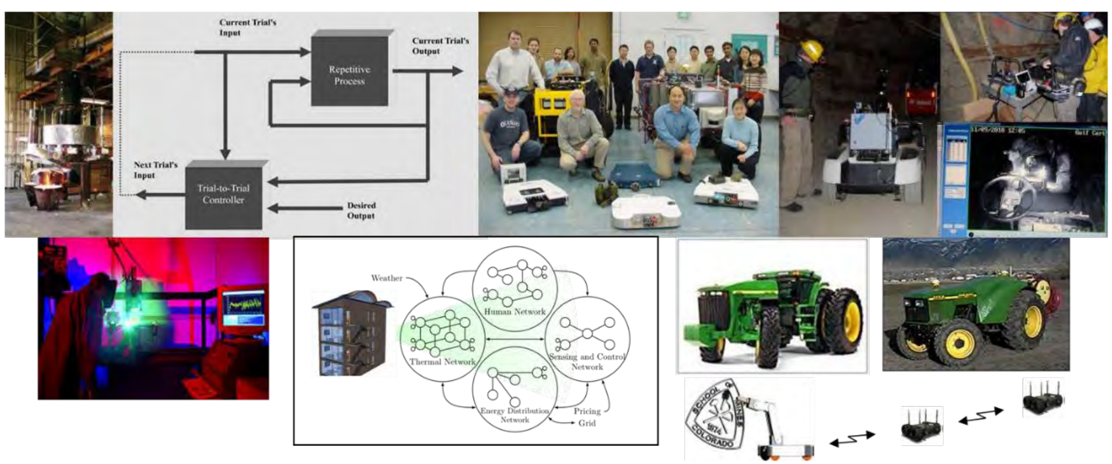
Perspectives and Connections: From Control Theory to Intelligent Robots to Cooperative Autonomy

Kevin L. Moore
Executive Director, Humanitarian Engineering
Director, Mines Robotics Program
Professor of Engineering, Design, and Society
Professor of Electrical Engineering
Colorado School of Mines

Seminar
University of Minnesota Duluth
Department of Electrical Engineering
28 October 2020



Colorado School of Mines

Located in Golden, Colorado, USA
10 miles West of Denver

Mines sits in the foothills of the Rocky Mountains

Mines has ~320 faculty and ~6500 students (~5000 undergrad and ~1500 grad)

Mines is a public research institution devoted to engineering and applied science, especially:

- Discovery and recovery of **resources**
- Conversion of resources to **materials and energy**
- Utilization in advanced **processes and products**
- Economic and social systems necessary to ensure **prudent and provident use of resources in a sustainable global society**

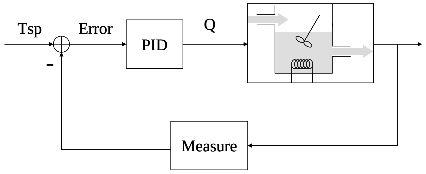
Sustainable Global Society: We mean it!

- Foundational research and education programs in extractive industries, conventional and unconventional energy sources
  - Mining, Geology, Petroleum Engineering, Geophysics, Material Science
- Recent research excellence in renewables and power
  - Solar materials research, Wind, Bio fuels
  - Alternate, reduced carbon fuel technologies (fuel cells, batteries)
  - Power transmission and distribution, and grid integration research
- Top-flight Geotechnical and Subsurface Sciences; Water Resources (NSF ERC);
- Novel laboratories (e.g., wind tunnel + physical subsurface simulator)
- Coursework in sustainable design; energy minor
- Programs in Biomedical, Biomechanics, Bio-environmental; Quant-bio
- Unique program in Space Resources
- Humanitarian Engineering – undergraduate minor and Masters degree
- Graduate Robotics Program
- Undergraduate robotics emphasis in interdisciplinary BSE program

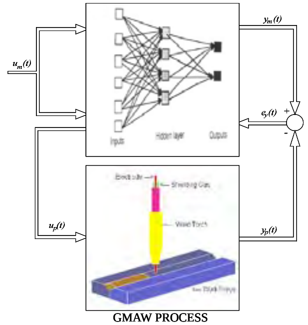
(6 signals/systems/control peoples and another 5 robotics people)

Today: Perspectives and connections

From Control …

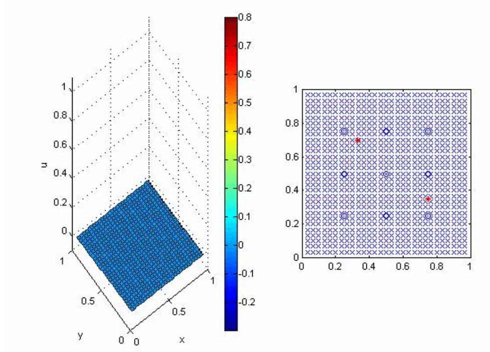


… to Intelligent Control …



… to Intelligent Behavior …

… to Cooperative Autonomy …



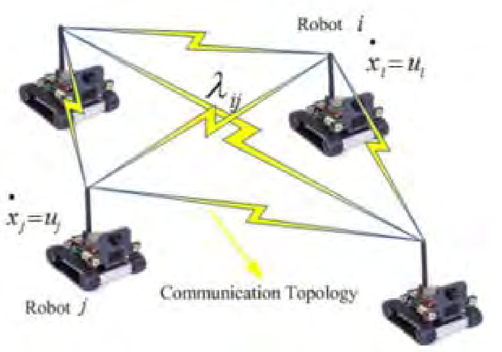
Purpose and Outline

Purpose

- Describe threads of research from my experience
- Give perspectives on research, emphasizing
  - Importance of connections (topics and people)
  - Importance of intentionality
- Illustrative examples, some technical results, but no proofs (though, controls people like proofs!)

Outline

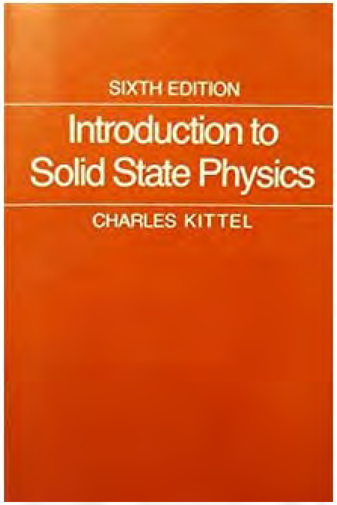
- Part 1: Developing a control-theoretic perspective
- Part 2: Concepts for single-entity autonomy
- Part 3: Concepts for multi-entity autonomy
- Concluding Remarks



Part 1: A control-theoretic perspective

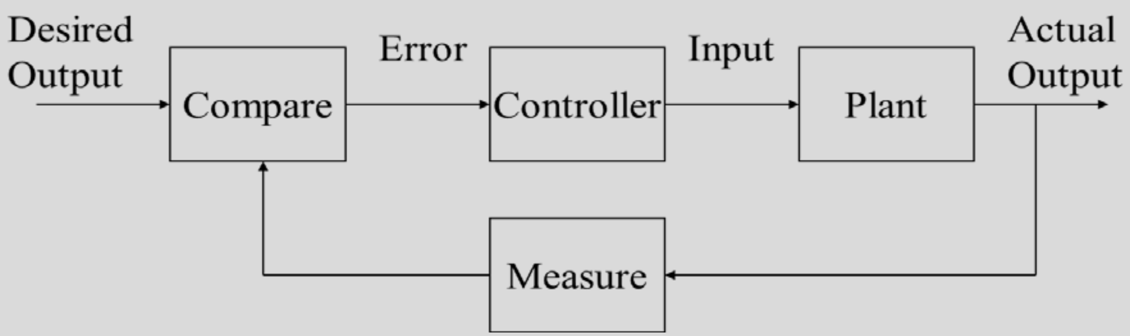
- Liked control systems a lot as an undergrad, but …
- I started my MS degree determined to “save the world” by focusing on materials for improved photovoltaics
  - Got through Kittel and infinite potential well Schrodinger's equations, but …
  - Hated it! (and got C’s!)
- Took an special topics in Nonlinear and Adaptive Control from Petros Ioannou
  - Had a special lecture by Petar Kokotovic, and …
  - Loved it! (and got A’s, except in stochastics!)
- Similar work experiences at Hughes Aircraft
- Switched my major and changed departments at work

Comment 1: You’ll do better work if you truly like what you are doing

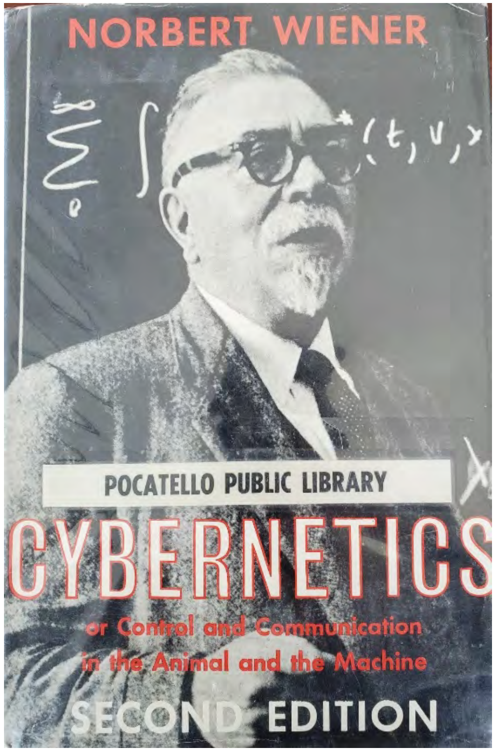


Research: Follow your interests

- I quickly (over the next 10 years) developed a perspective and mindset that came to guide my academic career: that of the role of feedback
- Was particularly influenced by a 1948 book by Norbert Wiener
- Perspective from my introductory lecture to both undergraduates and graduate students:

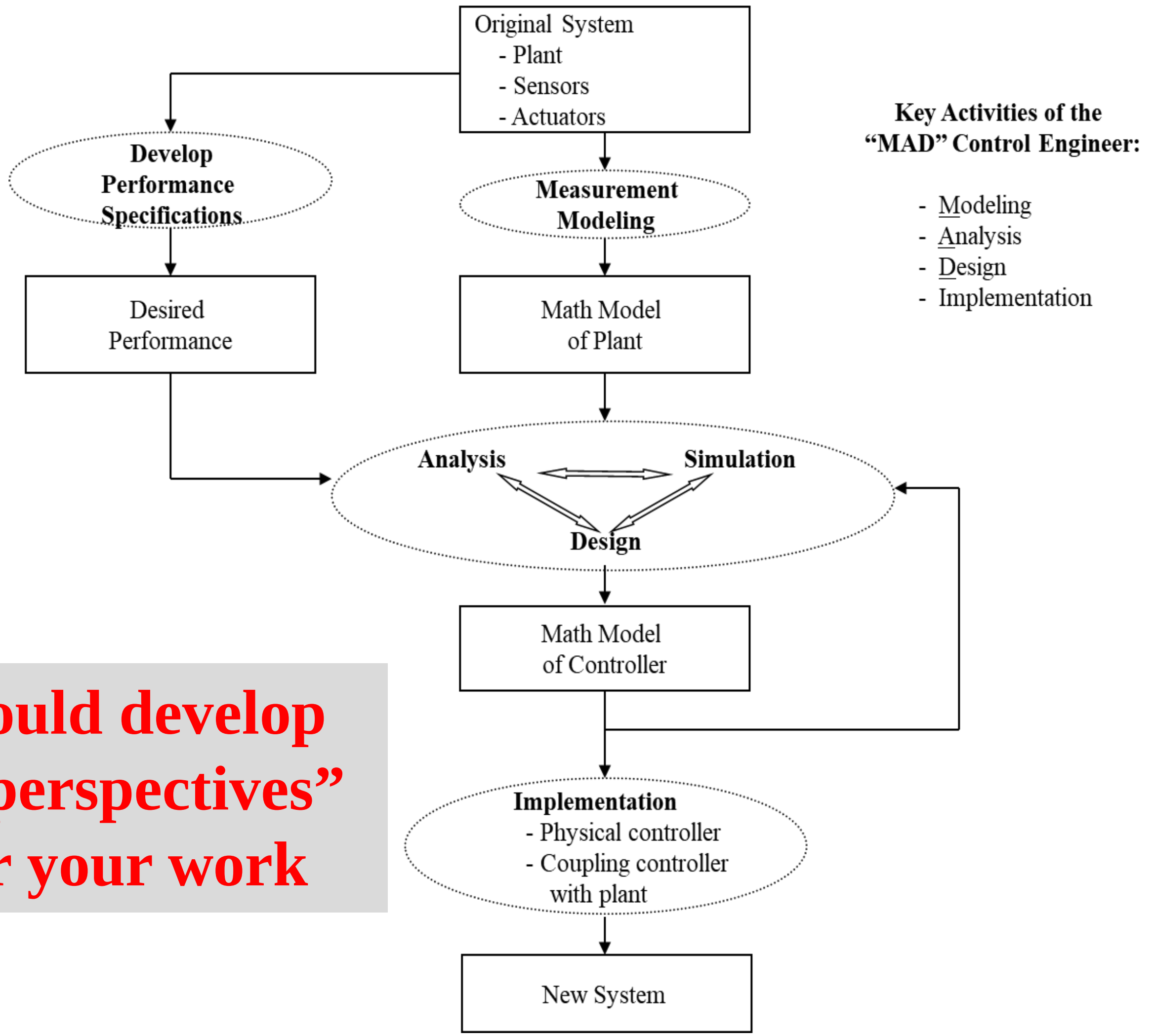


Feedback is the essential feature of an effective control system (it **helps us deal with uncertainty** in our knowledge of the plant and changes in the plant behavior): applies to **engineered and natural systems**



The “MAD” control engineer

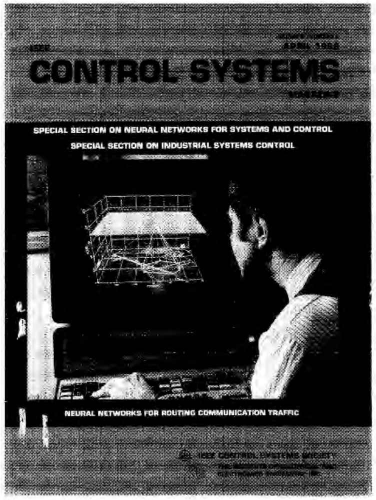
- I also developed and teach a particular framework: the “MAD” control engineer
- This framework has been a useful guide to nearly all my R&D experiences as well as my pedagogical approach to teaching control systems



Comment 2: You should develop your own personal “perspectives” and frameworks for your work

From control to intelligent control

- I wanted to study "Large Scale Systems"
- My advisor wasn't interested
  - Could change advisors
  - Other possible person wasn't interested
- Instead, he asked me to consider an open-problem: minimizing overshoot in a closed-loop system
- Developed two concepts:
  - Optimization-based approach
  - Multi-rate control approach
- Them, two seemingly separate things came together
  - My advisor said "look into these things called neural networks" (April 1988 Special Section in CSM)
  - A visitor (Shinji Hara) came to visit my advisor and gave a talk on learning in robotic systems
- This led to my research focus: connecting control with learning in neural networks and other "intelligent" ideas

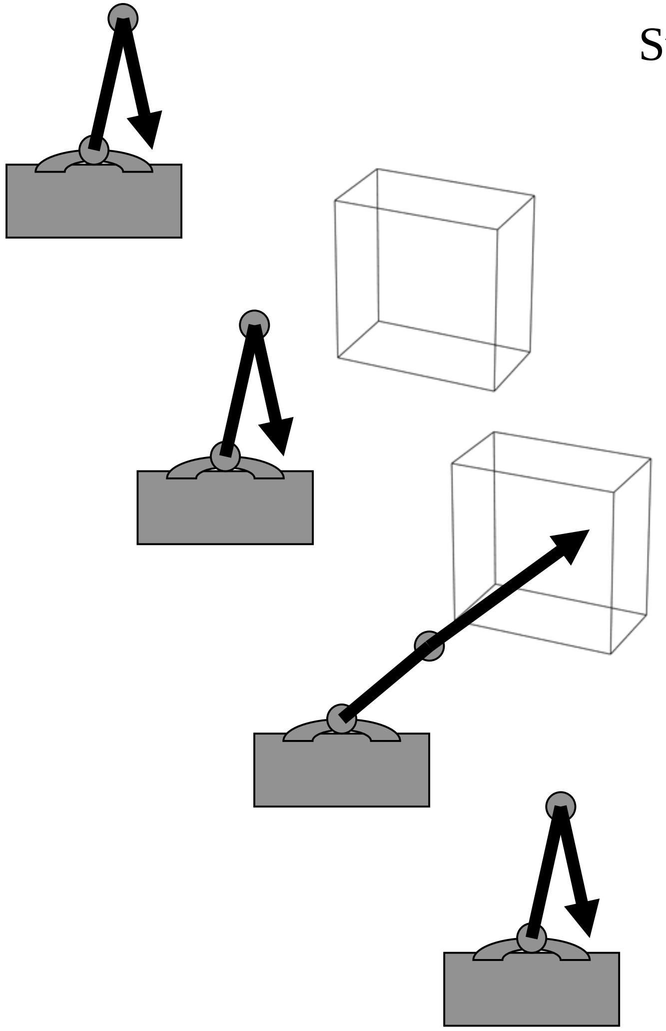


Comment 3: Be patient and look for the connections

On intelligent control

- "**Intelligent control**" is a well-used phrase
- In IEEE CSS society, intelligent control has come to be called "**computational intelligence**" and is often associated with
  - Fuzzy logic
  - Neural nets
  - Evolutionary algorithms (GAs)
  - Adaptive and learning control
- It was in the context of adaptive and learning control that I began working on my main research topic:
  - **Iterative learning control**, or ILC

Iterative Learning Control – Systems that repeat the same action over-and-over



Step 1: Robot at rest, waiting for workpiece

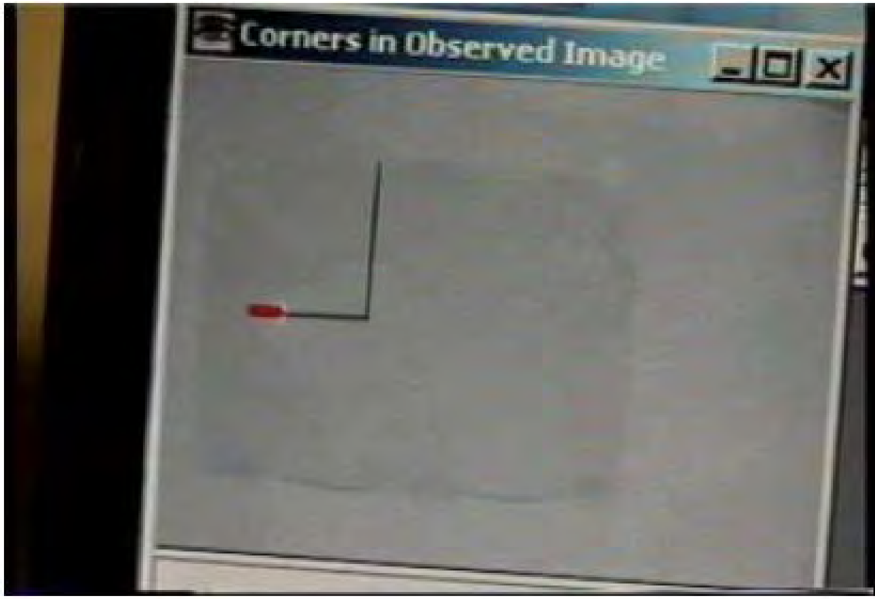
Step 2: Workpiece moved into position

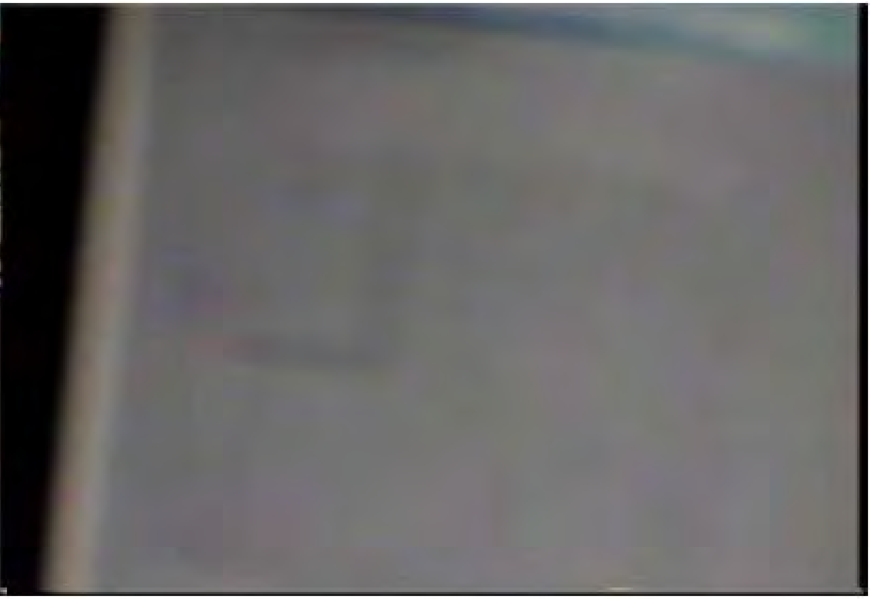
Step 3: Robot moves to desired location and executes its task

Step 4: Robot returns to rest and waits for next workpiece

ILC Goal: Improve from trial-to-trial

Iterative Learning Control example





Gimbal Motion (k = 1)

Gimbal Motion (k = 4)

Correction is made in pixels of error along the path, in the image frame, not along a time-based trajectory

Iterative Learning Control (ILC)

- ILC can be described using the previous example:
  - The error in the result of an action ….

    is used to correct the action on the next try
- Now, this sounds like nothing more than feedback control
- But ILC adds a few more wrinkles:

  The error in the **sequence of results** of a sequence of actions

  …. is used to correct the **sequence of actions** on the **next try**

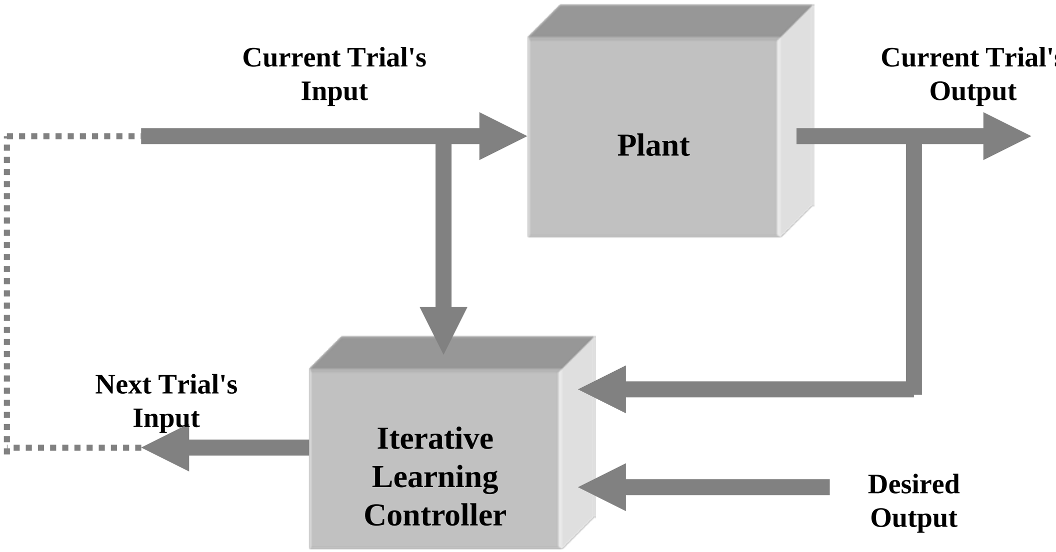
Assume the error is relative to a **reference that is the same from trial to trial**

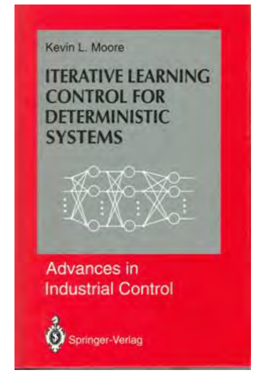
We are concerned with behavior of a system over a **complete trajectory**

Repetition in operation

We will be concerned with **dynamic systems**

Iterative Learning Control algorithm





1993

2007

$$u_{k+1}(t) = u_k(t) + f\left(y_d(t+1) - y_k(t+1)\right)$$

Adaptation in trial

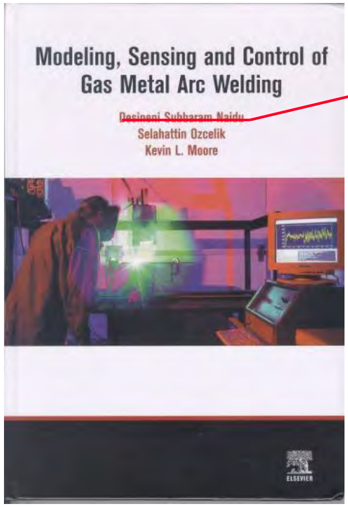
"Non-causal" update in time or space

Intelligent control for material processing

COLORADO SCHOOL OF MINES

- For about the next 10+ years I **applied the ideas of ILC, Neural Networks, and Fuzzy Logic** to a variety of problems, mostly all related to **materials** – an area I had originally thought I wanted to study for my MS!
- Did this work at Idaho State's Measurement and Control Engineering Research Center, which Dr. Naidu and I started

Foundry Control



Welding Control (2003)

D.S. Naidu!

Comment 4: Things sometimes lead you back to where you started

From intelligent control to intelligent behavior in robots and autonomous systems

- I was not a roboticist *per se*
- But, robotics was a key application area for ILC and control
- An opportunity came up to take over a robotics/autonomy research center: Center for Self-Organizing and Intelligent Systems (CSOIS) at Utah State:

Comment 5: You should always be willing to pivot when opportunity arises – it will always lead you somewhere interesting

Part 2: Single-entity autonomy (robots!)

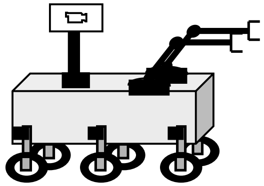
- Give a **tutorial-level introduction** to the technical aspects of designing unmanned autonomous systems.
- We emphasize
  - a **system engineering perspective** on the conceptual design and integration of both
    - the **components** used in unmanned systems including the locomotion, sensors, and computing systems needed to *provide inherent autonomy capability*, and
    - the **algorithms and architectures** needed *to enable control and autonomy*, including path-tracking control and high-level planning strategies.
- Concepts are illustrated using **case study examples** from robotic and unmanned system developed by me and my colleagues

Unmanned Systems

- Let us define:
  - **Unmanned system**: any (usually electro-mechanical) system with the capability to carry out a prescribed task or portion of a prescribed task automatically, without human intervention
  - **Unmanned vehicle**: a vehicle that does not contain a person
    - Can be tele-operated
    - Can be autonomous
    - Typically deploys a payload (sensor or actuator)
- Focus today will be on **unmanned vehicles**
- Unmanned vehicles/systems can come in several flavors: **UxV**
  - Land: UGV
  - Air: UAV
  - Maritime: UUV, USV
  - Sensors: UGS

What Makes a UxV?

- All UxVs have **common elements**:
  - Mechanical components (drive, power, chassis)
  - Electronics
  - Sensing/mission payloads
  - Communication systems
  - Control
  - “Smarts”
  - Interface to user



- Our perspective is that all unmanned systems should be developed from the perspective of its **concept of operations** (CONOPS)
- Once a CONOPS has been defined, then **systems engineering** is used to flow-down requirements for subsystems

Example CONOPS - Automated Tractors

Example CONOPS -Unique Mobility Robots

(Autonomous Solutions, Inc.)

UxVs: Capabilities and Control

- We consider two key aspects of unmanned vehicles and autonomy:
  - Inherent physical capabilities built into the system
  - Intelligent control to exploit these capabilities
- **Inherent physical capabilities**
  - Mechanisms for mobility and manipulation
  - Power
  - Sensors for perception
    - Proprioceptive
    - External
  - Computational power

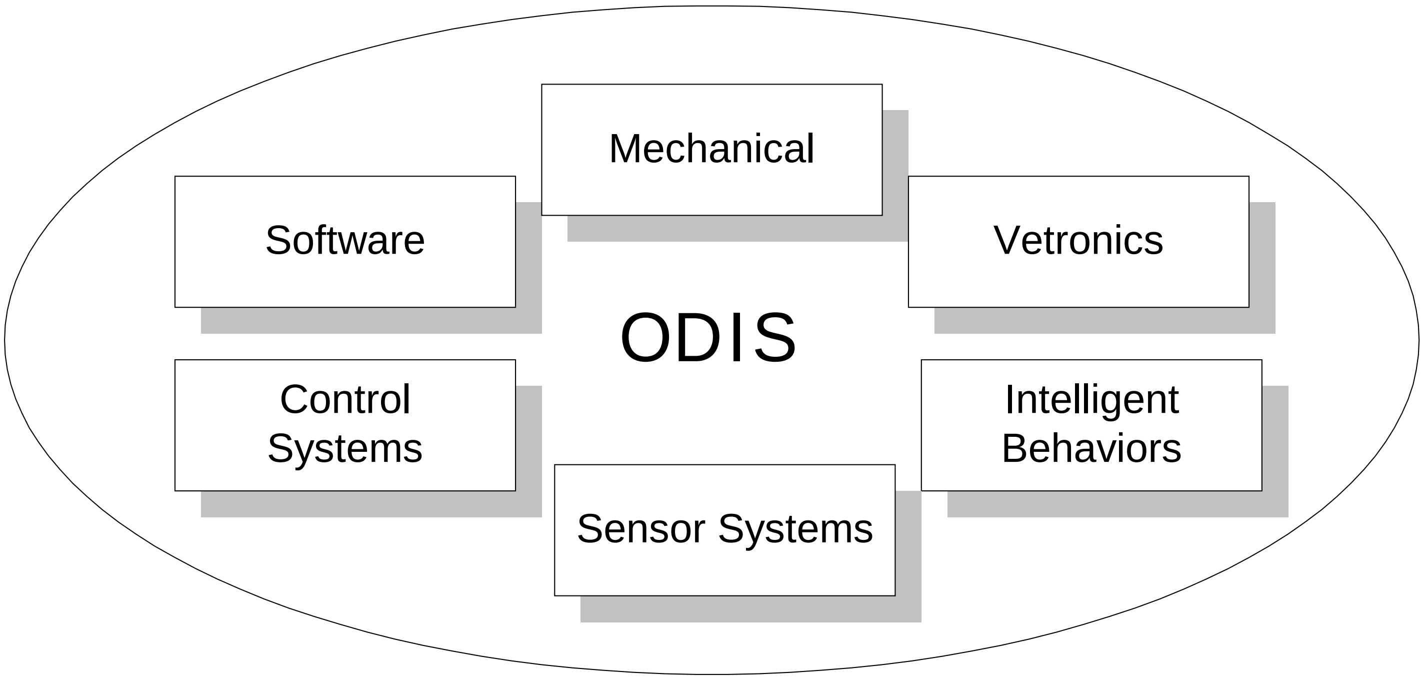
These are driven by your CONOPS

- **Intelligent control to exploit these capabilities**
  - Machine-level control
  - Perception algorithms
  - Reasoning, decision-making, learning
  - Human-machine interfaces

These are driven by your CONOPS, but also by your inherent physical capabilities

Systems Engineering a UGV: Case Study
ODIS Robot Design and Implementation

- Inherent capabilities in ODIS
- Exploitation of these inherent capabilities



Utah State University

CSOIS
CENTER FOR SELF-ORGANIZING AND INTELLIGENT SYSTEMS

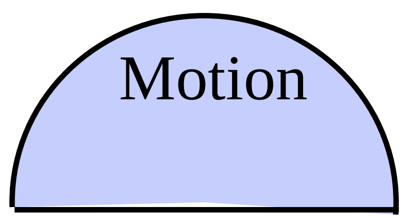
“Putting Robots in Harm’s Way So People Aren’t”

ODIS – the Omni-Directional Inspection System

An ODV Application: Physical Security

ODIS I – An Autonomous Robot CONOPS

UGV Technology: Inherent Capabilities



Motion and Locomotion for Unmanned Systems

Except for UGS, most unmanned systems must move:

- UGV: wheels and tracks
- UAV: fixed wing, rotary wing, VTOL
- USV/UUV: propeller based, jetted

In general the motion and locomotion aspects of an unmanned vehicle are not remarkably different than that of their manned counterparts:

- Design of motion and locomotion system becomes “only” an engineering task!

Some ODV Robots Built At USU

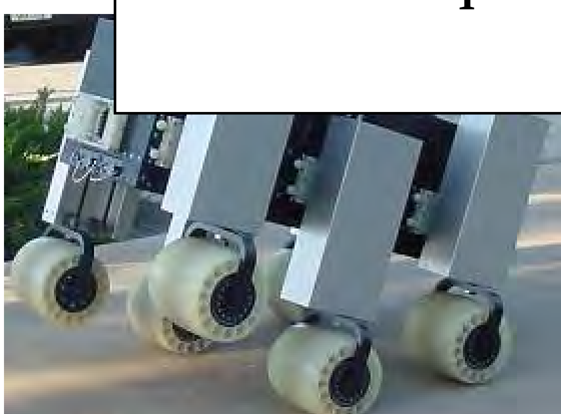
T1 -1998

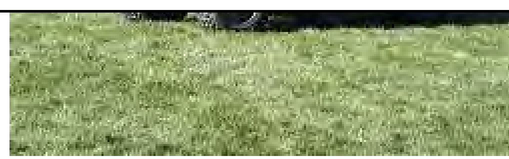
T2 -1998

ODIS I -2000

Typical UGV Mobility Platforms

Ackerman
Skid-Steer
Unicycle
Unique Mobility



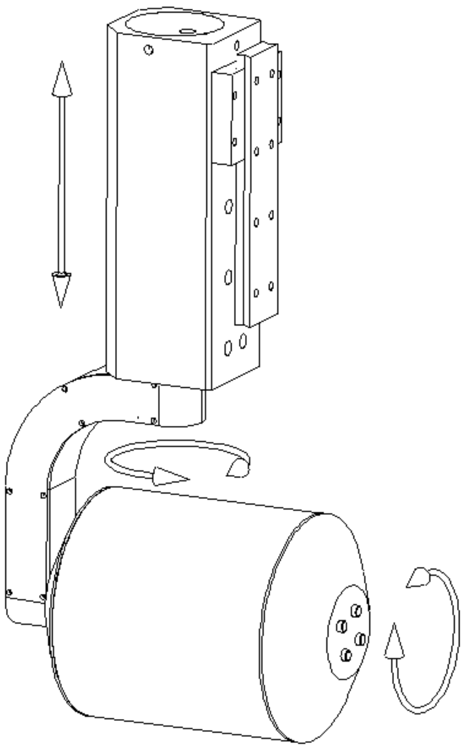


T3 -1999

(Hydraulic drive/steer)

Mobility Example: USU ODV Technology

- USU developed a mobility capability called the “smart wheel”
- Each “smart wheel” has two or three independent degrees of freedom:
  - Drive
  - Steering (infinite rotation)
  - Height
- Multiple smart wheels on a chassis creates a “nearly-holonomic” or omni-directional (ODV) vehicle



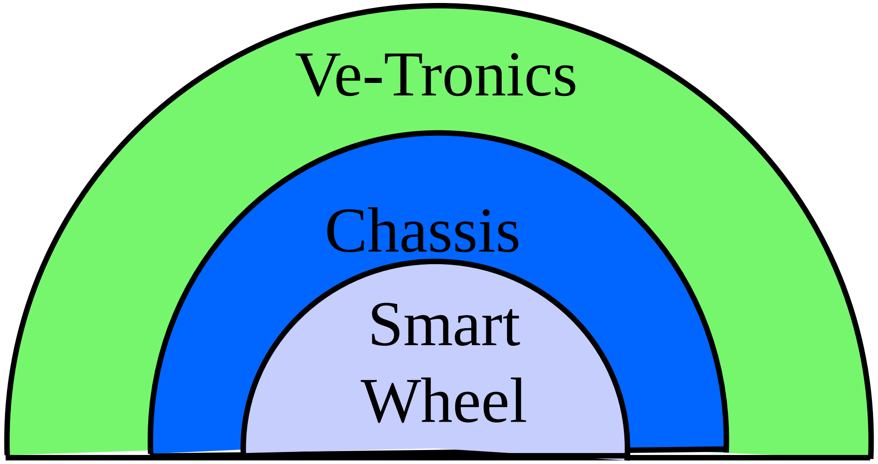
T1 Omni Directional Vehicle (ODV)

ODV steering gives improved mobility compared to conventional steering

Smart wheels make it possible to simultaneously
- Translate
- Rotate

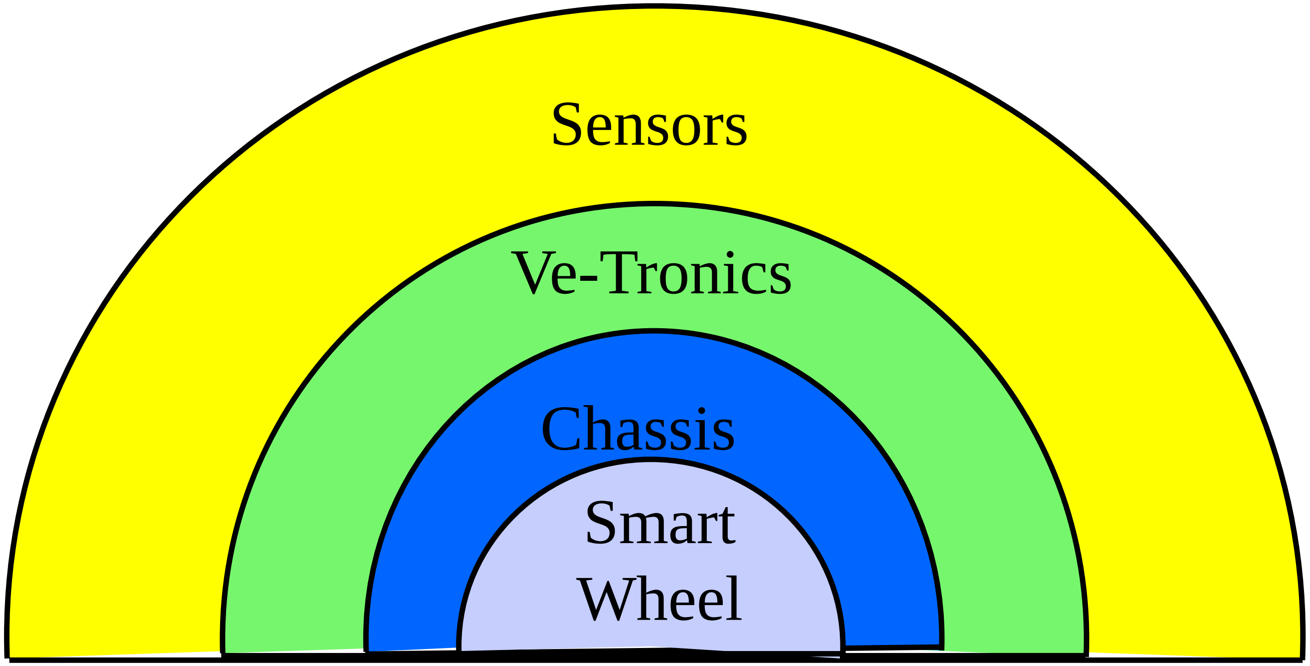
Comment to Grad Students: Notice the role of the graduate student in this video!

UGV Technology: Inherent Capabilities



ODIS Vetronics System

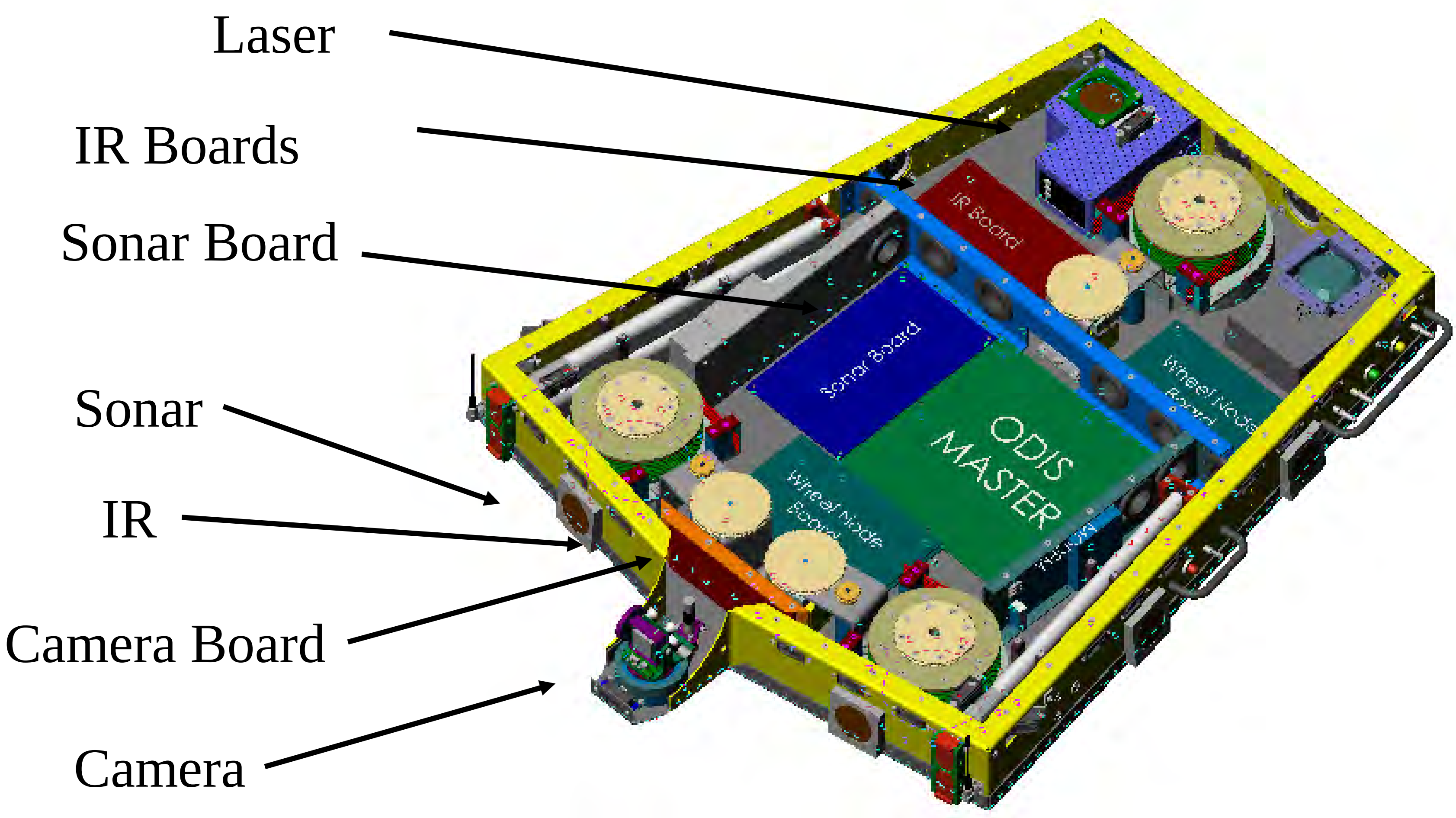
UGV Technology: Inherent Capabilities



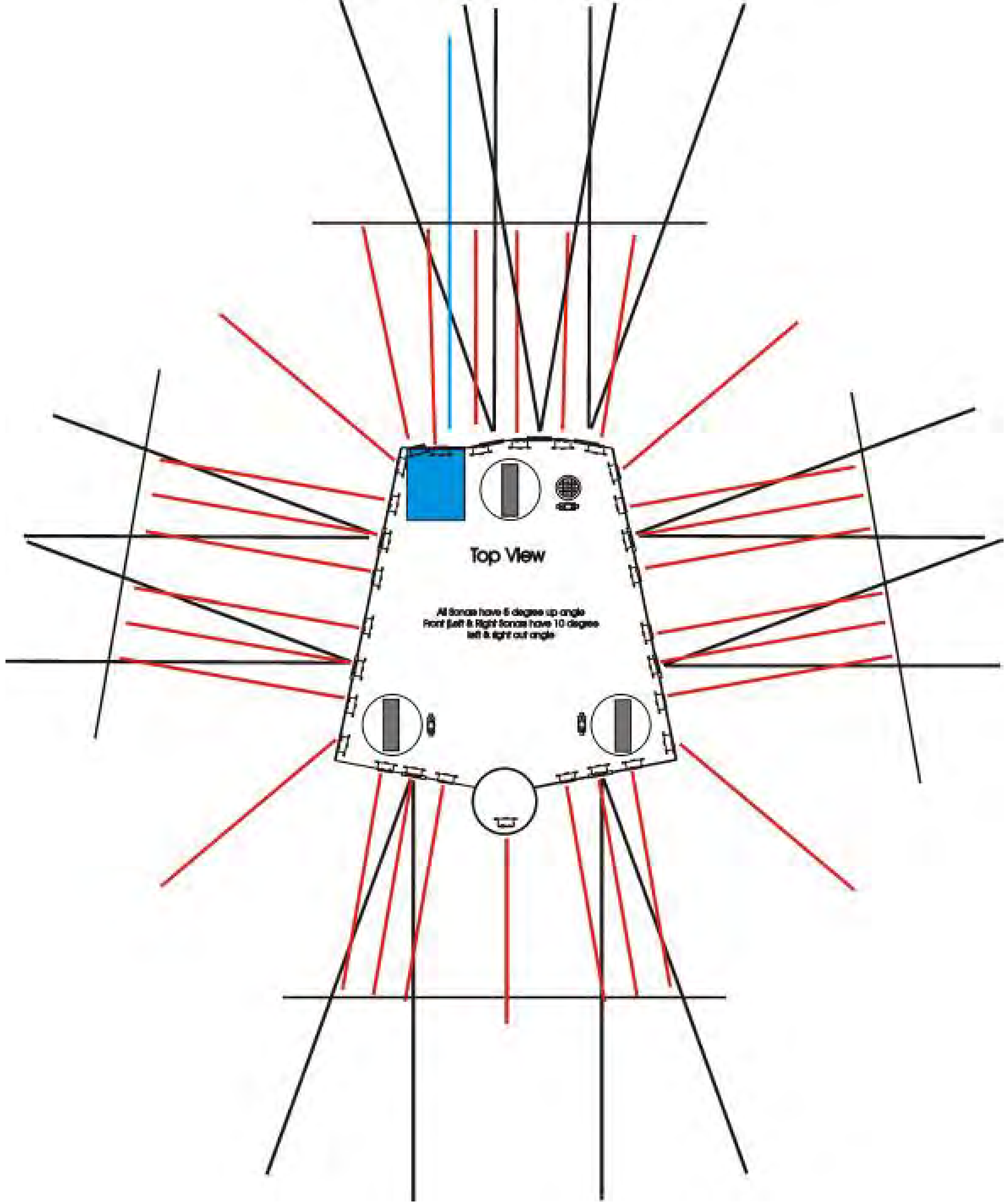
Utah State University

CSOIS
CENTER FOR SELF-ORGANIZING AND INTELLIGENT SYSTEMS

ODIS Sensor Suite



B E A M P A T T E R N

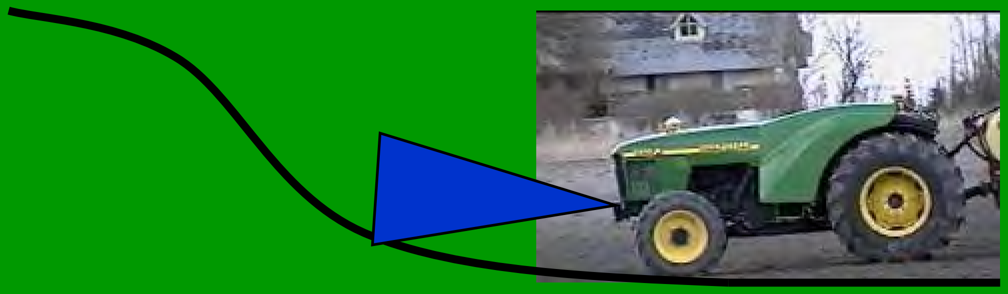


IR

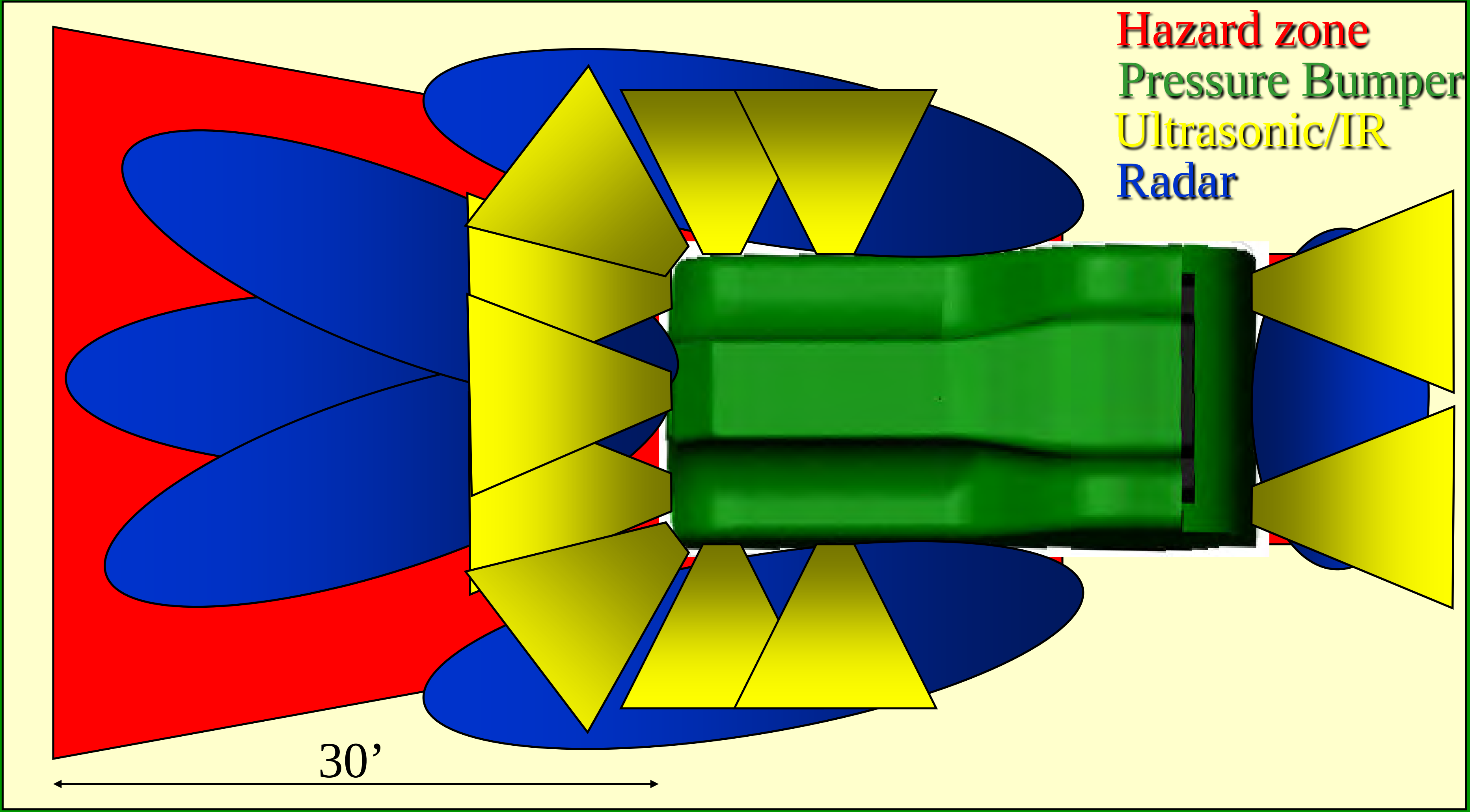
Sonar

Laser

Awareness Issues



3 Tiered Proximity Detection



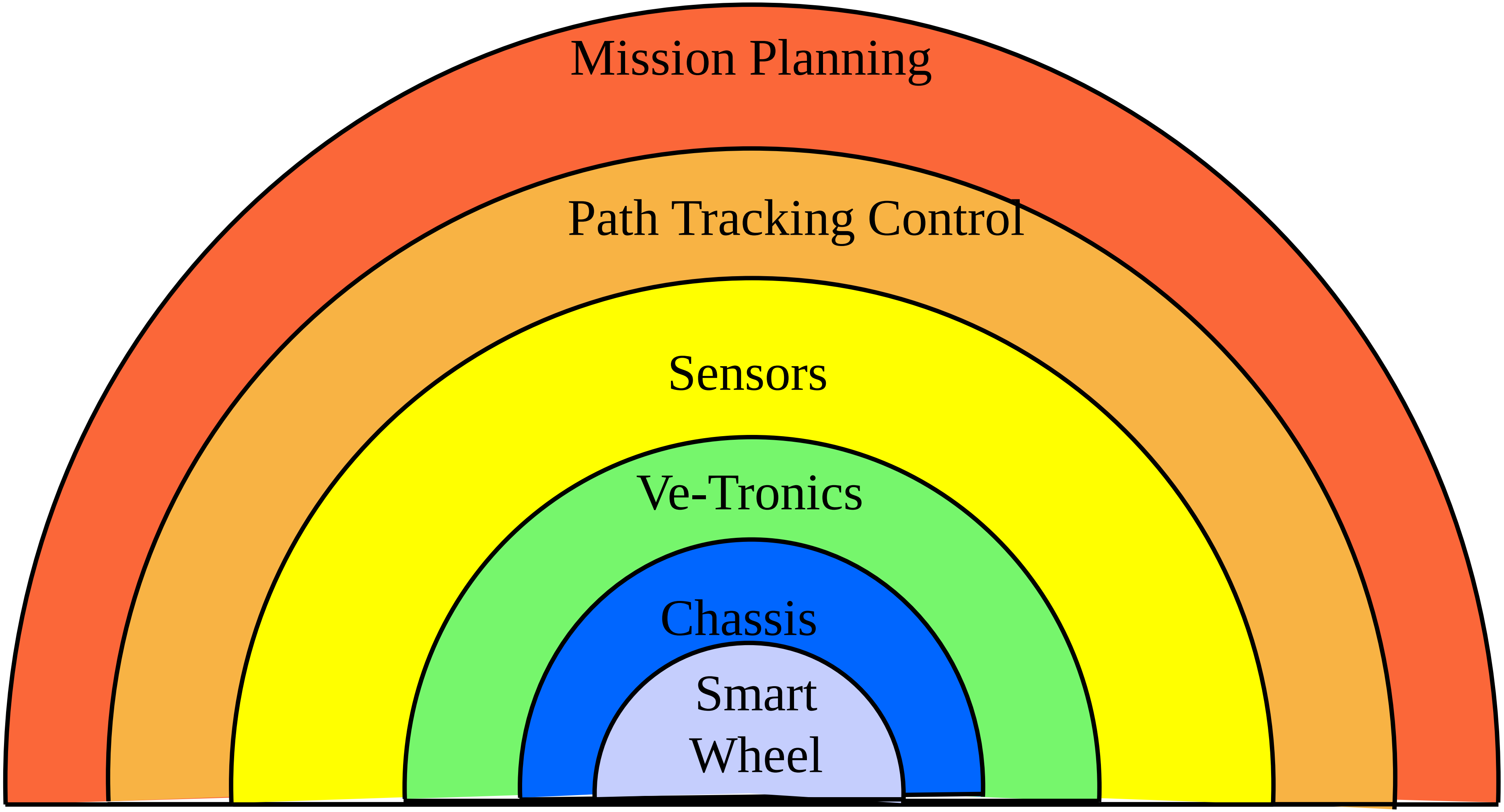
Mission Payloads for UxVs

- Different CONOPS will produce different mission payload requirements.
- ODIS-T Sensor Suites:
  - Visual – pan/tilt imaging camera
  - Passive & active thermal imaging
  - Chemical sniffers – i.e. nitrates, toxic industrial chemicals
  - Night vision sensors
  - Acoustic sensors
  - Radiation detectors – i.e. dirty bombs
  - Biological agents detection
  - MEMS technology – multiple threats
  - License plate recognition

IR Image – Warm Brake

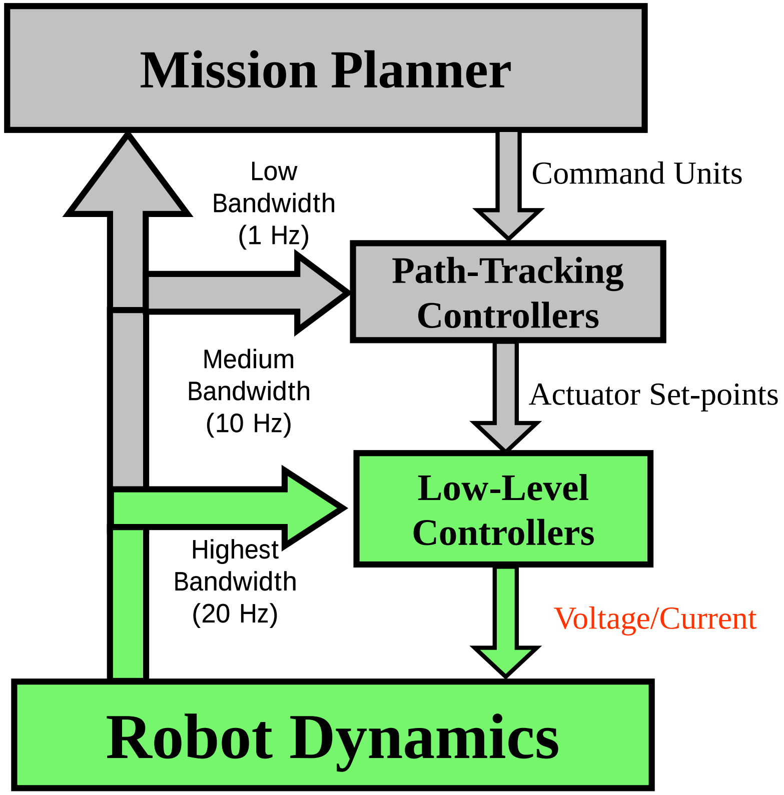
IR Image – Recently Driven Vehicle

UGV Technology: Intelligent Control



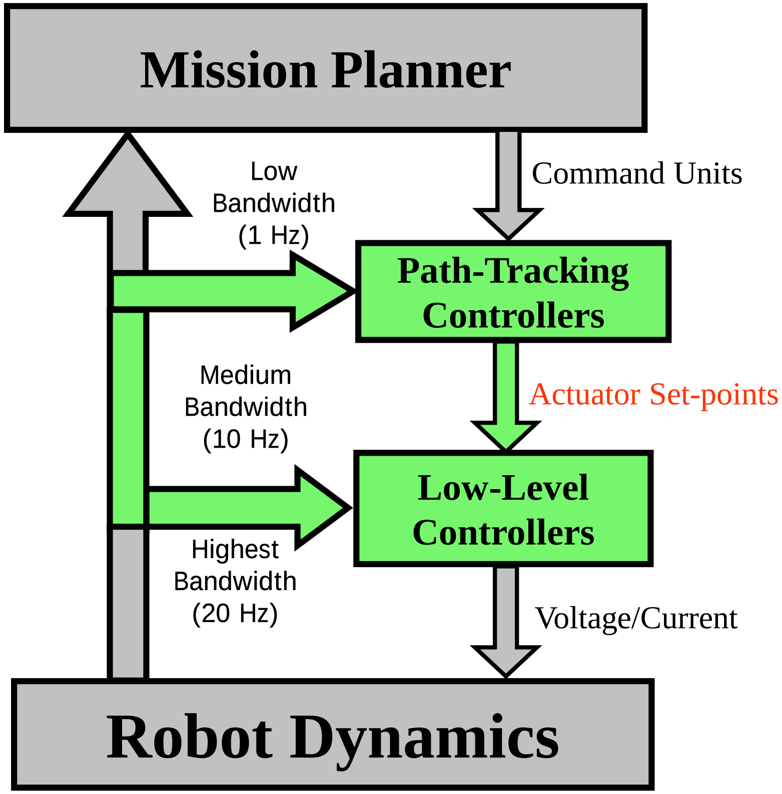
Multi-Resolution Control Strategy

- At the lowest level:
  - Actuators run the robot



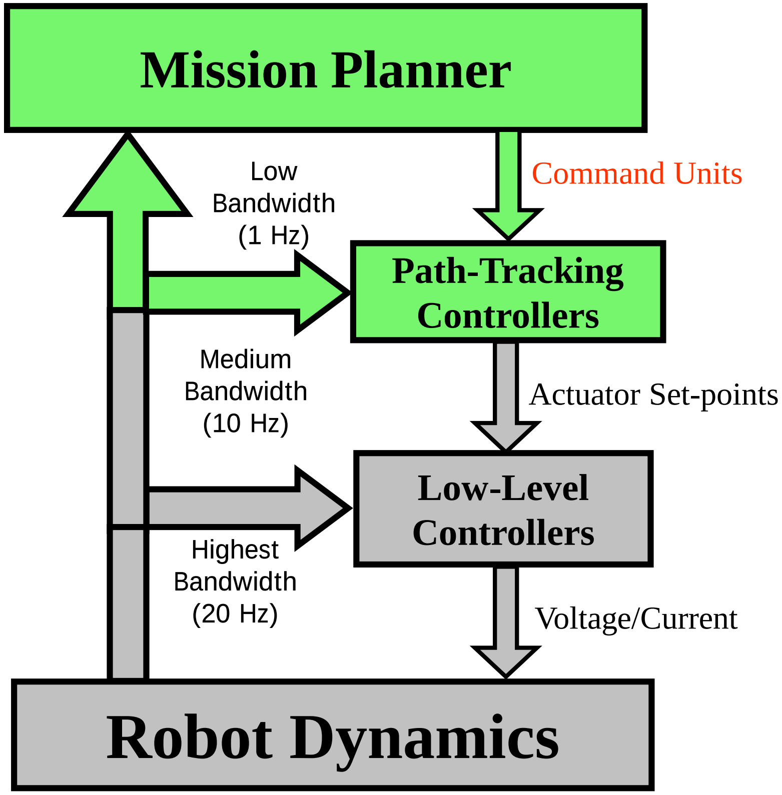
Multi-Resolution Control Strategy

- At the middle level:
  - The path tracking controllers generate set-points (steering angles and drive velocities) and pass them to the low level (actuator) controllers



Multi-Resolution Control Strategy

- At the highest level:
  - The mission planner decomposes a mission into atomic tasks and passes them to the path tracking controllers as command-units



ODIS Command Environment: MoRSE (ROS didn't exist then!)

- Based on command unit:
  - Set of individual commands defining various vehicle actions that will be executed in parallel
- *Commands for XY movement*:
  - moveAlongLine(Line path, Float vmax, Float vtrans = 0)
  - moveAlongArc(Arc path, Float vmax, Float vtrans = 0)
- *Commands for Yaw movement:*
  - yawToAngle(Float angle_I, Float rate = max)
  - yawThroughAngle(Float delta, Float rate = max)
- *Commands for sensing:*
  - SenseSonar
  - SenseLaser
  - SenseIR
  - Camera commands
- A set of rules defines how these commands may be combined

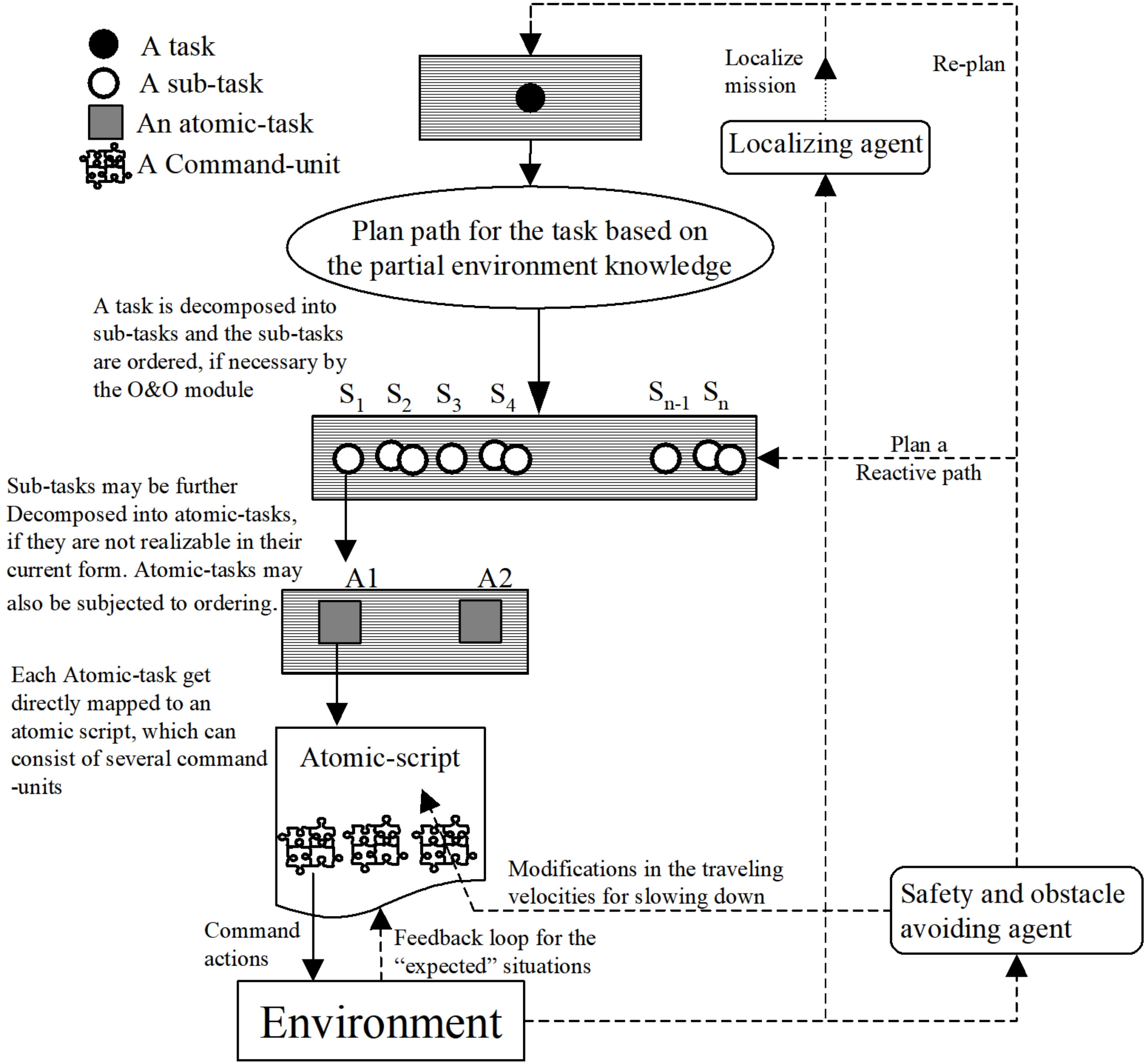
Software Architecture

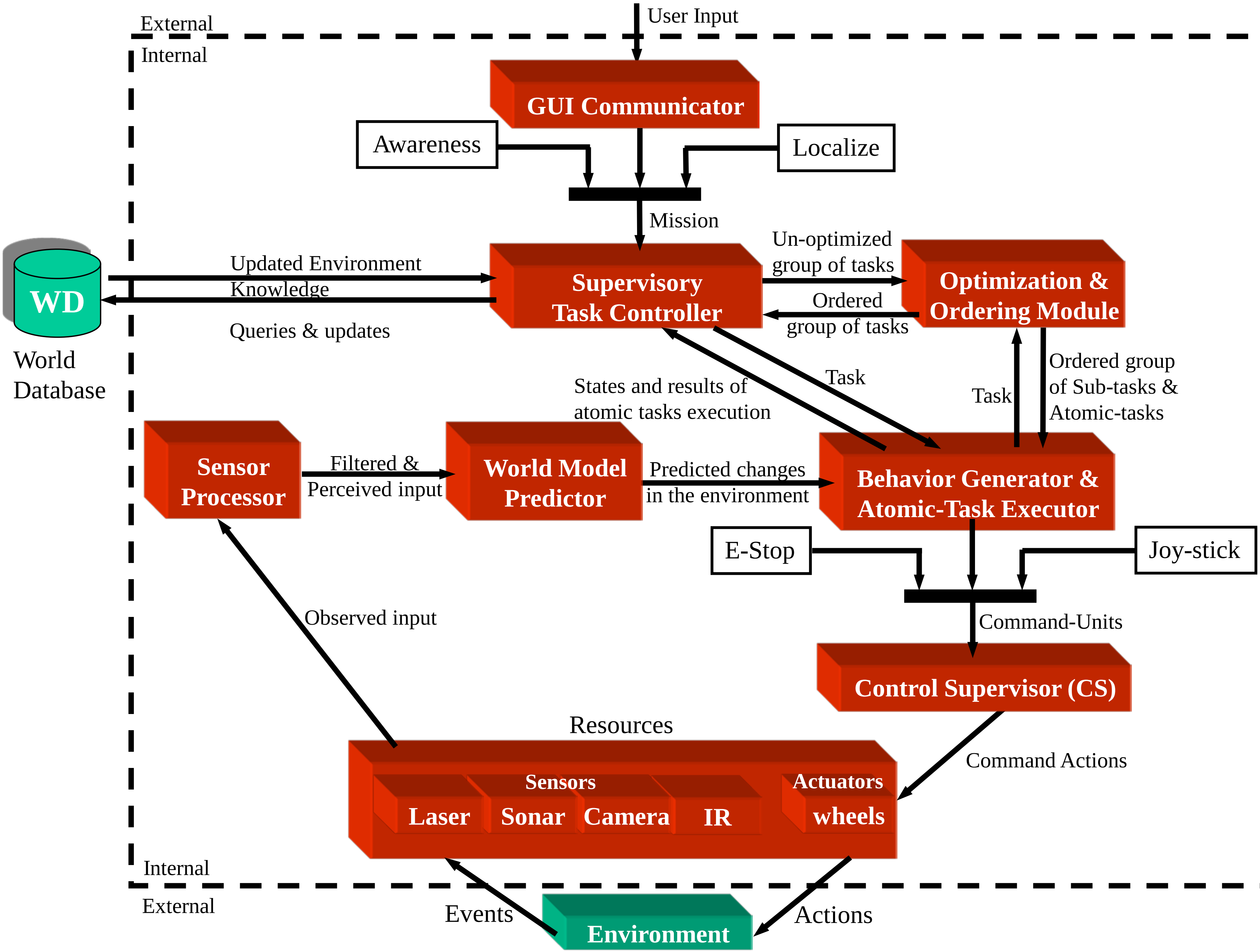
- Command actions are the lowest-level tasks allowed in our architecture that can be commanded to run in parallel
- For planning and intelligent behavior generation, higher-level tasks are defined as compositions of lower-level tasks
- In our hierarchy we define:

Mission	User-defined
Tasks Subtasks	Variable (planned)
Atomic Tasks (Scripts) Command Units Command Actions	Hard-wired (but, (parameterized and sensor-driven)

User-tasks in the environment

- {MoveTo Point}
- {Characterize a stall}
- {Inspect a stall}
- {Characterize a row of stalls}
- {Inspect a row of stalls}
- {Localize}
- {Find my Car}
- {Sweep the parking lot}
- {Sweep Specific area of the parking lot}

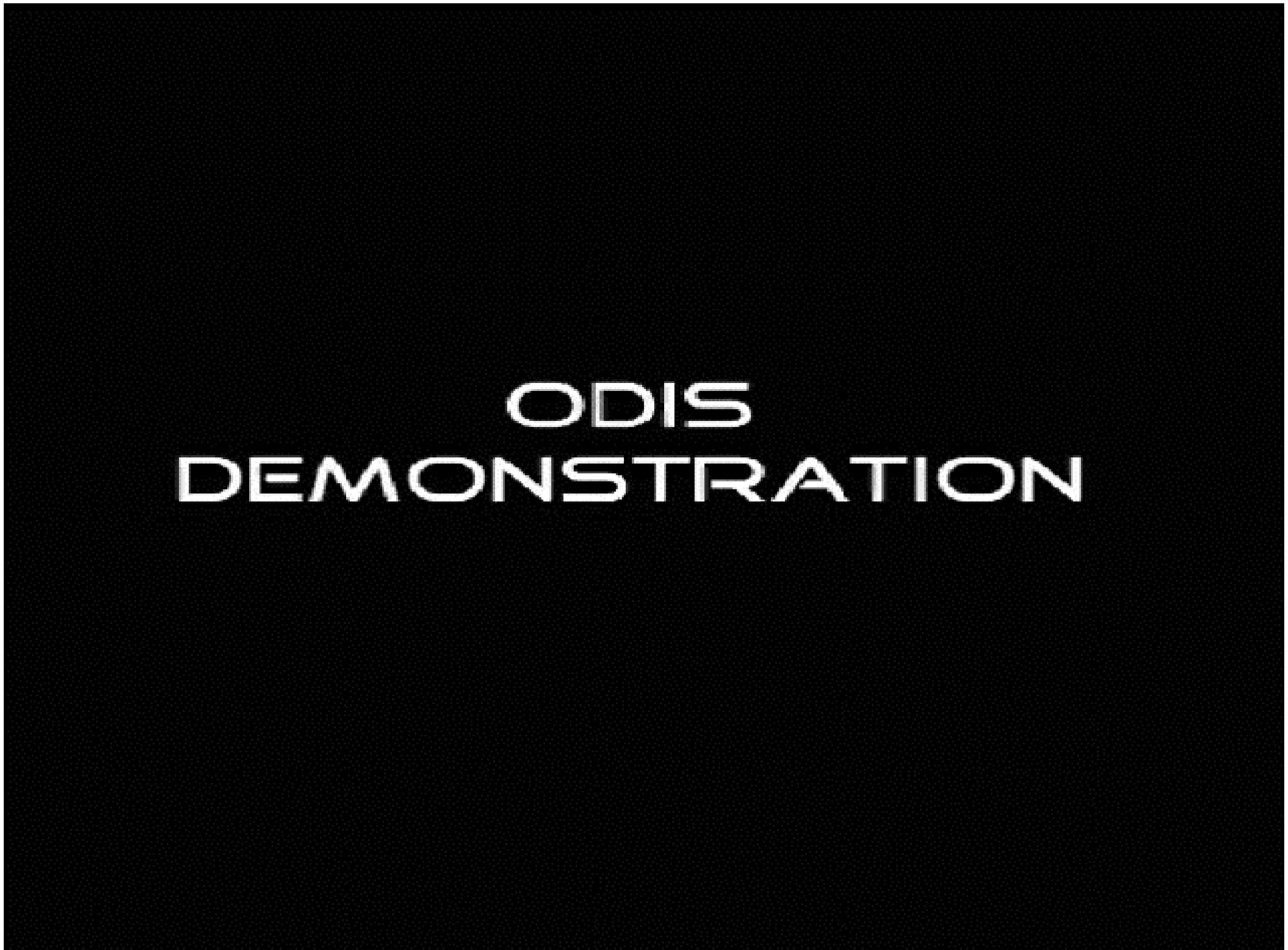

A task
A sub-task
An atomic-task
A Command-unit
Localize mission
Re-plan
Localizing agent
Plan path for the task based on the partial environment knowledge
A task is decomposed into sub-tasks and the sub-tasks are ordered, if necessary by the O&O module
S_1 S_2 S_3 S_4 S_{n-1} S_n
Plan a Reactive path
Sub-tasks may be further Decomposed into atomic-tasks, if they are not realizable in their current form. Atomic-tasks may also be subjected to ordering.
A1
A2
Each Atomic-task get directly mapped to an atomic script, which can consist of several command -units
Atomic-script
Modifications in the traveling velocities for slowing down
Safety and obstacle avoiding agent
Command actions
Feedback loop for the "expected" situations
Environment


User Input
External
Internal
GUI Communicator
Awareness
Localize
Mission
Updated Environment Knowledge
Queries & updates
WD
World Database
Supervisory Task Controller
Un-optimized group of tasks
Ordered group of tasks
Optimization & Ordering Module
Task
States and results of atomic tasks execution
Task
Ordered group of Sub-tasks & Atomic-tasks
Sensor Processor
Filtered & Perceived input
World Model Predictor
Predicted changes in the environment
Behavior Generator & Atomic-Task Executor
E-Stop
Joy-stick
Command-Units
Control Supervisor (CS)
Observed input
Command Actions
Resources
Sensors
Laser
Sonar
Camera
IR
Actuators
wheels
Internal
External
Events
Environment
Actions

Awareness Thread

Comment to Grad Students: Carefully consider when your advisor asks you to do something dangerous!!

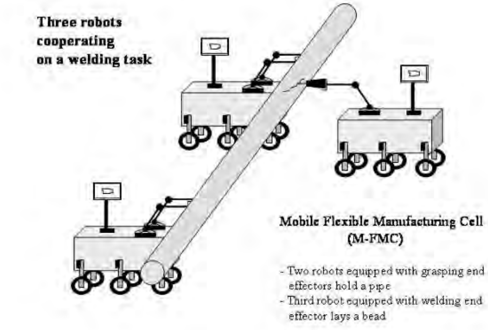
Example: ODIS FindCar() Script



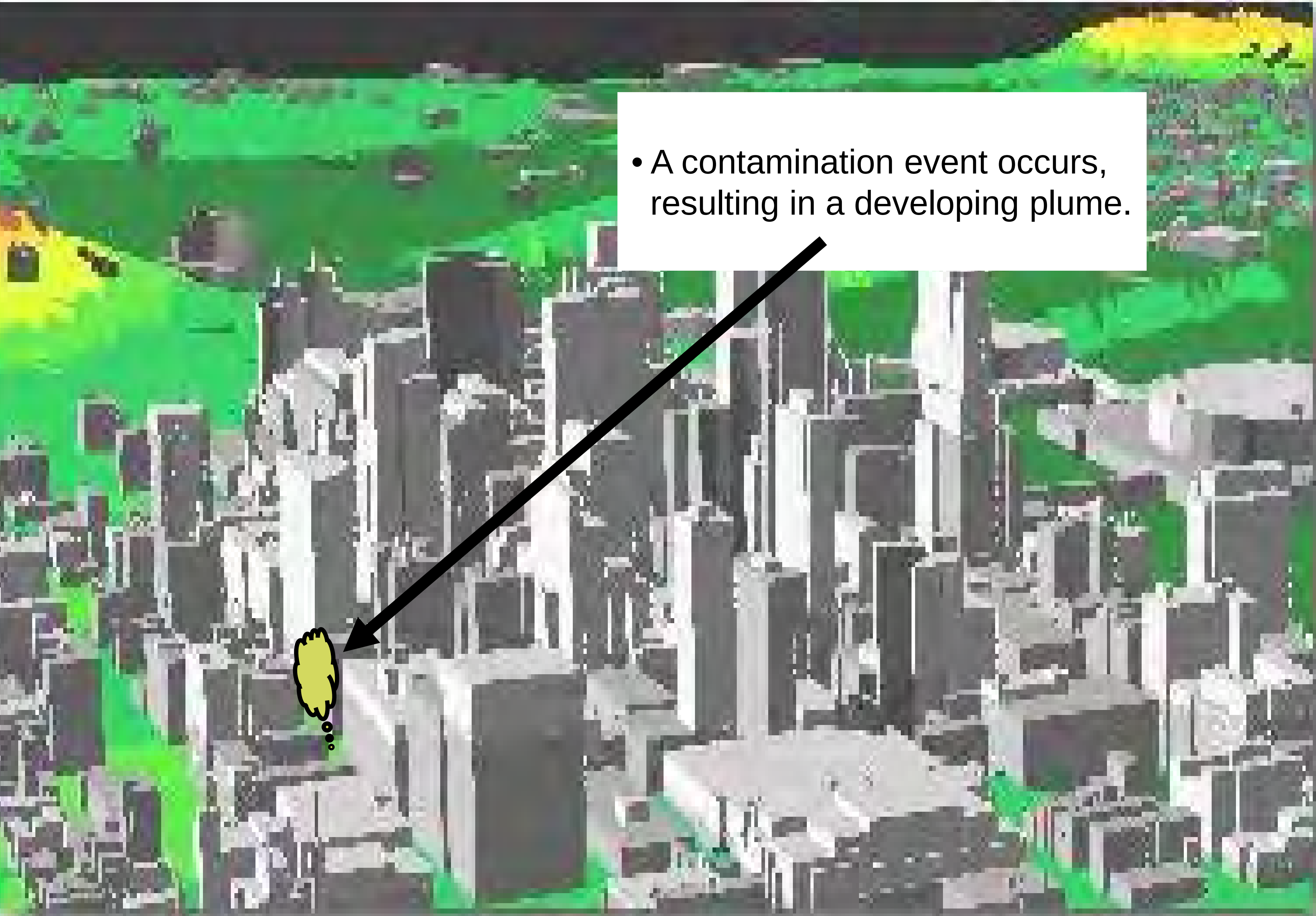
From intelligent behavior to cooperative autonomy

Comment 6: It is important to make your work relevant to society's needs – be mission-driven

- For me, a motivation came on 9/11/2001
  - ODIS was one response to that and we were able to commercialize and deploy in theatre
- But another idea caught my attention
  - Responding to a hazardous response
  - Question: If you can make one robot "intelligent," can you get two "intelligent" robots to cooperate?

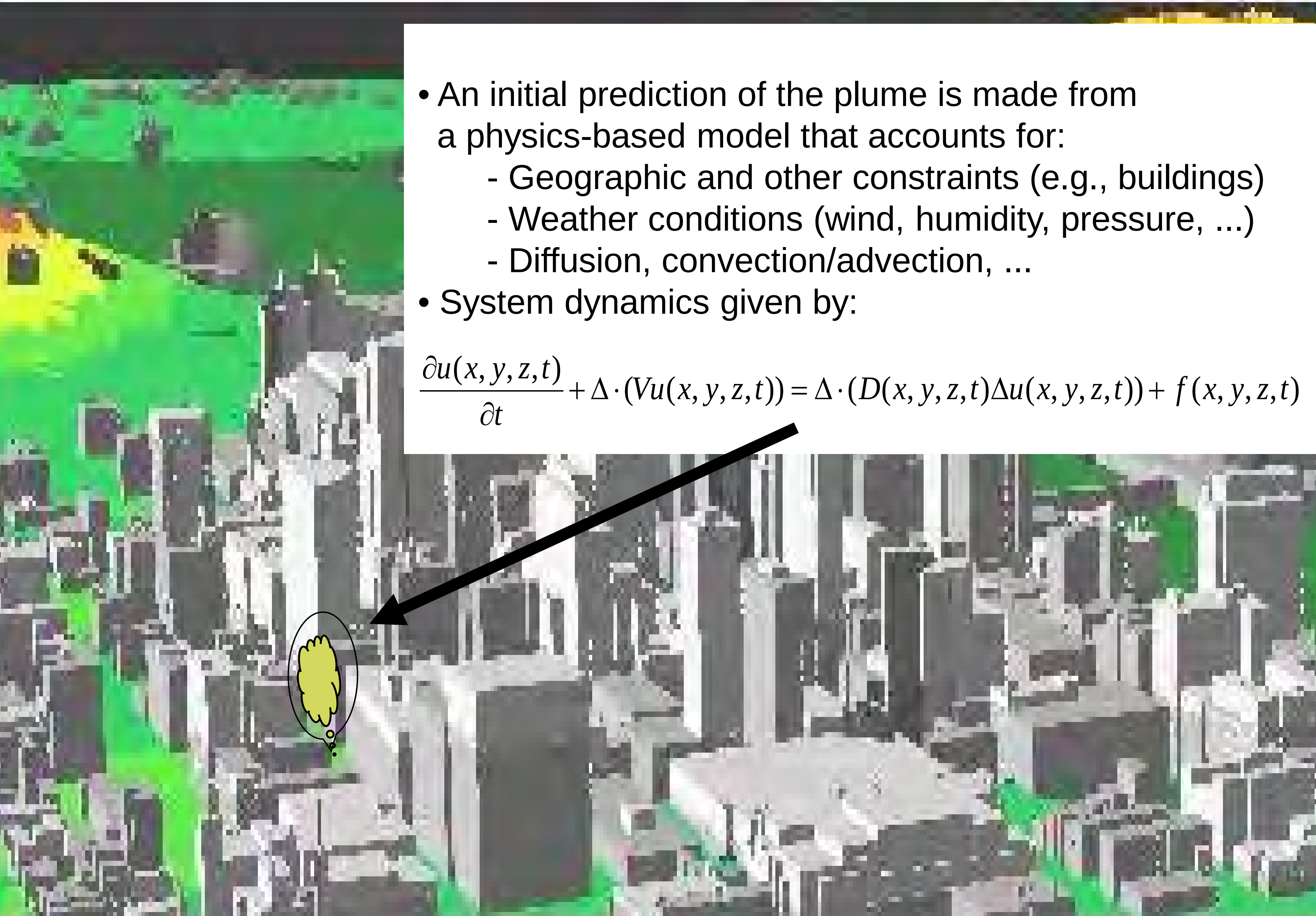


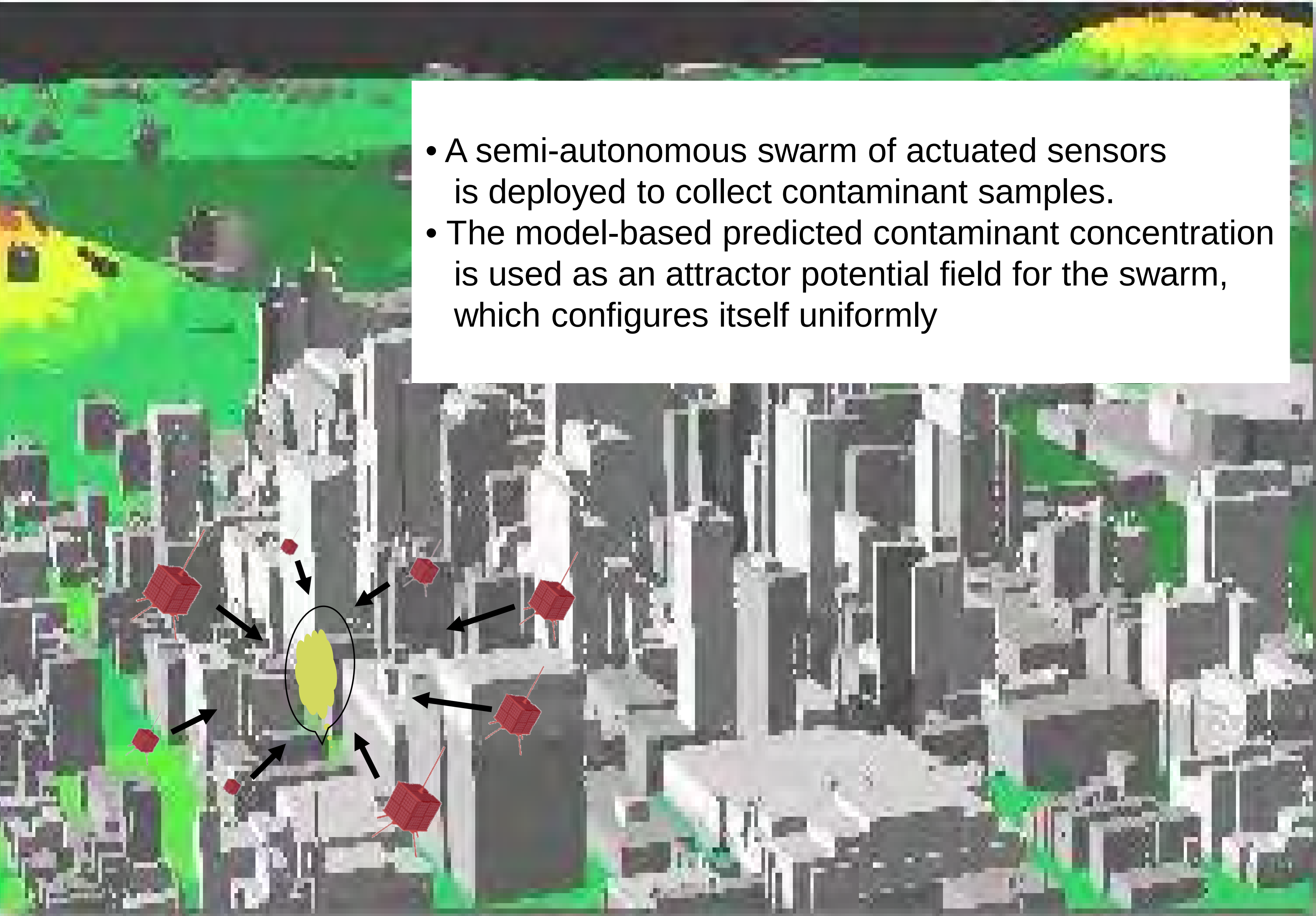
Comment 7: You should always be thinking about the extension of your ideas


• A contamination event occurs, resulting in a developing plume.

- An initial prediction of the plume is made from a physics-based model that accounts for:
  - Geographic and other constraints (e.g., buildings)
  - Weather conditions (wind, humidity, pressure, ...)
  - Diffusion, convection/advection, ...
- System dynamics given by:

$$\frac{\partial u(x,y,z,t)}{\partial t} + \Delta \cdot (Vu(x,y,z,t)) = \Delta \cdot (D(x,y,z,t)\Delta u(x,y,z,t)) + f(x,y,z,t)$$

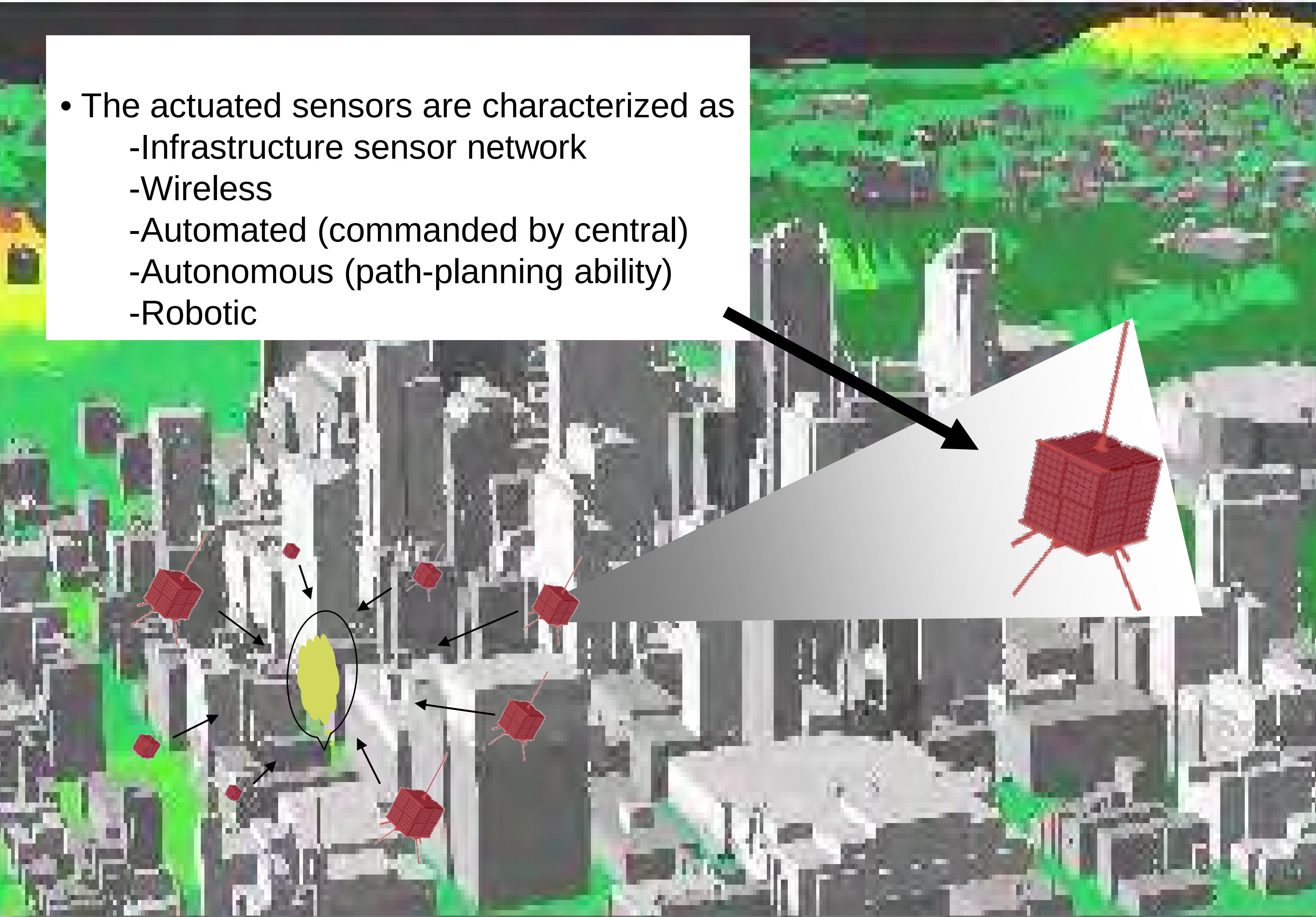


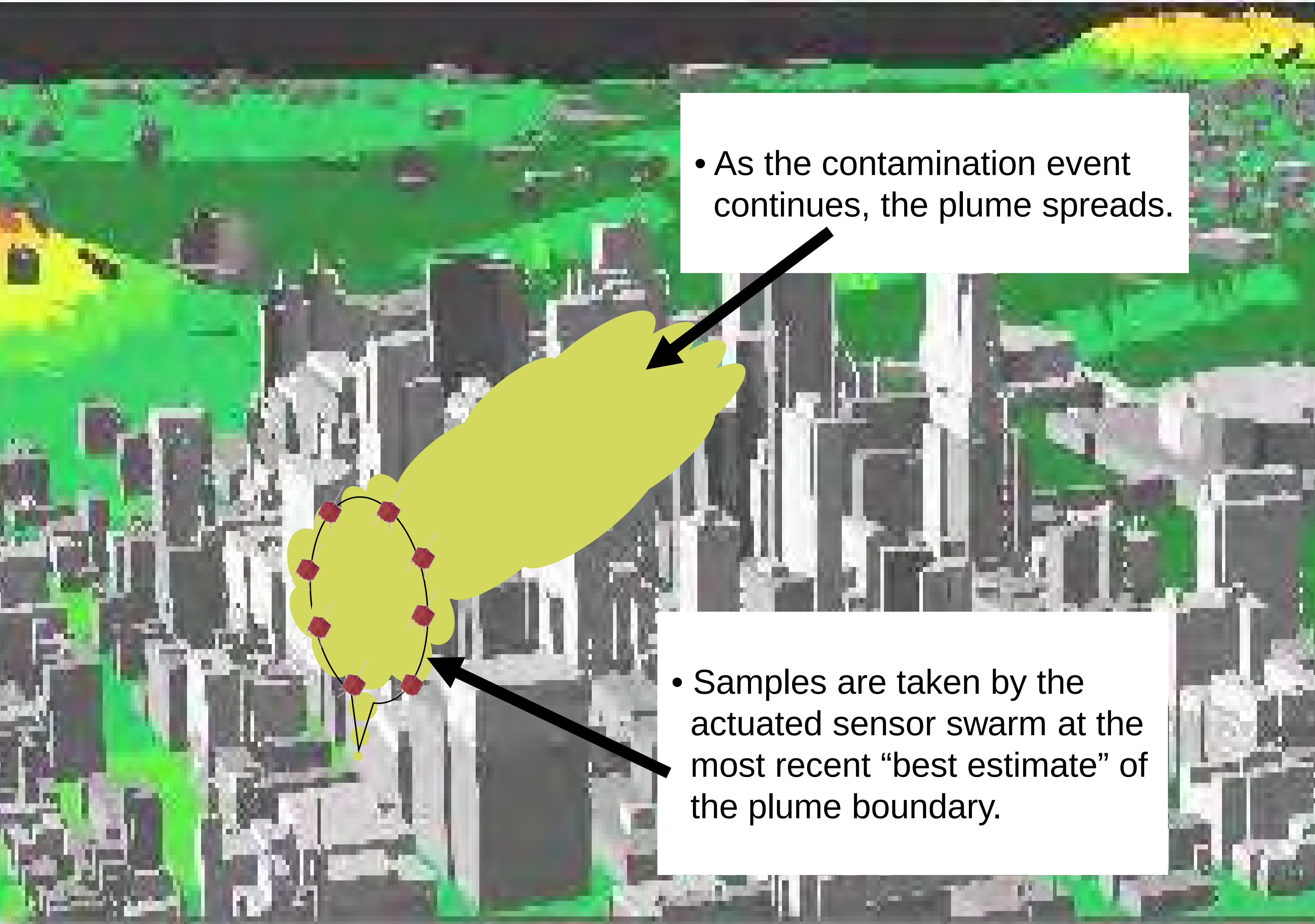


• A semi-autonomous swarm of actuated sensors is deployed to collect contaminant samples.

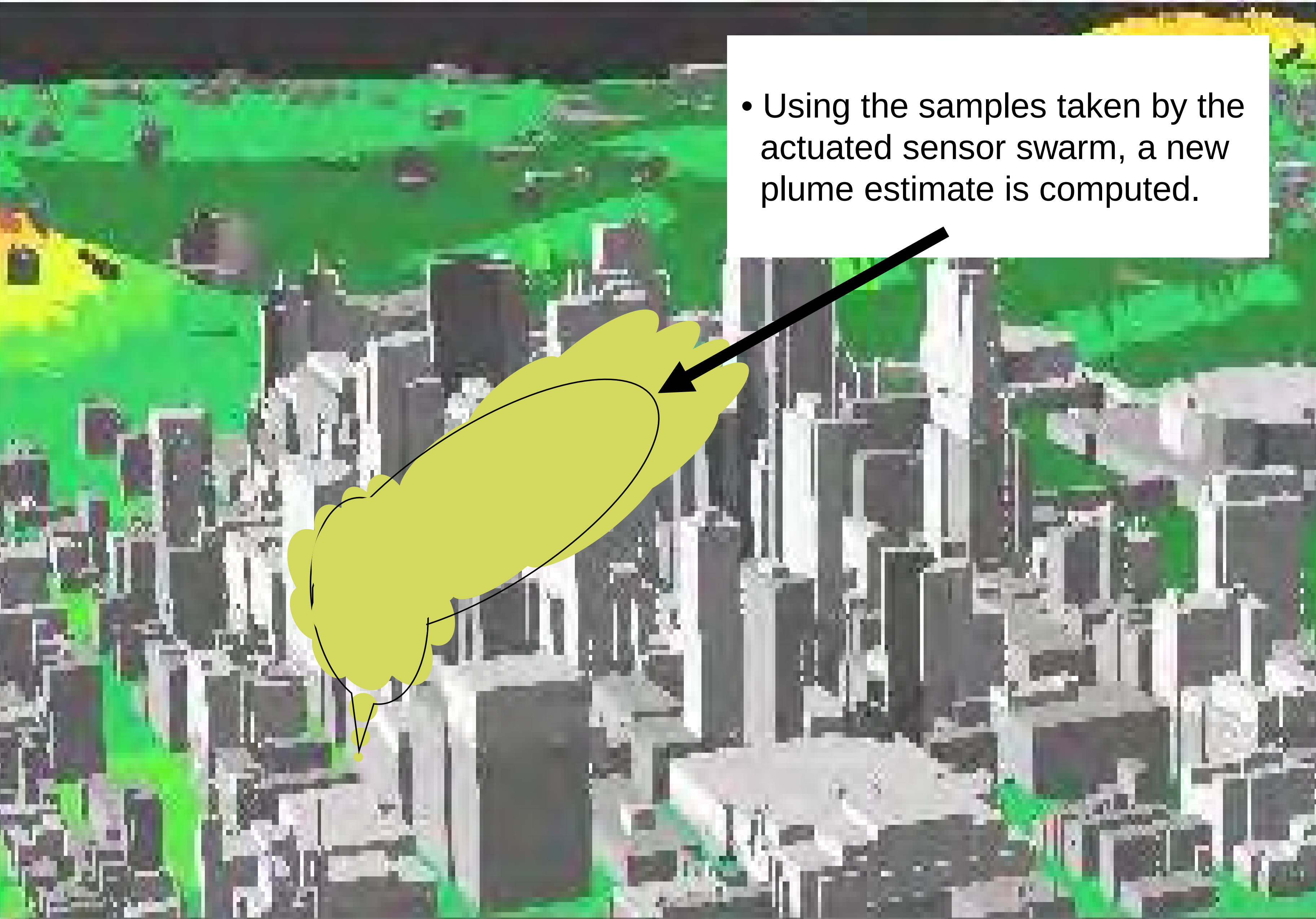
• The model-based predicted contaminant concentration is used as an attractor potential field for the swarm, which configures itself uniformly

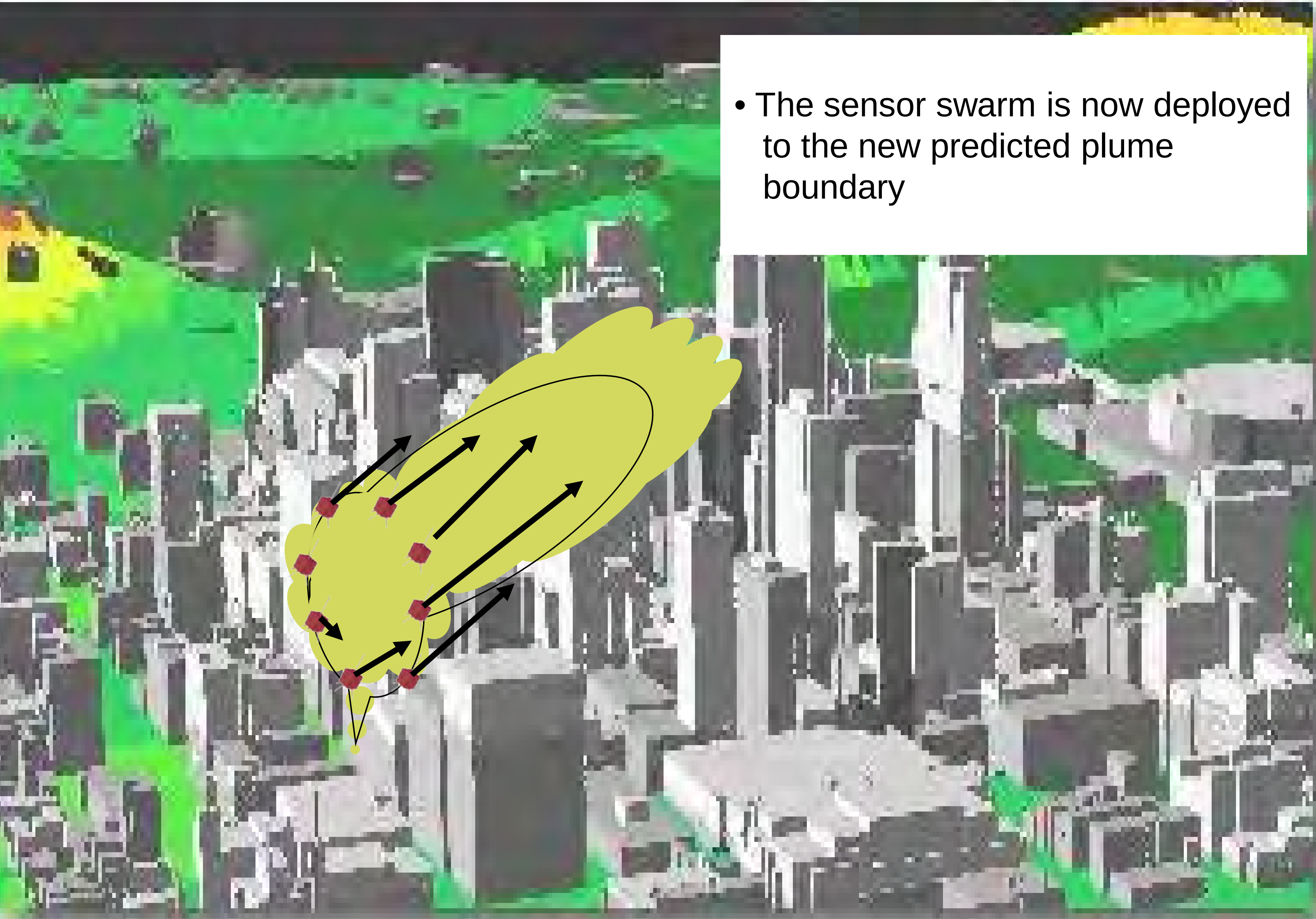
- The actuated sensors are characterized as
  - -Infrastructure sensor network
  - -Wireless
  - -Automated (commanded by central)
  - -Autonomous (path-planning ability)
  - -Robotic



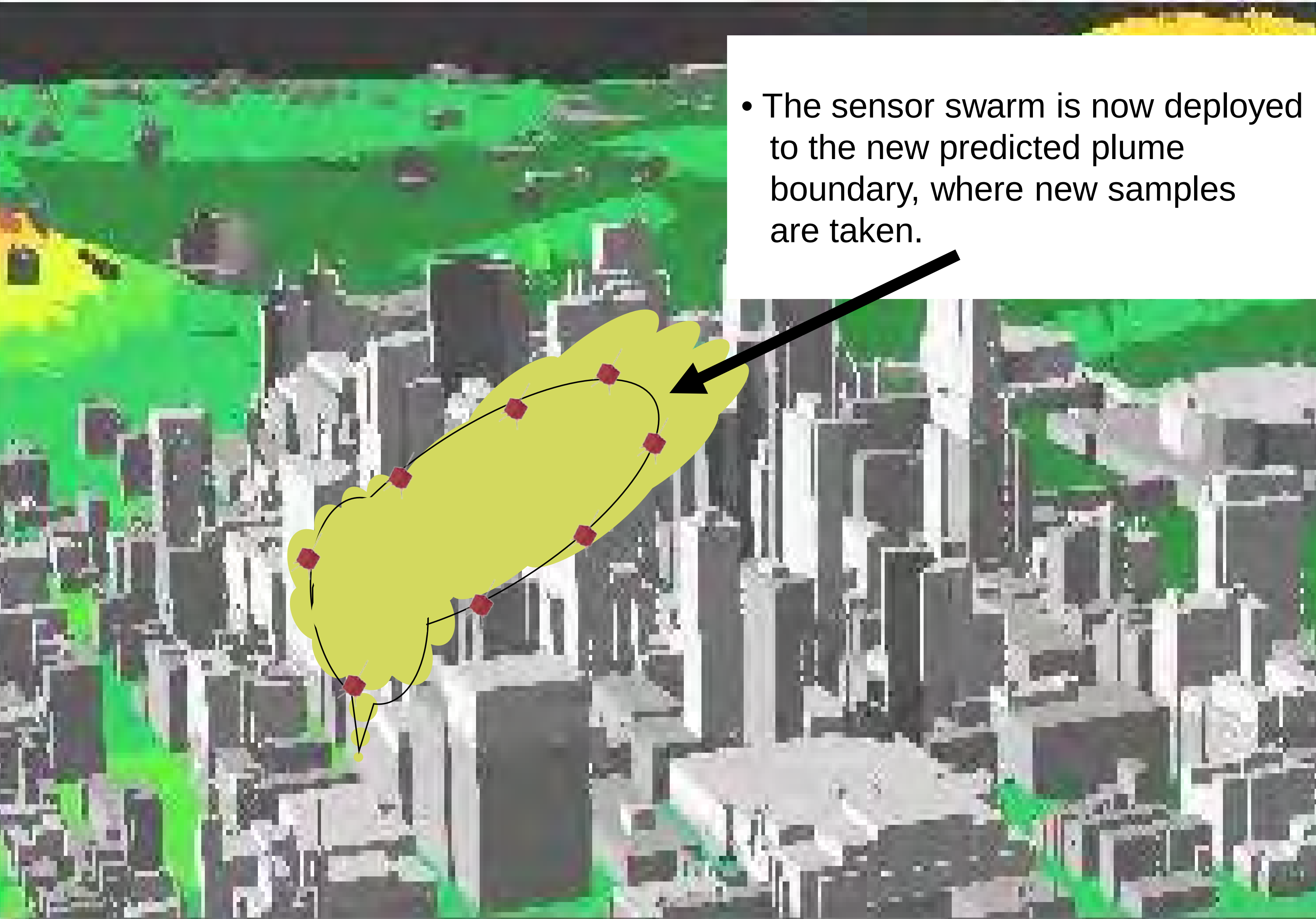


• As the contamination event continues, the plume spreads.
• Samples are taken by the actuated sensor swarm at the most recent “best estimate” of the plume boundary.

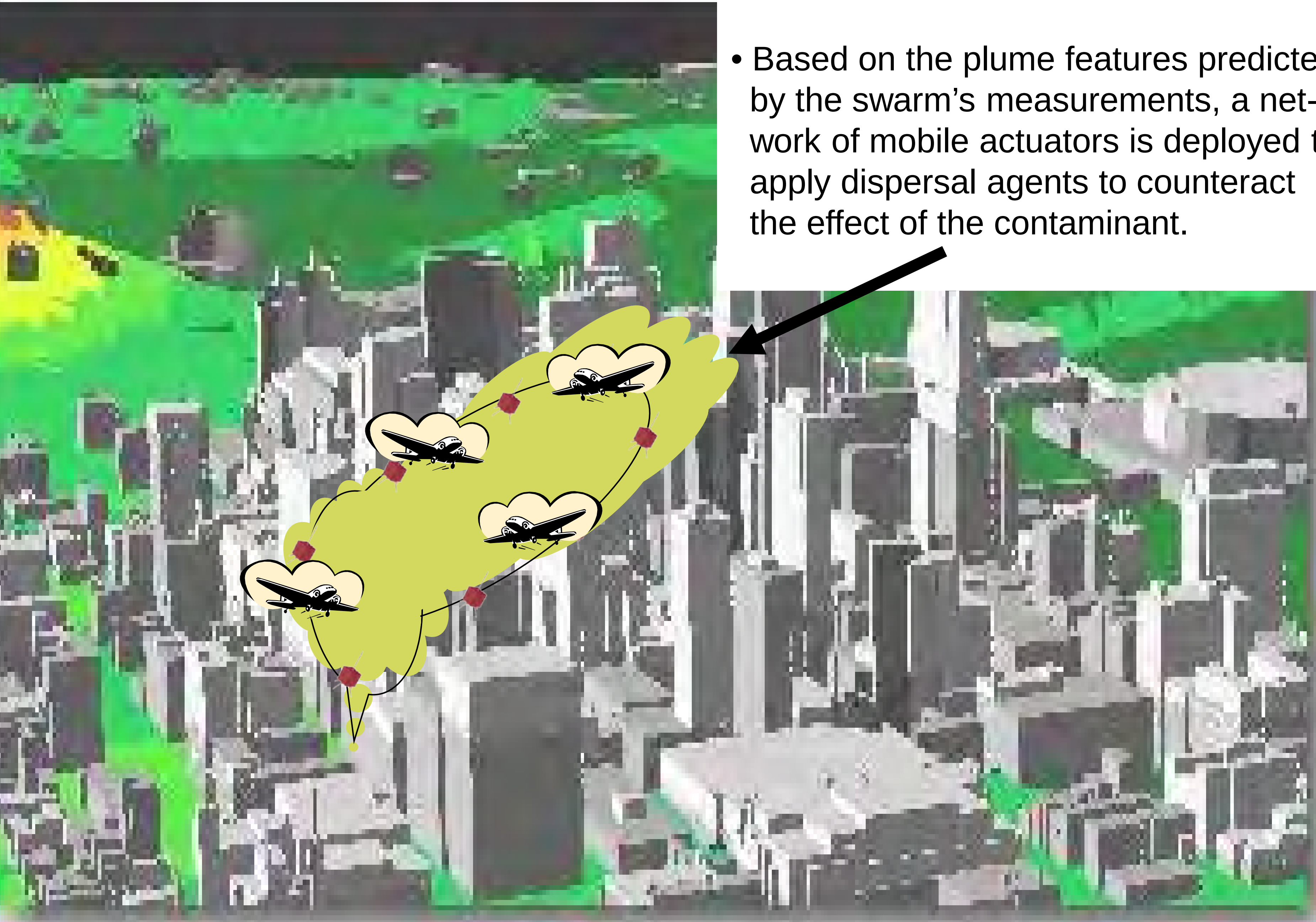

• Using the samples taken by the actuated sensor swarm, a new plume estimate is computed.



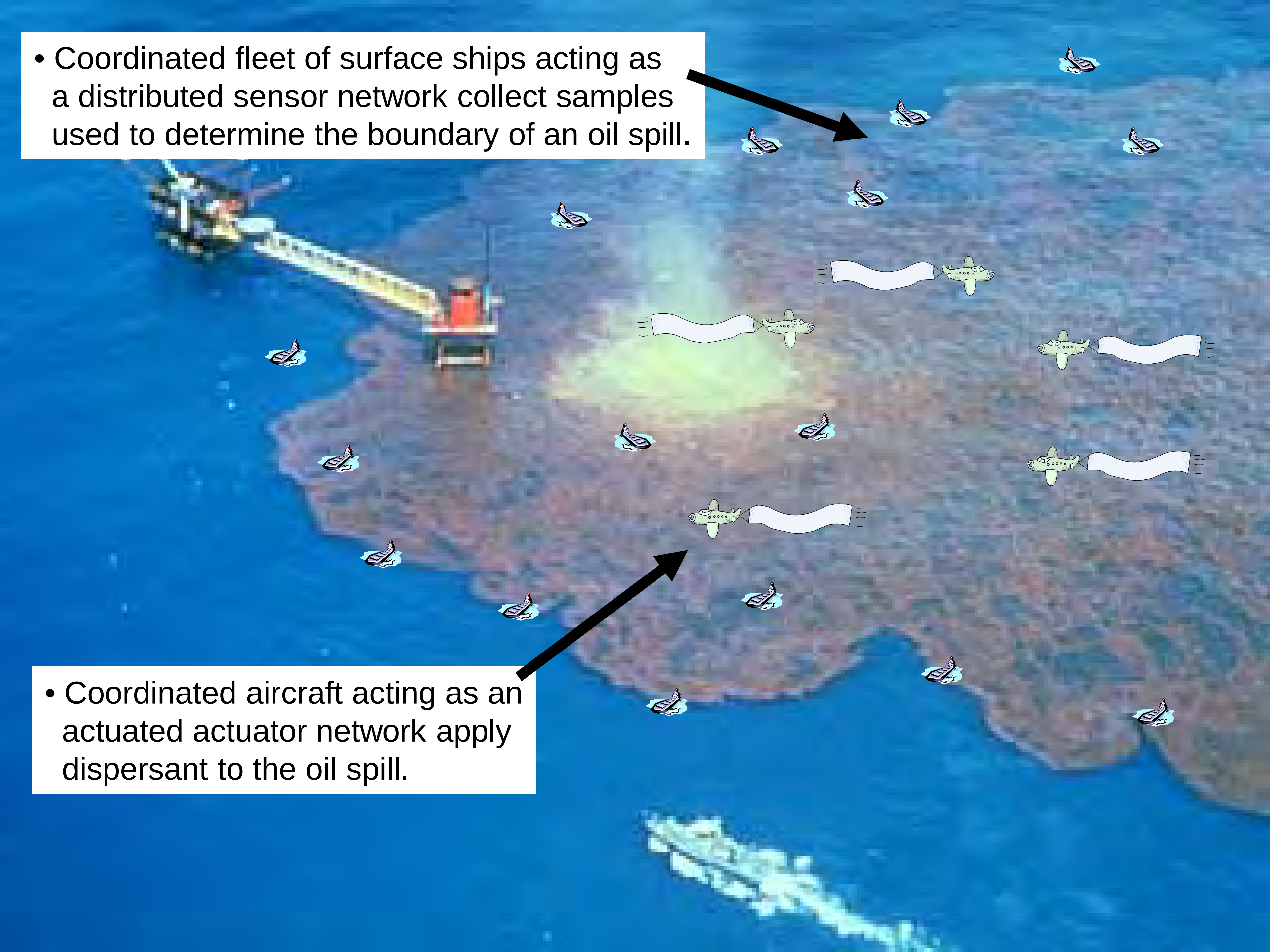
• The sensor swarm is now deployed to the new predicted plume boundary

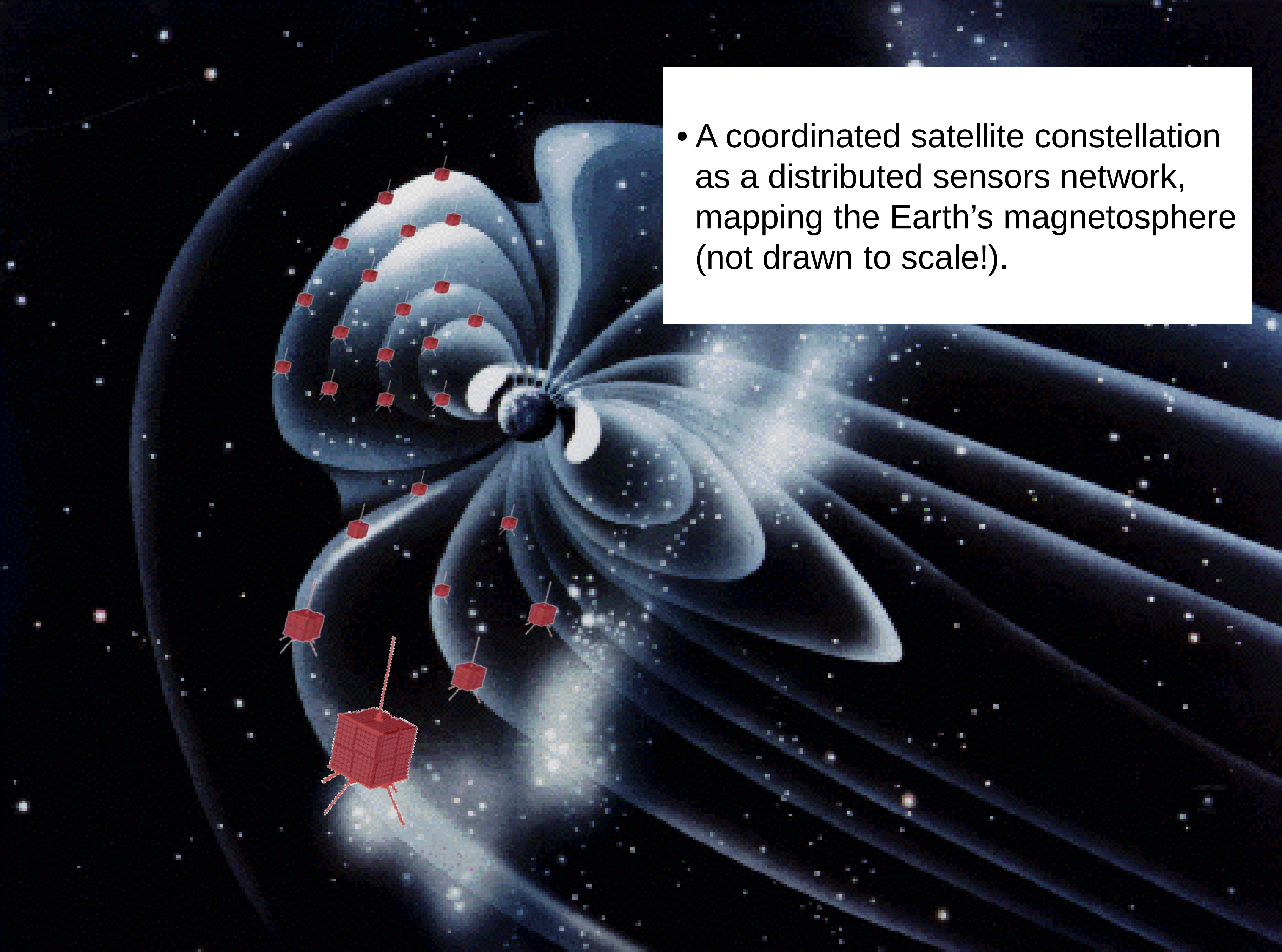


• The sensor swarm is now deployed to the new predicted plume boundary, where new samples are taken.



• Based on the plume features predicted by the swarm’s measurements, a network of mobile actuators is deployed to apply dispersal agents to counteract the effect of the contaminant.


• Coordinated fleet of surface ships acting as a distributed sensor network collect samples used to determine the boundary of an oil spill.
• Coordinated aircraft acting as an actuated actuator network apply dispersant to the oil spill.



- A coordinated satellite constellation as a distributed sensors network, mapping the Earth's magnetosphere (not drawn to scale!).

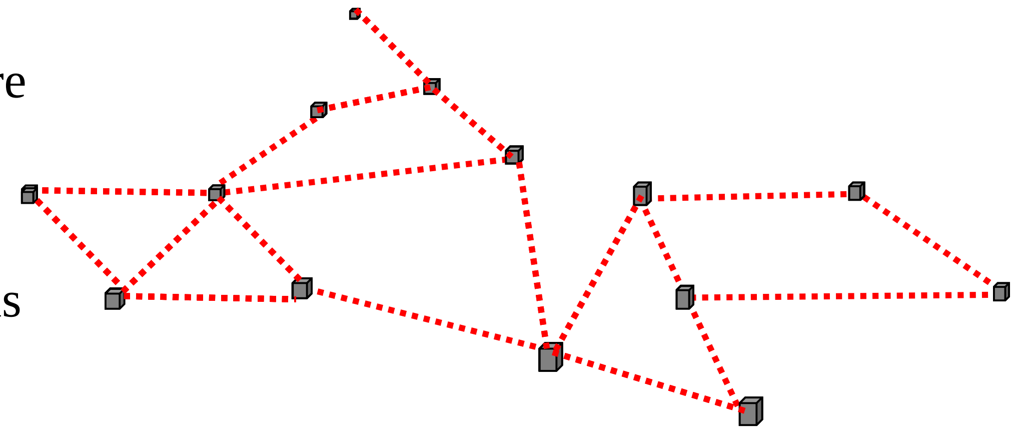
Coordination and Control of Distributed Networks of Actuated Sensors

Kevin L. Moore, Director

Center for Self-Organizing and Intelligent Systems
Utah State University
Logan, Utah

6 March 2003

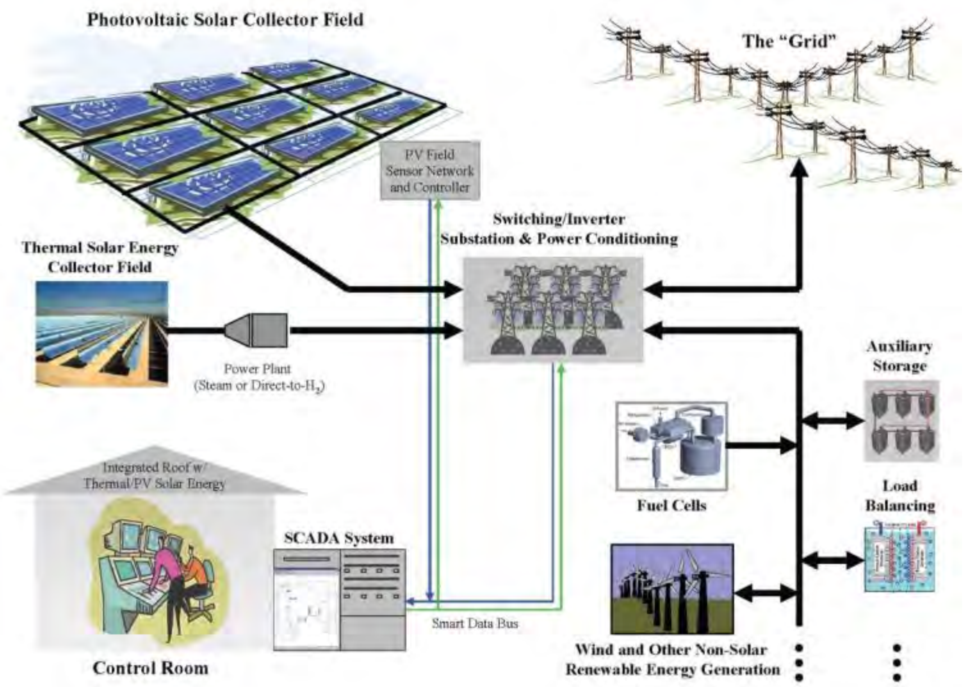
Part 3: Cooperative autonomy



- **Network of “entities”**
  - Communication infrastructure
  - Entity-level functionality
  - Implied global functionality
  - Not necessarily homogeneous
- **Nodes:**
  - Entities could be sensors
  - Entities could be actors (actuators)
  - Entities could be people
- **Dynamic**
  - Entities may or may not be mobile
  - Communication topology might be time-varying
  - Data actively and deliberately shared among entities
  - Decision-making and learning
  - Links between entities might be dynamic systems

Dynamic Networks as Models for …

- **Many systems of interest:**
  – Cooperating robots
  – Buildings, cities
  – Power systems
  – Water distribution
  – Information networks
  – Socio-economic systems
  – …. many more ….



- Need a **framework for analysis and design** of these networks
  – One useful paradigm is the **consensus variable** approach

Consensus Variables

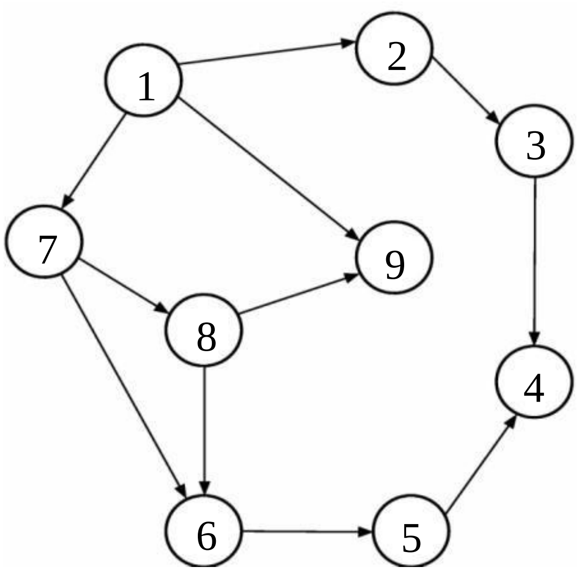
- Suppose we have *N* agents with a shared *global* consensus variable ξ
- Each agent has a *local* value of the variable given as ξ_i
- Ea **Change in value** eir local value based on the values of the agents that they can communicate wi **Difference with neighbors**

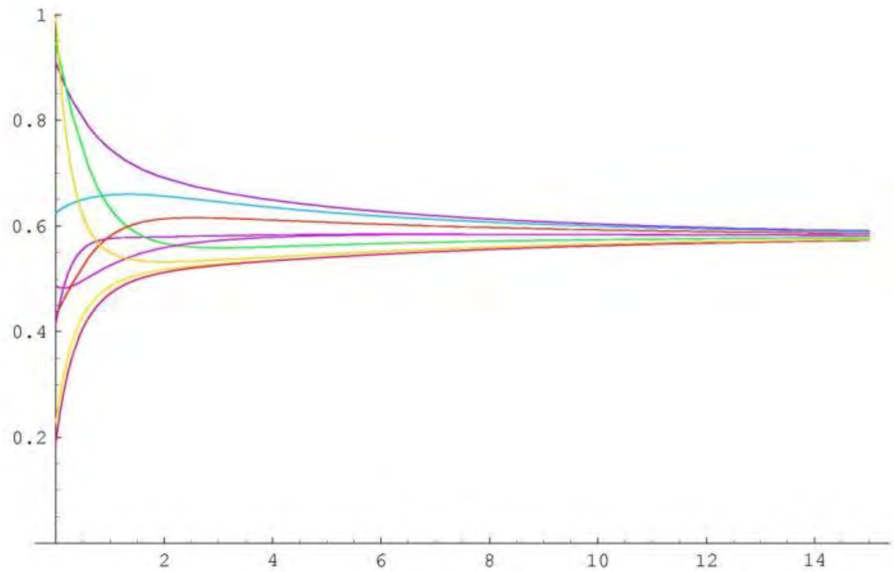
$$\dot{\xi}_i(t) = -\sum_{j=1}^{N} k_{ij}(t) G_{ij}(t) (\xi_i(t) - \xi_j(t))$$

where k_{ij} are gains and G_{ij} defines the communication topology graph of the system of agents

- **Key result** from literature: If the corresponding graph has a spanning tree then $\xi_i \to \xi^*$ for all *i*

Example: Single Consensus Variable





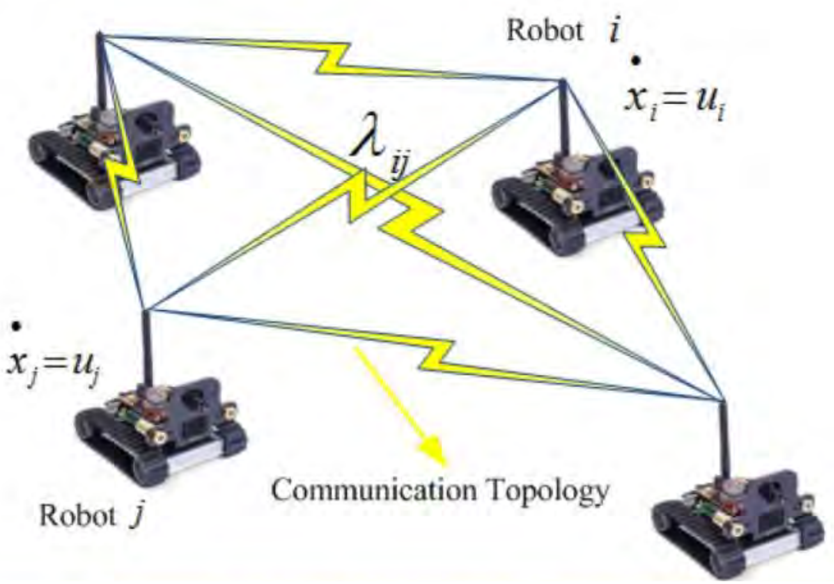
$$\begin{pmatrix} \dot{\xi}_1 \\ \dot{\xi}_2 \\ \dot{\xi}_3 \\ \dot{\xi}_4 \\ \dot{\xi}_5 \\ \dot{\xi}_6 \\ \dot{\xi}_7 \\ \dot{\xi}_8 \\ \dot{\xi}_9 \end{pmatrix} = \underbrace{\begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ k_{21} & -k_{21} & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & k_{32} & -k_{32} & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & k_{43} & -k_{43}-k_{45} & k_{54} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & -k_{56} & k_{56} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & -k_{67}-k_{68} & k_{67} & k_{68} & 0 \\ k_{71} & 0 & 0 & 0 & 0 & 0 & -k_{71} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & k_{87} & -k_{87} & 0 \\ k_{91} & 0 & 0 & 0 & 0 & 0 & 0 & k_{98} & -k_{91}-k_{98} \end{bmatrix}}_{\text{Laplacian Matrix}} \begin{pmatrix} \xi_1 \\ \xi_2 \\ \xi_3 \\ \xi_4 \\ \xi_5 \\ \xi_6 \\ \xi_7 \\ \xi_8 \\ \xi_9 \end{pmatrix}$$

Consensus by design

- Consider 4 robots with the velocity $\dot{x}_i = u_i$ and interconnected using the following static consensus protocol:

$$\dot{x}_i = u_i = -\sum_{j \in \mathcal{N}_i} [\lambda_{ij}(x_i - x_j)].$$

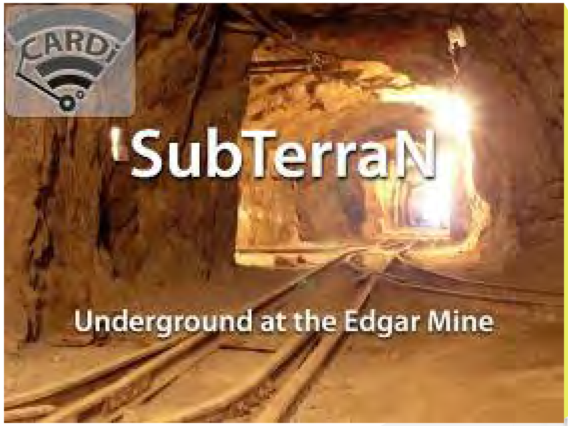
Then $$\dot{X} = -LX,$$
where L is the static Laplacian associated with the communication topology of the 4 robots graph.



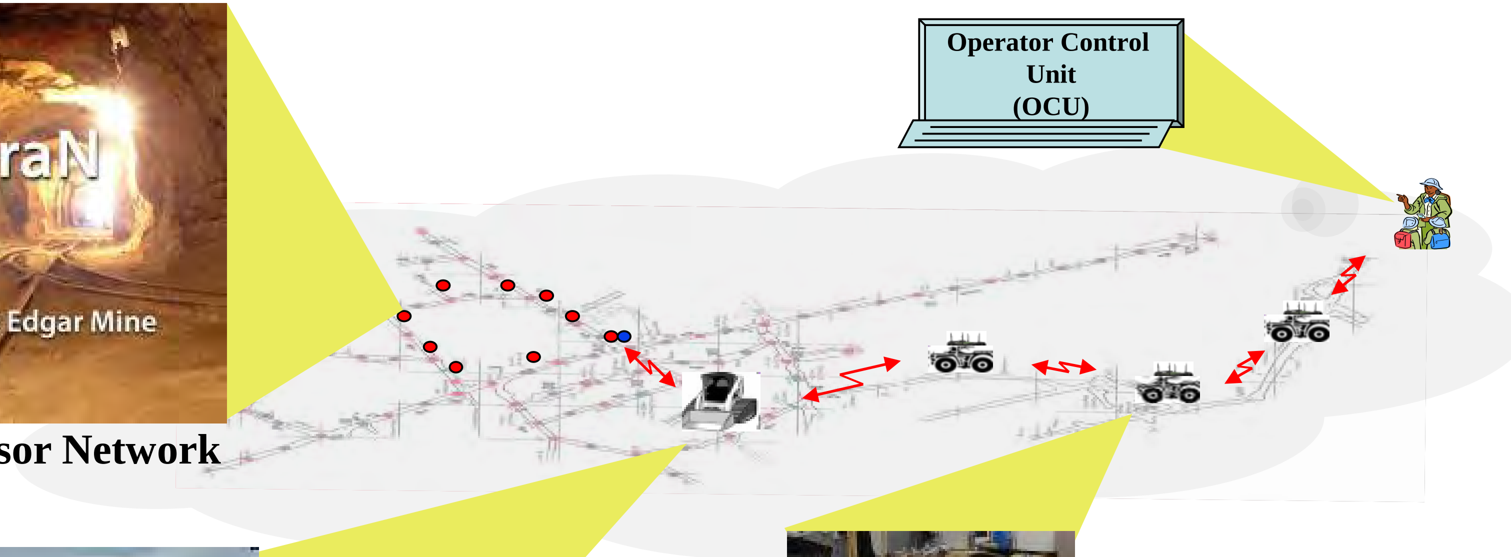
Example: Consensus with a Leader

(movies thanks to USU CSOIS, Wei Ren and YangQuan Chen)

MineSENTRY - Autonomous Mobile Radio Relays using 2nd-order consensus



Underground Sensor Network



Teleoperated Bobcat

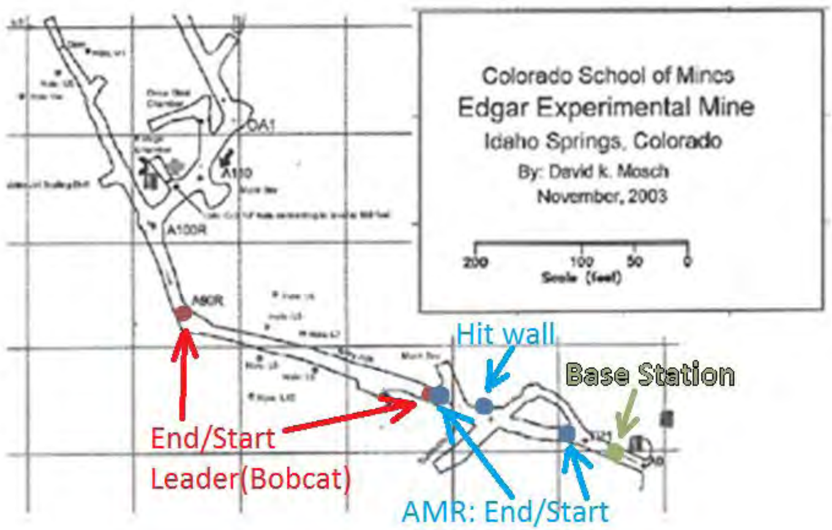
Mesh Radio System
(Rajant Breadcrumb™)

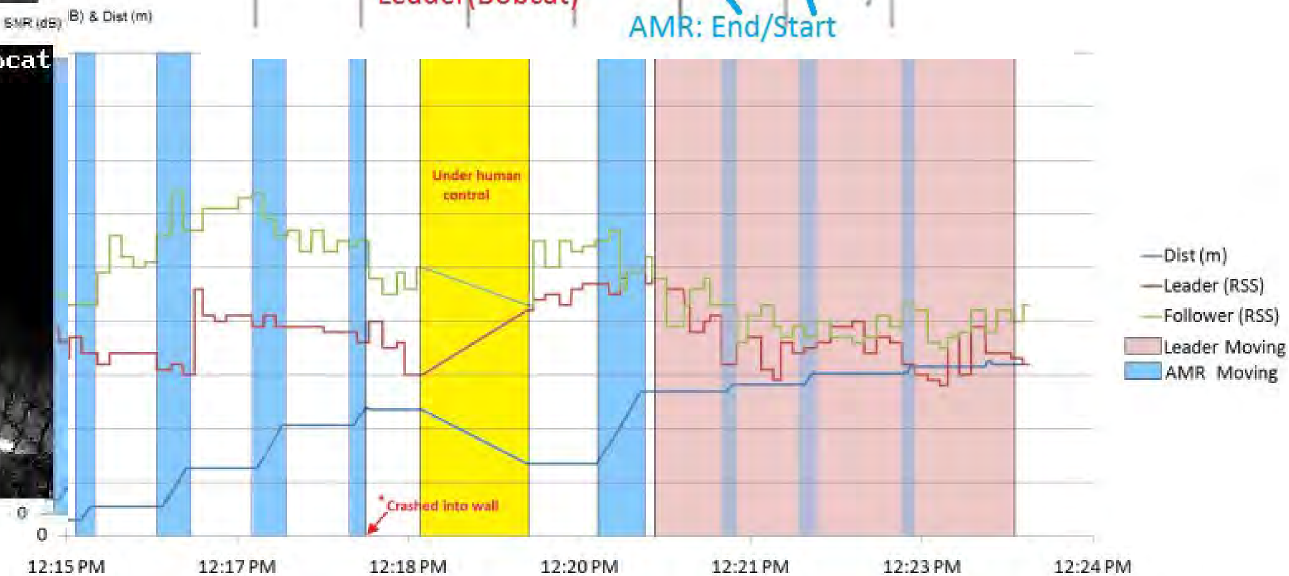
Autonomous Radio Node (AMR)
- Provides Communication Tether
- Uses CSM-developed UGV Autopilot

Experimental Results

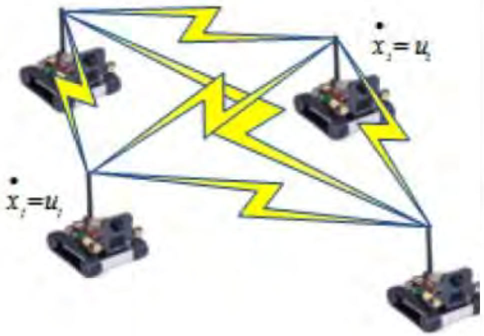
MineSENTRY
Mine Safety and Rescue through Sensing Networks and Robotics Technology

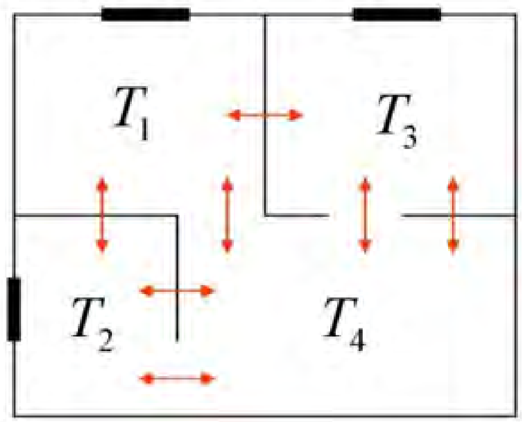
COLORADO SCHOOL OF MINES



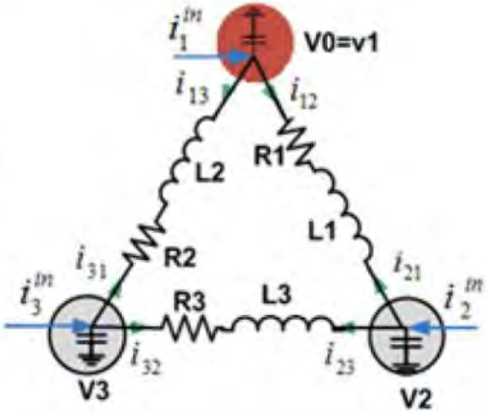


Consensus models many real systems





$$C_i^r \frac{dT_i}{dt} = q_i^{in}(t) - \sum_{j \in \mathcal{N}_i} q_{ij}(t)$$



Robot network:

$$sX(s) = -LX(s)$$

Thermal network:

$$sT(s) = -L(s)T(s)$$

Electrical network:

$$sV(s) = -L(s)V(s)$$

Static consensus protocol:

$$x_i(s) = -\frac{1}{s} \sum_{j \in \mathcal{N}_i} [\lambda_{ij}(x_i(s) - x_j(s))]$$

Dynamic consensus protocol:

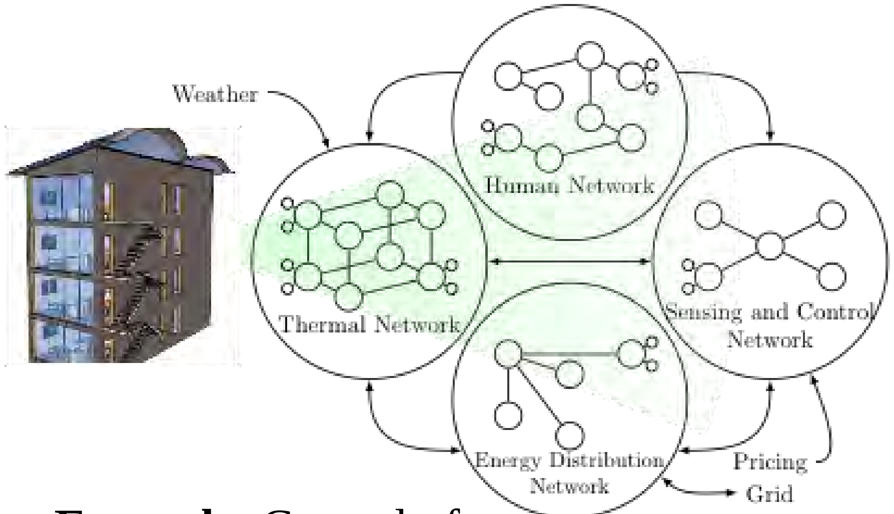
$$x_i(s) = -\frac{1}{s} \sum_{j \in \mathcal{N}_i} [\lambda_{ij}(s)(x_i(s) - x_j(s))]$$

Consensus conditions: Under some conditions on L and $L(s)$,

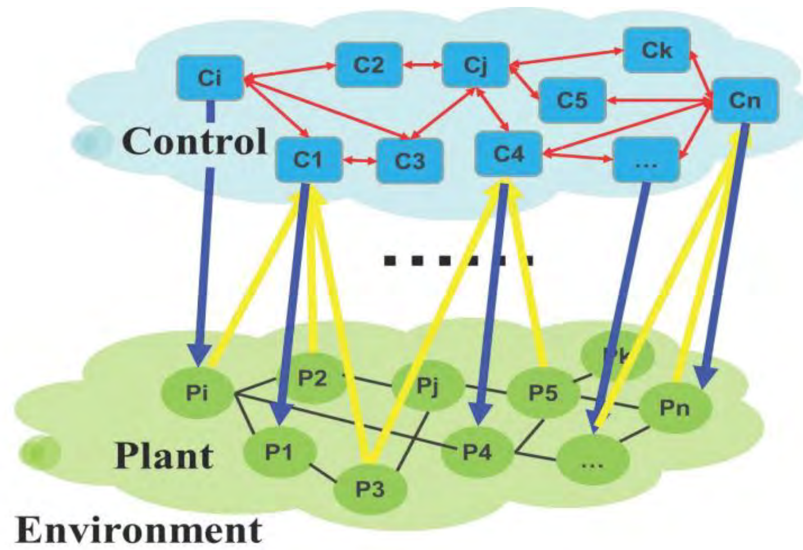
$$x_i(t) \to x^* \qquad T_i(t) \to T^* \qquad v_i(t) \to v^*$$

Introduces many new, exciting topics

- **Complex cyber-physical systems:** physical processes, interacting with other sources of energy, material, and information
  - Can often be modelled as **networks controlled by networks**



Example: Control of Thermal Processes in a building



Completing the circle – LSS and ILC

- Recall, I wanted to study large-scale systems as a grad student
  - Now I am!
  - Repeating:

Comment 4: Things sometimes lead you back to where you started

- And, recently I've also been able to connect LSS with ILC

"Learning to Cooperate: Networks of Formation Agents with Switching Topologies," Deyuan Meng and Kevin L. Moore, *Automatica*, vol. 64, pp 278-293, February 2016.

  - Paper uses ILC to model how a marching band learns its routine
  - Repeating:

Comment 3: Be patient and look for the connections

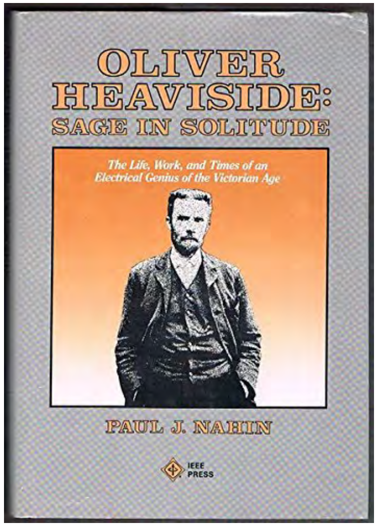
Concluding Comments: To research or not to research …

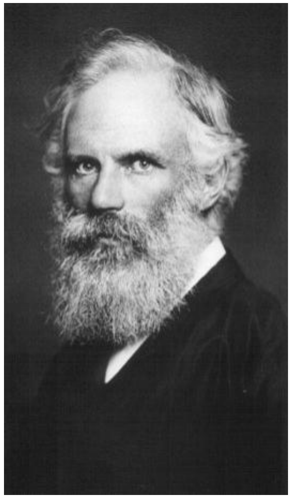
- Of course we should do research!
  - It is our right as academics
  - It’s why we got or are getting a PhD
  - It’s important to society
- But … research in the academy is not what (we think) it used to be
  - State budgets are declining
  - Research does not pay for itself
  - There is not enough support from society to pay
  - Academics are under demand to be more productive and to “pay for ourselves”

This is not a new debate …

- Recall that **Oliver Heavyside** introduced the **Heavyside transform**, the precursor of the Laplace transform as we know and use it
- He was criticized in the journal *Nature* for doing such esoteric research
- In his defense, in a Letter to the Editor of *Nature* in 1892, Thermodynamicist **George Francis Fitzgerald** wrote

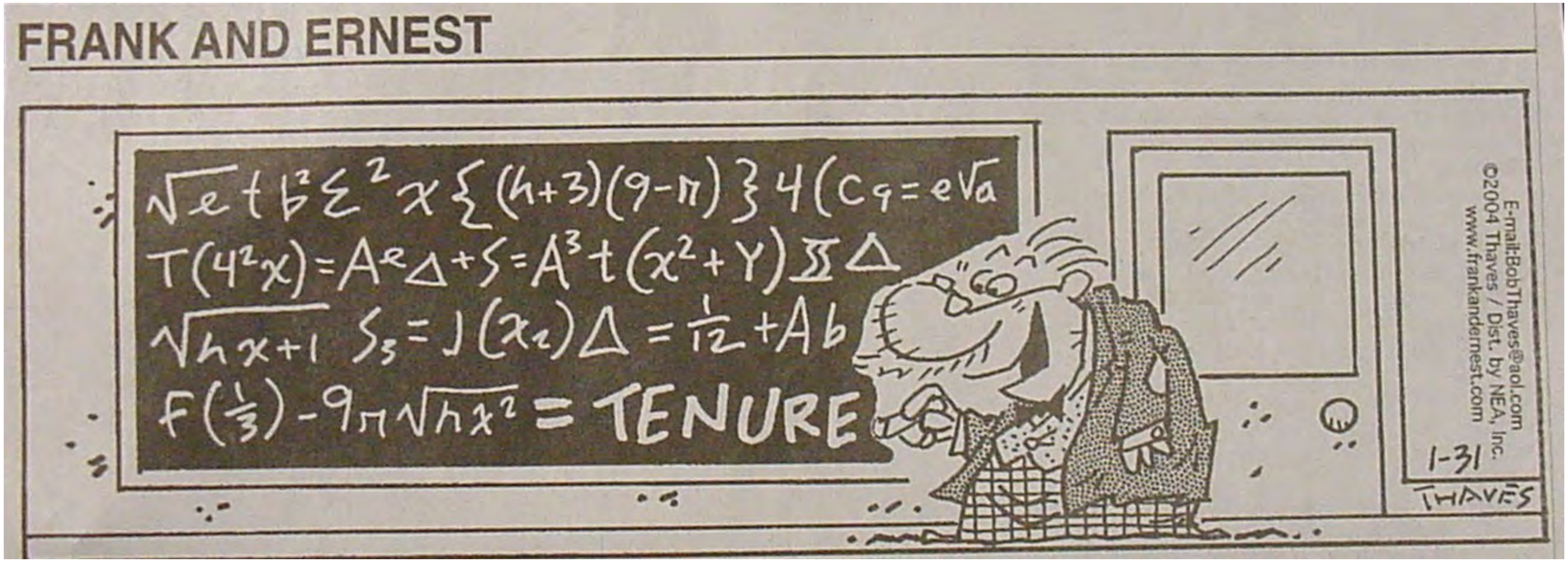
"… if Universities do not study useless subjects, who will?"





And, it’s an on-going debate …

- From Jan 31, 2003



- I tell students, the “P” in “PhD” does not stand for “practical”
- And, honestly, that fact creates a problem

The perils of applied, useful work …

"There I was, walking on the bridge between theory and practice, and suddenly, out of nowhere, comes this gigantic, strongly positive, self-adjoint operator with a dense domain, and smashes the bridge all to pieces! What a nightmare!"

http://controlcartoons.com/

So, what to do…

- To a significant degree, I agree with Fitzgerald, and feel you should: follow your interests, even if useless!
- But, earlier I said: be relevant and do projects that matter
- This is not a conflict:
- My own research has followed the pattern of "Practical-Theory-Practical"
  – I am pragmatic enough to know that policymakers and the public don't want to fund "useless things."
  – So, I shape my programs so I can do theory on the side while doing practice.
  – I call this the "Engineering Method" as opposed to the scientific method

Practical Problem → **Practical Solution** → **Theoretical Solution** → **New, Practical Problem**

Comment 8: Find a way to do things that are both useful and interesting.

Some final comments on research …

- Again, be relevant
  - Carefully define your constituency
  - Build alliances with people that want your "product"
    - If no one is funding you, citing your work, or paying you royalties, then …
  - Try to make a difference
- Work at the intersection of traditional disciplines
  - Focus on niche, intersectional areas
  - Work in teams, with others
- Don't be all things to all people
  - Do what you are good at
  - Be true to your passion
  - Be intentional

Comment 9: Do work that connects between different fields, at least in applications

Acknowledgments

Professor D. Subbaram Naidu, UMD, formerly Idaho State

– Measurement and Control Engineering Research Center

Professor Nicholas Flann, Utah State

– Center for Self-Organizing and Intelligent Systems

Professor YangQuan Chen, UC-Merced

– The MESA Lab

David Watson, David Schiedt

Dr. I-Jeng Wang, Dr. Dennis

– Johns Hopkins Appl)

Professors Tyro e, Mines

Dr. Manoj ce

– C utomation and Distributed Intelligence

Profes g Ahn; Kwang-Kyo Oh

– Gan stitute of Technology, Korea

Professor Deyuan Meng

– Beihang University, Beijing, Chinea

Dr. Craig Rieger, Idaho National Laboratory

and many, many students!

Thank you for your attention!

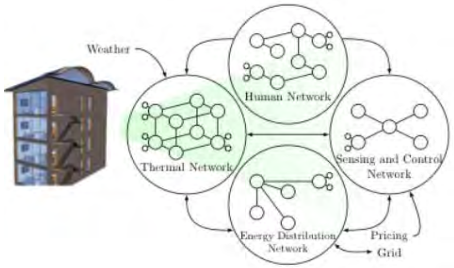
Questions?

Mote-based Robots

UAVs

MineSENTRY Autonomous Mobile Radios

Just for Fun



Building Efficiency Control

Backup Slides

Suggestions for graduate students

Comment 1: You'll do better work if you truly like what you are doing

Comment 2: You should develop your own personal "perspectives" and frameworks for your work

Comment 3: Be patient and look for the connections

Comment 4: Things sometimes lead you back to where you started

Comment 5: You should always be willing to pivot when opportunity arises – it will always lead you somewhere interesting

Comment 6: It is important to make your work relevant to society's needs – be mission-driven

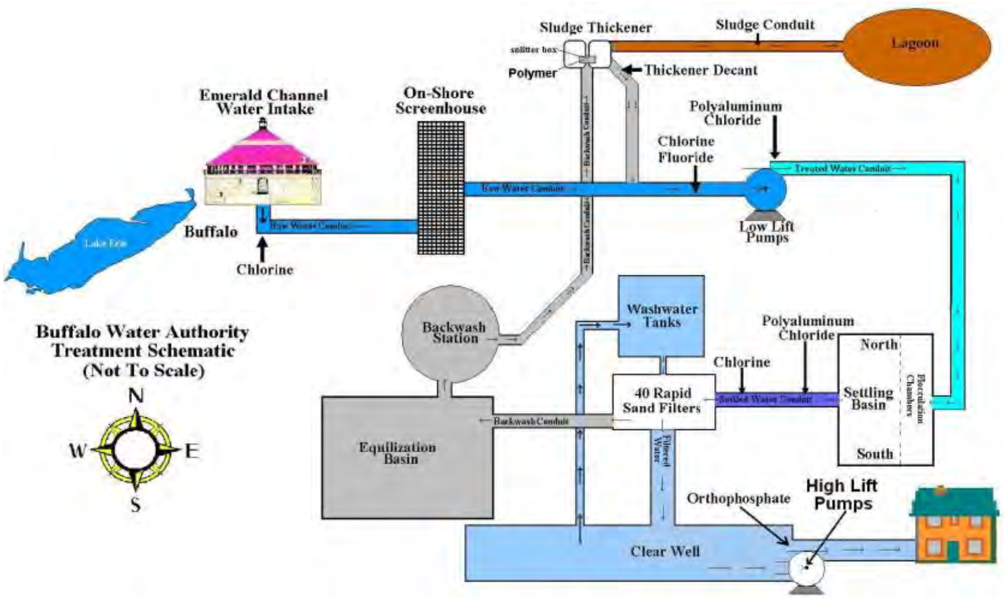
Comment 7: You should always be thinking about the extension of your ideas

Comment 8: Find a way to do things that are useful and interesting.

Comment 9: Do work that connects between different fields, at least in applications

Motivation -1

- Many important systems can be modeled as
  - Collection or network of **integrating agents or subsystems**
  - **Exchanging energy, material, or information**
  - According to some **protocol or physical laws**
  - Subject to an
    - **interconnection topology**
    - **cooperative goal**

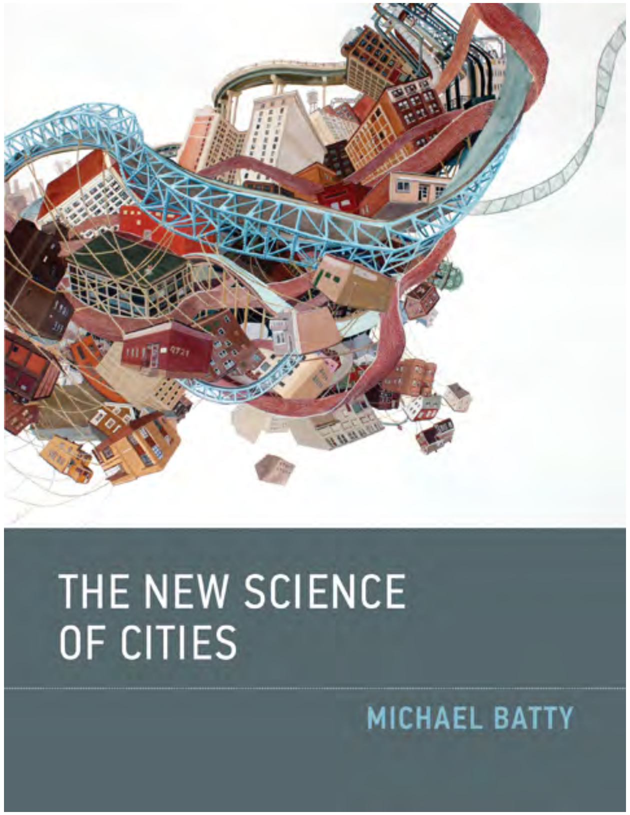


Example: Water treatment system

Figure: from http://www.buffalowater.org/files/Schematic-2011.JPG. Used without permission.

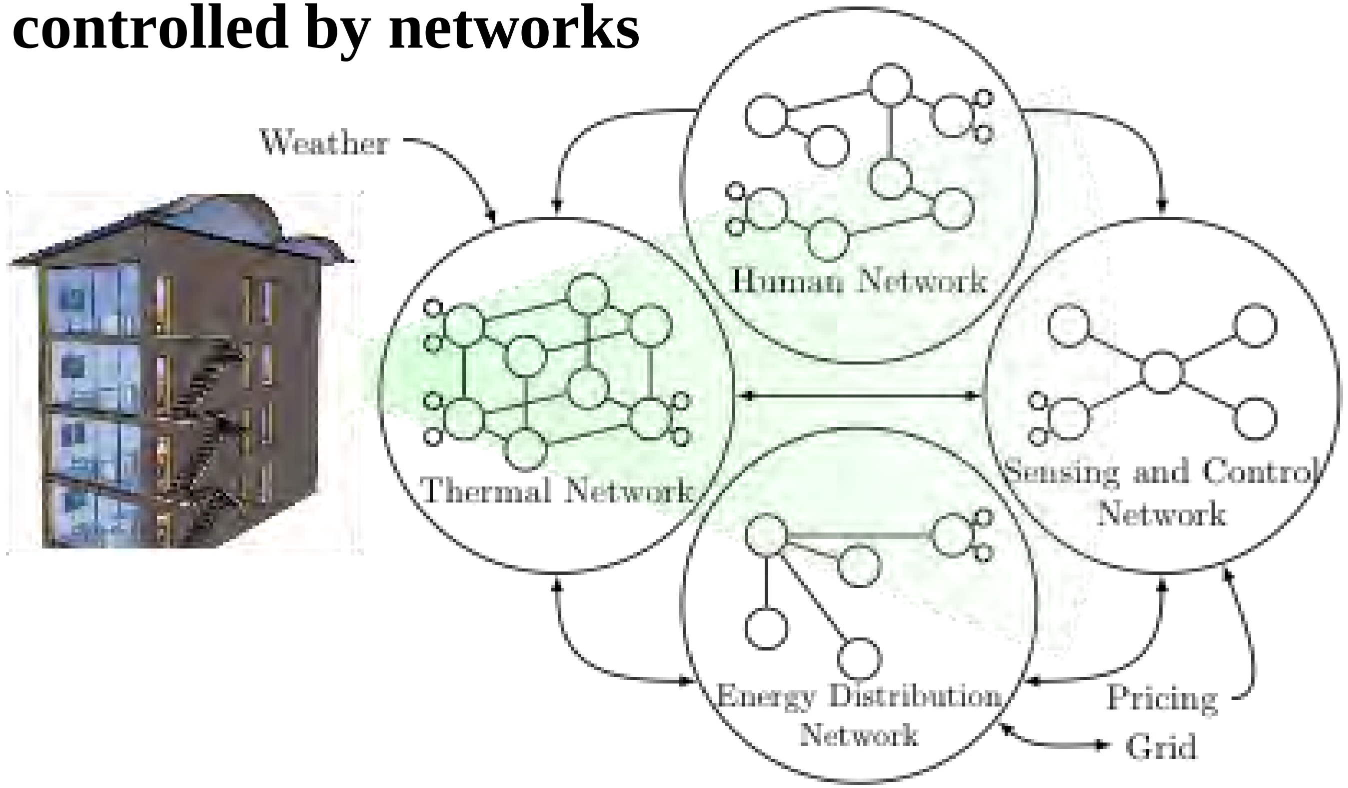
Motivation -2

- When such systems are governed by differential equations we call them a **dynamic network**
- Also called a **cyber-physical system** when there is
  - Tight integration of
    - Physical system dynamics
    - Sensors and actuators
    - Computing infrastructure
  - Multiple time and spatial scales
  - Multiple behavioral modalities
  - Context dependent interactions
- **Example**: Intelligent vehicle-highway system; cities; robot swarms



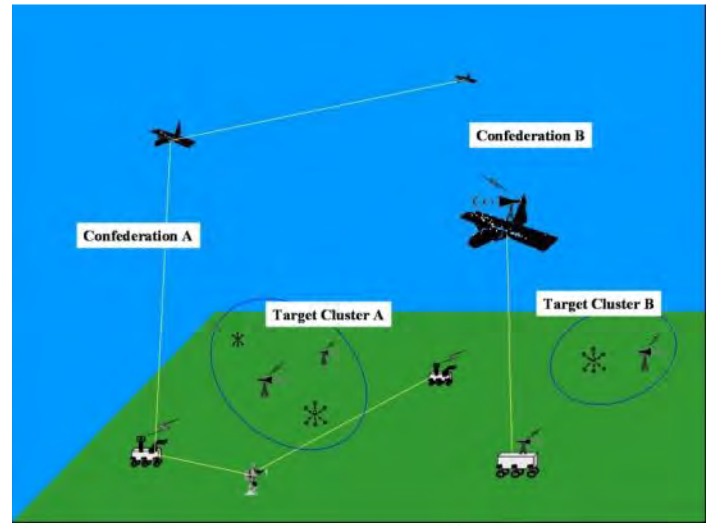
Motivation -3

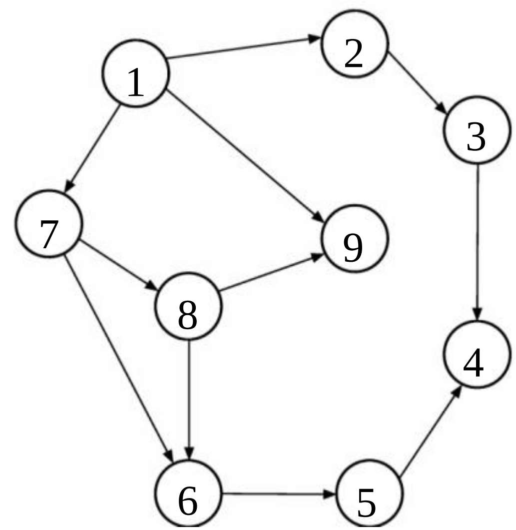
- Physical processes interacting with other sources of energy, material, and information further suggests the interpretation:
  - **Complex cyber-physical systems are networks controlled by networks**

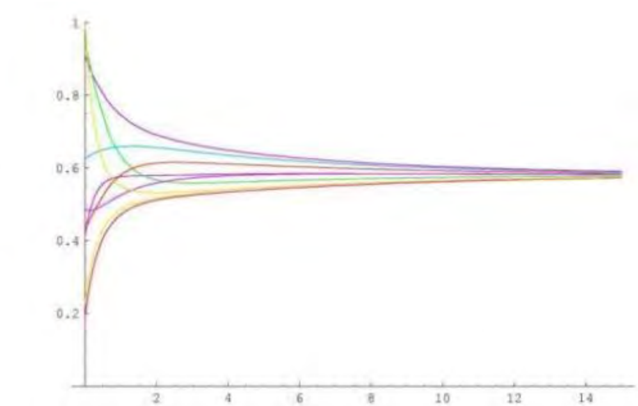


Consensus: an Algorithmic Approach to Coordination and Control in Networks

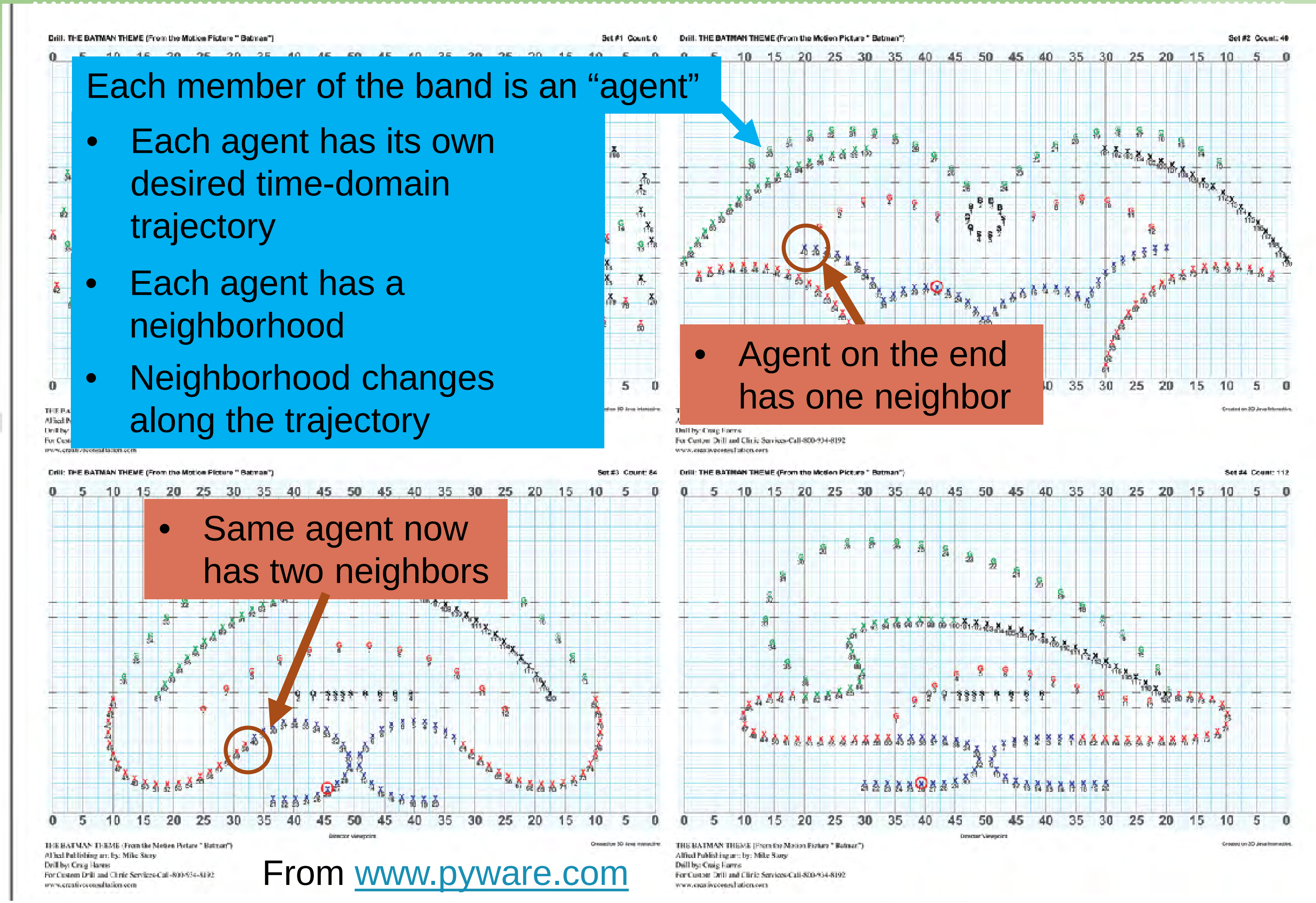
- The **consensus variable paradigm** is a generalization of potential field approaches and has connections to problems in:
  - Coupled-oscillator synchronization
  - Neural networks
- Also called agreement protocol
- Related to gossip algorithms
- Articulated in context of team theory in 1960s







Marching Band Example



Each member of the band is an "agent"

- Each agent has its own desired time-domain trajectory
- Each agent has a neighborhood
- Neighborhood changes along the trajectory

- Agent on the end has one neighbor

- Same agent now has two neighbors

From www.pyware.com

Marching Band "Agent" Control

How to Practice Marching Band Formations[1], notes

1. "Keep your eyes off the ground. It isn't going anywhere! If you rely on the ground to know where you are supposed to go, you will be learning nothing and will regret it later on, such as during performances ... Rely on the distance you are away from others to determine if you are in the correct position…"

   (i.e., nearest neighbor, relative distance control)

2. "… Repetition is key. Try practicing your formations ..."

   (i.e., iterative learning control)

[1]http://www.wikihow.com/Practice-Marching-Band-Formations