

Embedded Systems: Part III – Self-configuration and adaptive coordination

April 17, 2003

Class Meeting 26



Announcements

- Remember correction from last time:
 - Reading for last time should have been Chapters 1 and 2 of Embedded Everywhere!!
 - Reading for today should have been Chapter 3 of Embedded Everywhere!!
- Assignment #6
 - Due 1 week from today (no extensions)
 - Revised (x,y) goal/task placement constraints:
 - $10,000 \geq |x| \geq 3000$
 - $10,000 \geq |y| \geq 3000$
 - Questions?

Announcements (con't.)

- No class on Tuesday, April 29 (I have to be in D.C.)
 - Last class is next Thursday (April 24th).
 - We'll include review for final next Thursday
- Final Exam:
 - Tuesday, May 6th, 10:15 – 11:30 (same length as other exams)
 - Will predominantly cover last 1/3 of class (since Exam #2), but will assume working knowledge of all material covered this semester
 - Questions regarding final (since I'll be out of town Apr. 28 – May 1):
 - Email me your questions, if they can reasonably be answered via email
 - Or, send me an email to arrange an appointment for either Friday, May 2nd or Monday, May 5th, to discuss your questions in person

Today: Self-Configuration and Adaptive Coordination in Embedded Networked Systems (EmNets)

- Today's reading:

Embedded, Everywhere, Chapter 3, National Academy Press, 2001.

- Focus:

- Mechanisms needed to achieve automatic reconfiguration

- Individual nodes will often need to:

- Assemble themselves into a networked system
- Find available resources on the network
- Respond to changes in desired functionality
- Operate with little human intervention or guidance

Underlying Mechanisms

- Components need to:
 - Discover other resources on network and communicate with them
 - Sense changing environmental conditions or changing system capabilities and respond appropriately
- Both software and hardware adaptability are important
 - SW can change
 - HW can take advantage of reconfigurable computing technologies to adapt
- Efficient system designs:
 - Exploit higher-capacity and resource-rich components where they exist
 - Exploit redundancy provided by deploying large numbers of inexpensive components
- **Question:** How is this different from PDAs and cell phones?

Differences with PDAs and Cell Phones

- PDAs and cell phones don't need to take into account :
 - Very large numbers of interacting elements
 - Distributed control
 - Severe energy constraints
 - Kinds of physical coupling that many EmNets must accommodate

Outline for Discussion

1. Key concepts that provide mechanisms for self-configuration and adaptive coordination
2. Elements of self-configuration and adaptive coordination in existing distributed systems
3. Research needed to realize vision for robust, scalable EmNets

1. Terminology

- Self-configuration and adaptive coordination:
 - Spectrum of changes that a system makes to itself in response to occurrences in its environment and internally
 - Does not imply infinite flexibility
 - Changes will be within constraints of system's planned functionality

- Spectrum of functionality:

Adaptive coordination

Self-configuration

Increasing amount of change →
← Increasing speed of change

- Often, involves different time scales

Self-Configuration

- Involves:
 - Addition, removal, or modification of elements contained in EmNet
 - Process of establishing interoperability among components
 - Locating essential services (e.g., data aggregation nodes in sensor networks)
- **Self-configuration** is process of interconnecting available elements into an ensemble that will perform the required functions at the desired performance level
- Changes composition of EmNet
 - May alter the distribution of functionality across components
 - May alter system's overall function based on which components are available

Adaptive Coordination

- Involves:
 - Changes in behavior of system as it responds to changes in the environment or system resources
 - Perhaps, multiple levels of service, depending on locally perceived conditions or detected events
- Does not involve: changes in system's function
- Example:
 - Changing actions based upon density of redundant components
 - Components may alternate responsibilities
- **Adaptive Coordination** refers to changes in operational parameters that are made because of variations in available resources or load
 - E.g., energy
 - Computational resources
 - Bandwidth

Examples: What types of adaption/reconfiguration expected?

- In automotive applications?
- In precision agriculture applications?
- In battlefield applications?

Nature of Configuration/Adaptive Coordination

- Depends heavily on type of application the EmNet supports
- In automobiles:
 - Focus of self-reconfiguration would probably be on accommodating the heterogeneity of system components introduced to or removed from the system
- In precision agriculture or environmental monitoring:
 - System composition varies less because application is more constrained
 - More attention paid to adapting the nodes' operation parameters to unpredictable and varying environmental conditions
- In battlefield:
 - Both adaptive coordination and reconfiguration required, where changes in environment and system makeup can occur rapidly

2. Elements of Self-Configuration and Adaptive Coordination in Existing Distributed Systems

- Discovery in Distributed Systems
- Joining the Network
- Advertising and Finding Services
- Location
- Interfaces and Interoperability
- Address Configuration
- Wire Protocols
- Mobile Code
- Adaptive Coordination in Existing Networks
- Load Balancing
- Ad Hoc Routing
- Adaptive Congestion Control in TCP

3a. Research Challenges for Self-Configuration

- Design basics and criteria:
 - EmNets will appear in hybrid environments of mobile and static networks
 - Users will expect to connect to networks and services as they enter vehicles, buildings, and outdoor environments
 - Nodes will be diverse in capability, energy availability, nature and quality of connectivity, and priority
 - Physical node access will depend on context
 - Variability in priority will dictate when and if a node is revealed or has services revealed to it at the physical layer

Wireless Physical Layer

- Limited by:
 - Low data communications rates
 - Sharp decay of radiated power with increasing range
 - Susceptibility to interference
 - → May have highly variable performance across space and time
- Is diverse:
 - Includes local-area, spread-spectrum networks
 - Includes wide-area cellular, pager, and satellite communication systems
- Methods are needed for joining these different networks and bridging across adjoining cells
- Must include capabilities for low-bit-rate, low-power, low-cost access for virtually all nodes

Example: Ad Hoc Sensor Networks

- Excellent example of issues to be addressed
- Deployment of sensors over wide areas with no predetermined arrangement
- Devices must discover each other and decide how sensor information will flow through the network
- Different devices may have different roles as generators, routers, or aggregators
- Global efficiency achieved only if locally derived information is propagated to other nodes
- Devices must configure functions to produce desired overall effect, rather than optimizing strictly local concerns

Key Research Areas Needed in Self-Reconfiguration

- Dealing with heterogeneity
- Trust models
- Failure models

Dealing with Heterogeneity

- Tradeoff between memory use and use of mobile code
 - Many small, embedded components have very limited memory → mobile code gives up ability to control memory use
- Code loaded onto a node is determined by use of network in a specific situation
- Is an aspect of the design of the network, not the components
- Protocol-based configuration mobile code:
 - Allows tailoring devices to new applications and evolve their functions over time

Trust Models

- “Trust”

- Admission and allocation to resources
- Monitoring and policing
- Ability to deal with failures, some of which may be intentionally afflicted

- Key questions:

- Whether the receiver trusts the mobile code and allows it to run in any fashion
- What local resources code can access if it is allowed to run
- What rights the local client might want to delegate to the code if it moves on or needs to make calls to other members in the network

Failure Models and Monitoring

- Failure models/monitoring:

- Once a device has joined a network, how is it discovered that the device has failed?

- Issues:

- Lack of elements in one area → low-bandwidth bottleneck or surplus → communications interference

- Need graceful degradation

- How to characterize systems and their components based on this concept

3b. Research Challenges for Adaptive Coordination

- Exploiting redundancy
- Centralized vs. Decentralized control
- Collaborative processing

Exploiting Redundancy

- Large # of elements suggest using brute force to achieve adaptive coordination → just add more elements
- But, requires communication bandwidths that would drain available energy of battery-powered elements
- Redundancy:
 - Elements that overlap
 - E.g., in sensor networks, multiple nodes may provide coverage of overlapping geographic areas
 - Smart Space → Multiple printers, displays, databases might exist

Possible Techniques for Exploiting Redundancy

- Clustering: a single nodes collects information from other nodes and takes on task of communicating that information to other clusters
- Time division multiple access (TMMA): enables nodes to take turns using communication slots
- Design EmNets to take advantage of opportunistic behaviors:
 - E.g., Delay basic reporting until greater bandwidth, energy, or processing capabilities become available

Centralized vs. Decentralized Control

- If individual elements of EmNet can change their technical characteristics, capabilities, and operating modes, how can system guarantee its overall performance and stability?
- If system contains large numbers of nodes, how can a central node control overall configuration of the network in a timely fashion?

Centralized

System self-configures;
components do not

Decentralized

Components self-configure

Hierarchical, Tiered Structure Helpful

- Provides a degree of centralized control in a large EmNet with numerous elements
- Hierarchy itself can reconfigure/adapt
- Examples of how it would work?
- Centralized control rules: can be optimized, but decision-making process is very complex
- Decentralized control rules: Theory for system characterization is not well developed

Collaborative Processing

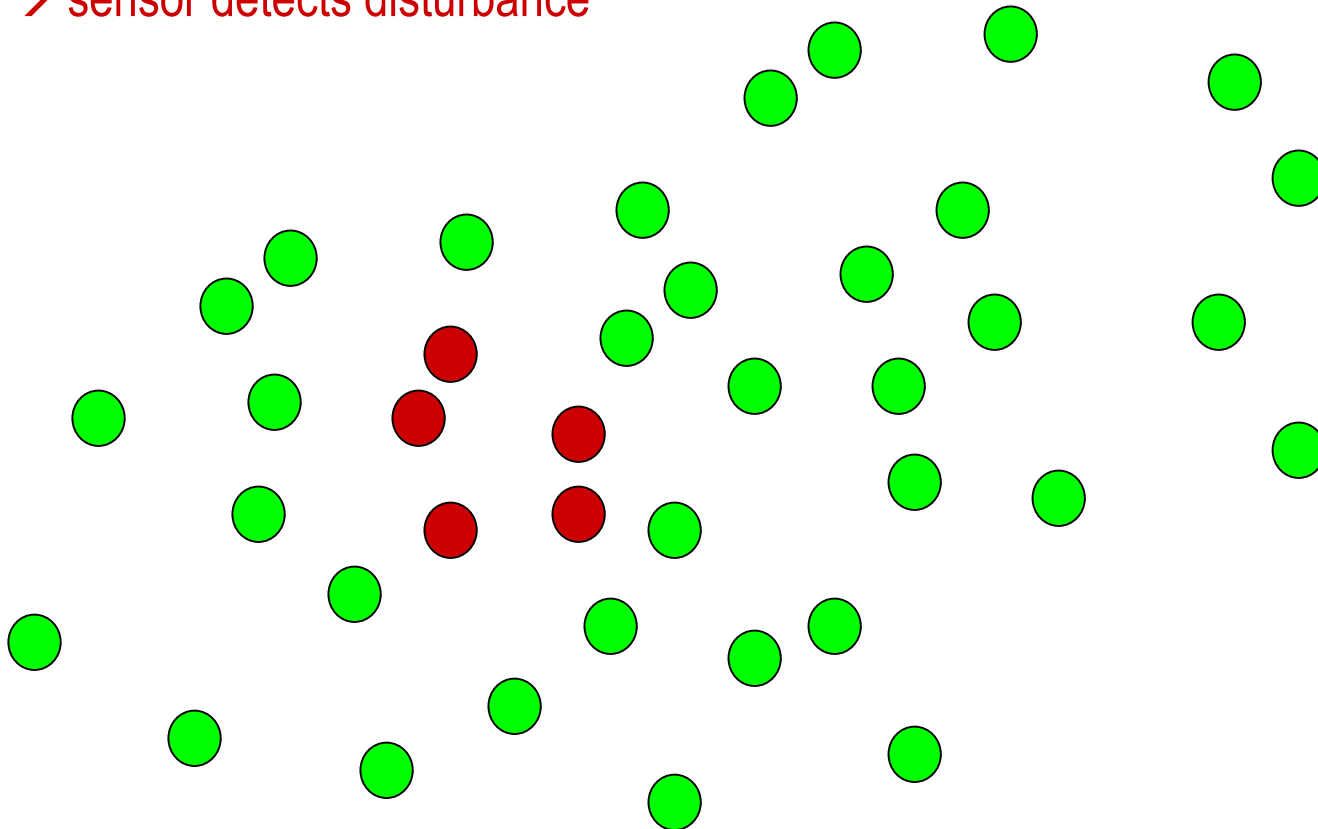
- Tradeoffs: extent of processing conducted by individual sensors and amount of information communicated among them
- Processing hierarchy: can lead to huge reductions in energy consumption while assuring the required level of reliability
- Issues:
 - How long and where are data stored – different at different levels of hierarchy
 - Amount of energy consumed by communications and availability of energy affect processing strategy
- Collaborative processing: can extend effective range of sensors and enable new functions

Collaborative Processing Example: Target Tracking

- How/what do they report, and how is this information used to achieve target localization?

● = sensor

● → sensor detects disturbance



Summary

- Self-configuration:
 - Involves addition, removal, or modification of elements in EmNET and subsequent process of establishing interoperability
- Adaptive coordination:
 - Addresses changes in behavior of a system as it responds to changes in the environment or system resources
- Significant research progress needed:
 - To achieve automatic self-configuration among large numbers of distributed nodes
 - To conform to well-defined trust and failure models
 - To achieve distributed adaptive coordination
 - To exploit system redundancies and localized processing and collaborative signal-processing techniques

Next Time

- Smart Rooms