System Characteristics

(both with respect to the algorithmic and protocol characteristics as well as with respect to the hardware)



# Focus on Embedded Processor

---

- Device Circuit Level
- Microprocessor Architecture Level
- Compiler Level

(at least)



# Device Circuit Level

---

- Fundamental Limits

- $3 \times 10^{-21}$  J/bit (if entropy is not preserved)

- close to zero (if entropy is preserved)

- at the cost of delays

- MICA 2, MICA 2 DOT

- ~80mw at 40 kbs ( $\sim 2 \times 10^{-6}$  J/bit)

- Newer: ~60mw at 250 kbs ( $\sim 2 \times 10^{-7}$  J/bit)

GAP:  $\sim 10^{14}$  !!

# Device Circuit Level (cont.)

$U_d$ : supply voltage

$U_{th}$ : threshold voltage

$U_\theta$ : thermal voltage

$\alpha$ : switching activity,

C : load capacitance

T : temperature

$T_{ox}$  = oxide thickness

f = clock frequency  $\sim \frac{(U_d - U_{th})^2}{U_d T}$

Dynamic Dissipation  $\sim \alpha C U_d^2 f$

Leakage Dissipation  $\sim \underbrace{U_d e^{-\frac{U_{th}}{U_\theta}} (1 - e^{-\frac{U_d}{U_\theta}})}_{\text{sub threshold}} + \underbrace{\frac{U_d^3}{T_{ox}} e^{-\frac{kT_{ox}}{U_d}}}_{\text{gate leakage}}$

- Miniaturization ==> - Reduced Dynamic Energy Expenditure
- Increased Leakage Energy Dissipation

**IMPORTANT:**  $\alpha$  &  $f$  depend on “upstairs”, hence coupling



# Device Circuit Level (cont.)

---

- Mixed Analog and Digital Techniques
  - ASIC
  - Borrowing from bio-systems (NIPS)
    - energy-efficient
    - speed
- Adaptive Hardware
  - e.g. ADC            (build rate conversion into hardware)

Bottom Line: - Hardware techniques save energy at the very  
bottom level

- There is connection to the upper levels



# Microprocessor Architecture Level

---

- Culprit: general purpose design



becomes opportunity

- data paths
  - memory management
  - bus size
  - arithmetic cells
  - address translation
  - page frames
- Hence:
    - Specialize to fit application requirements
    - Adaptive architecture



# Compiler Level

---

- Are there limits on computational energy?
- Key: migration to software (of certain functions)
  - Some functions can gain energy advantages  
example: software-managed scratch-pad memory (turning off unused memory banks) & replacing cache memories
- Implications: Connection to MAC protocols and Routing
  - scheduled transmissions vs. random access
  - proactive vs. reactive routing

➔ Avoid Event-Driven Processes ←



# Conclusion

---

- Focus on a specific problem highlights the ASOC nature of sensor networking
- Importance of energy (split between transmission and processing)
- Focus on hardware (but not in isolation) leads to “zero-base cross-layering”
- Rich research agenda ahead