Multiple Mobile Robot Systems

Dr. Lynne E. Parker
Professor and Associate Head
Fellow, IEEE

Dept. of Electrical Engineering & Computer Science
University of Tennessee, Knoxville USA
(leparker@utk.edu)

August 17, 2010

Outline

- Introduction to Distributed Intelligence Laboratory (DILab)
- Practicalities of Physical Robot Research
- Overview of some of the projects in the DILab
- Brief Overview of Distributed Robotic Research
  - Applications
  - Architectures for Multi-Robot Systems
  - Communication
  - Swarm Robots
  - Heterogeneity
  - Task Allocation
  - Learning

Distributed Intelligence Lab, Univ. of Tenn.
Dept. of Electrical Engr. & Computer Science

- **Mission:** Create autonomous software solutions for distributed intelligent systems, including teams of multiple agents and robots, sensor networks, and embedded systems.
- **Personnel:**
  - *Director:* Prof. Lynne E. Parker
  - *Ph.D. Students:* Richard Edwards, Mike Franklin, John Hoare, Sudarshan Srinivasan, Chris Reardon, Hao Zhang, Tony Zhang
  - *M.S. Students:* Bob Lowe, Nick Overfield
- **Sponsors:**
  - NSF, Lockheed Martin, ORNL, Georgia Tech, SAIC, DOE, JPL, DARPA, Intel

Research Robots and Sensors

DILab URL: http://www.cs.utk.edu/dilab

Some Practicalities of Physical Robot Research

- Robots have physical size – they can't move through each other, can't share the same location, and must avoid collisions
- Robots rarely operate in wide-open spaces; they must act around obstacles (e.g., trees, walls), other agents that are not part of the team.
- Sensors are affected by noise, occlusion and poor/variable operating conditions (e.g., low light, shadows, hot temperatures, etc.)
- Perfect localization is rare; uncertainty in position fluctuates over time

Some Practicalities of Physical Robot Research (con't.)

- Problems caused by symmetry/deadlock sometimes resolve themselves due to noise
- Robot motions are constrained by physics, and robot design (e.g., limited turning radius, limited acceleration, inefficiencies when stopping after each unit move, etc.); they often cannot move to arbitrary points
- Robots are faulty – sensors/actuators break
- The outcome of deterministic algorithms on robots is non-deterministic (e.g., due to sensing/motion/comms uncertainty)

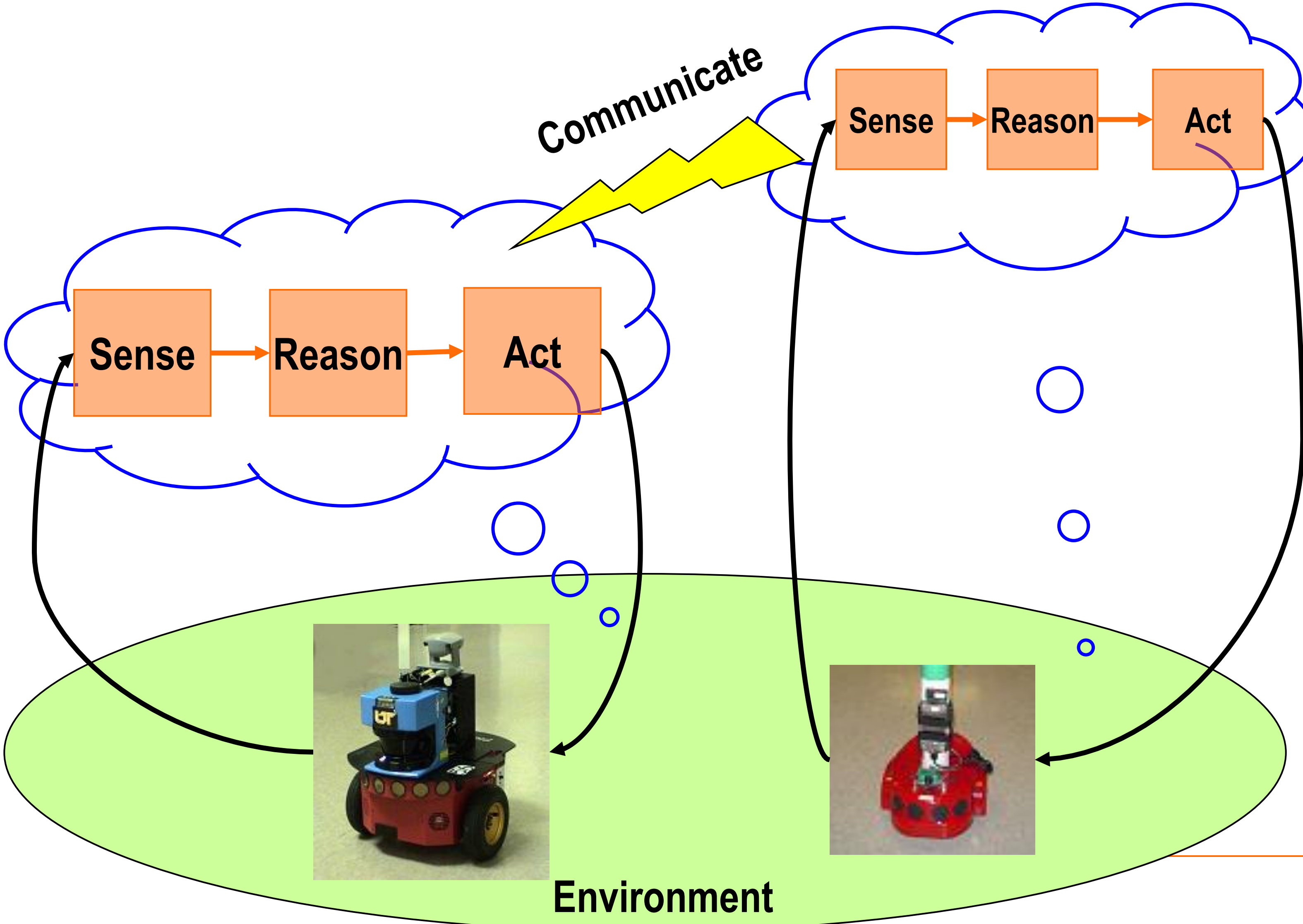
Some Practicalities of Physical Robot Research (con't.)

- Uncertainties (e.g, in position) are typically not uniform, but grow over time
- Often, performance (e.g., time, energy) is the metric of most significance, not just convergence
- Defining unique IDs for each robot is fairly easy
- Communication of a few bytes of information is fairy easy (via wireless or infrared (IR))

Some Practicalities of Physical Robot Research (con't.)

- For many capabilities, SW is not a good substitute for HW. (i.e., in general, a little extra hardware can often make the software much simpler)

How Do You Design Smart Robot Teams?



How Does a Robot "Sense"? Can use vision…

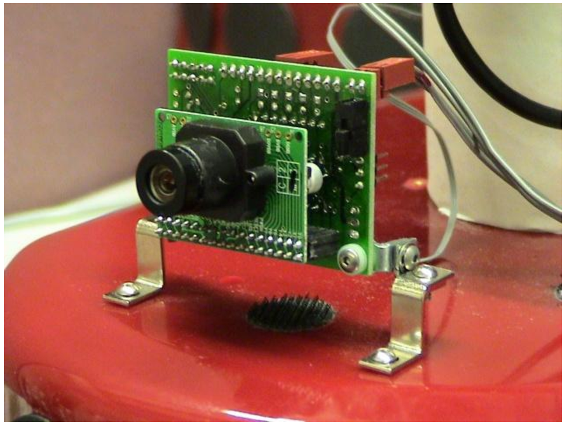
Camera

Raw data

"Perceived" info. (using intelligent software algorithms on robot):

Marker 1:	Marker 2:	Marker 3:
ID = 1010110	ID = 0101010	ID = 0011000
d = 23"	d = 42"	d = 26"
$\theta = 117^o$	$\theta = 86^o$	$\theta = 61^o$
$\alpha = 76^o$	$\alpha = 0^o$	$\alpha = 32^o$

Example of Robots Using Cameras to Cooperatively Follow the Leader



Simple camera (CMUCam) for color blob tracking

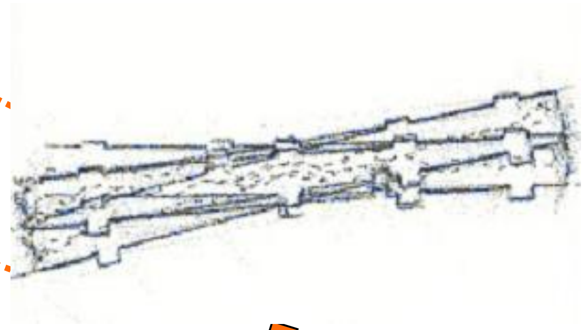
Sample image from camera

Parker's DI Lab, UTK, 2004

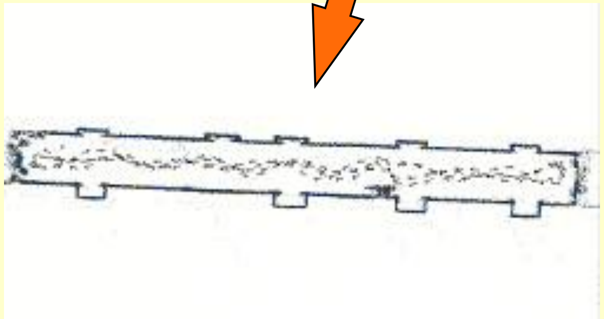
How Does a Robot "Sense"? Can use laser range scanner…

Laser range scanner

Raw data (multiple scans)



"Perceived" info. (using intelligent software algorithms on robot):



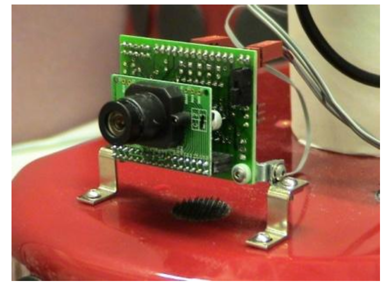
Example of Robots Using Laser Range Scanners to Help Push Box

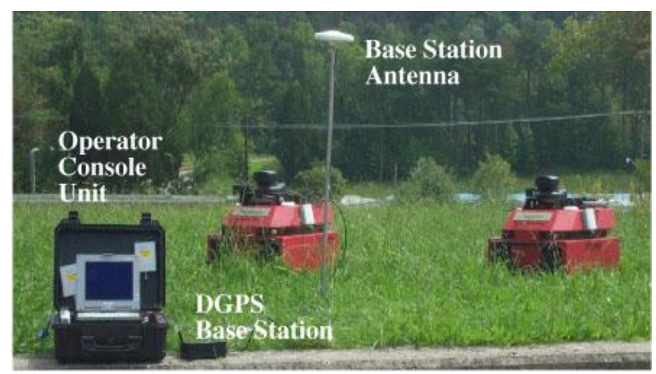
Lm-hri/visulalization--slowed

Hoare, UTK, 2009

And, there are lots of other sensors…

- "Low-end" camera
- Infrared
- Sonar
- Microphone
- DGPS
- Compass
- Odometry (wheel encoders)
- Inclinometer (tilt)
- Tactile
- Chemical, radiation, wind, …





But it's not so easy for a robot to sense…

What if camera breaks,

or the grass is too high,

or the lens gets dirty,

or the sensor gives bad data,

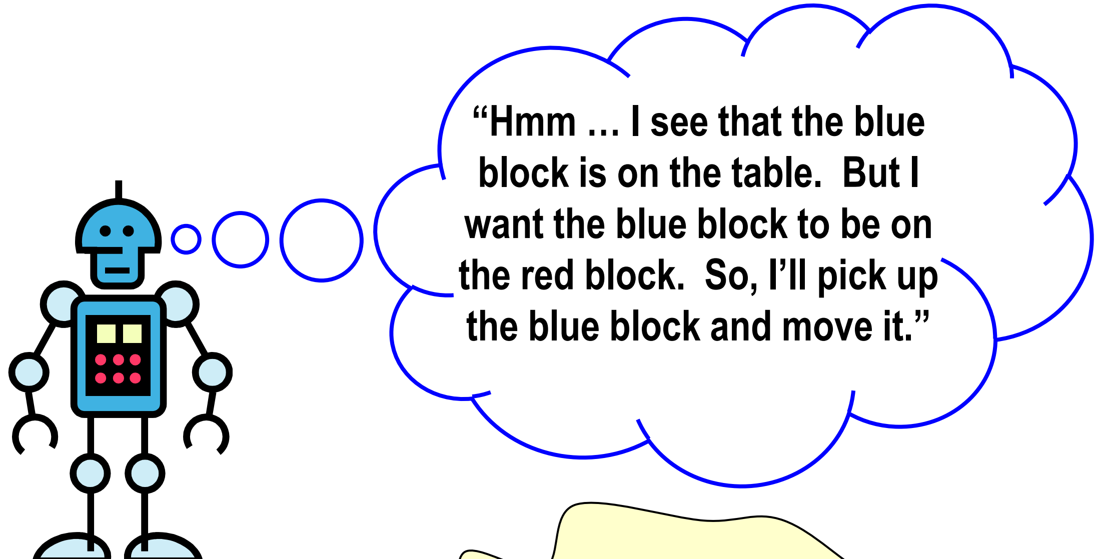
or 2 sensors tell you contradictory things,

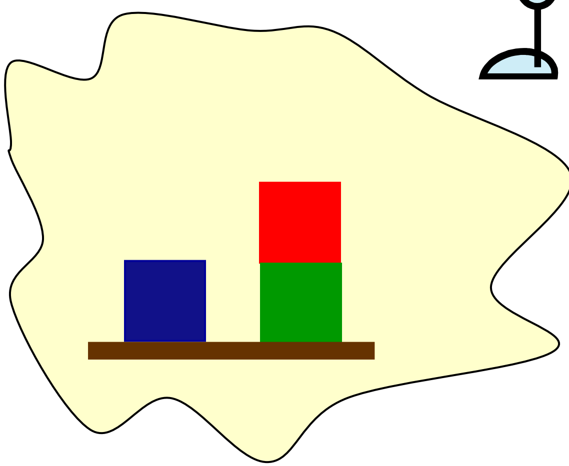
or the lights get turned off,

or it starts raining,

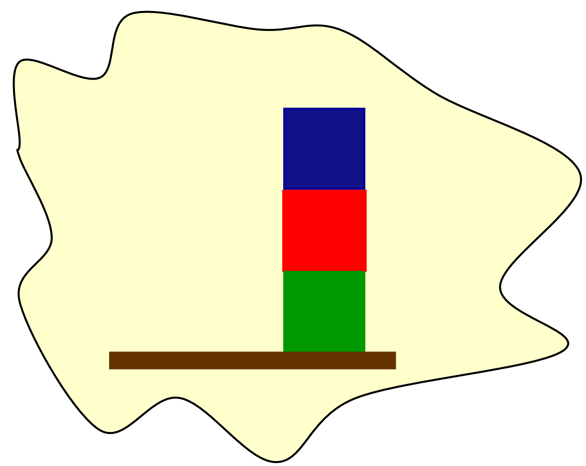
or, … ?

How Does a Robot "Reason"?





Current Situation



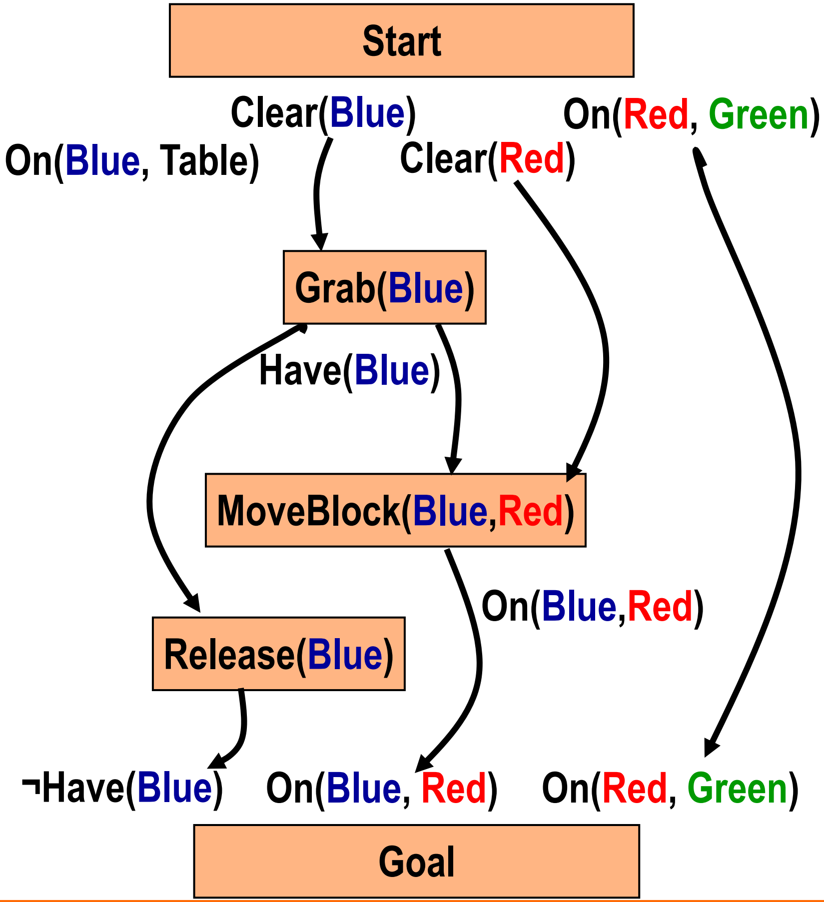
Desired Situation

This Reasoning is called “Autonomous Planning”

Uses Logical Reasoning (“First Order Predicate Calculus”)

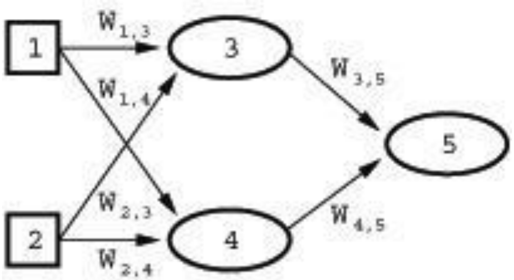
Rules expressed in Logic:

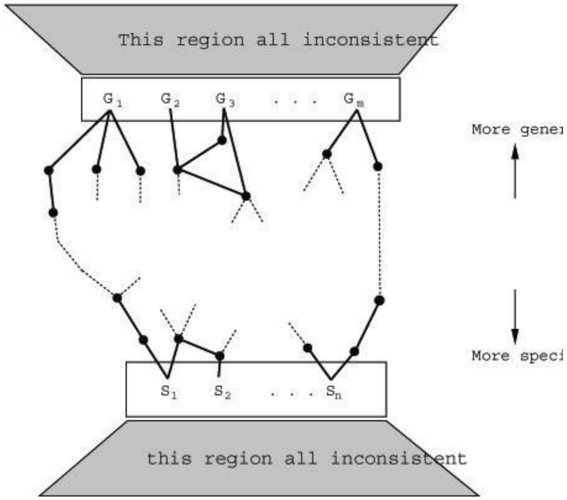
- `Action:   MoveBlock(x,y)`
  - `Preconditions:`
    `Clear(y) ∧ Have(x)`
  - `Effects:  On(x,y)`
- `Action:   Grab(x)`
  - `Preconditions:`
    `Clear(x)`
  - `Effects:  Have(x)`
- `Action:  Release(x)`
  - `Preconditions:`
    `Have(x)`
  - `Effects:  ¬Have(x)`



And, there are lots of other ways to "reason"…

- Constraint propagation
- Production ("expert") systems
- Decision networks
- Probabilistic reasoning
- Dynamic Bayesian networks
- Hidden Markov Models
- Genetic algorithms
- Neural networks
- …





But it's not so easy for a robot to reason…

What if blue block is glued to table,

or you accidentally drop your block,

or somebody keeps knocking down your blocks,

or the pathway is blocked,

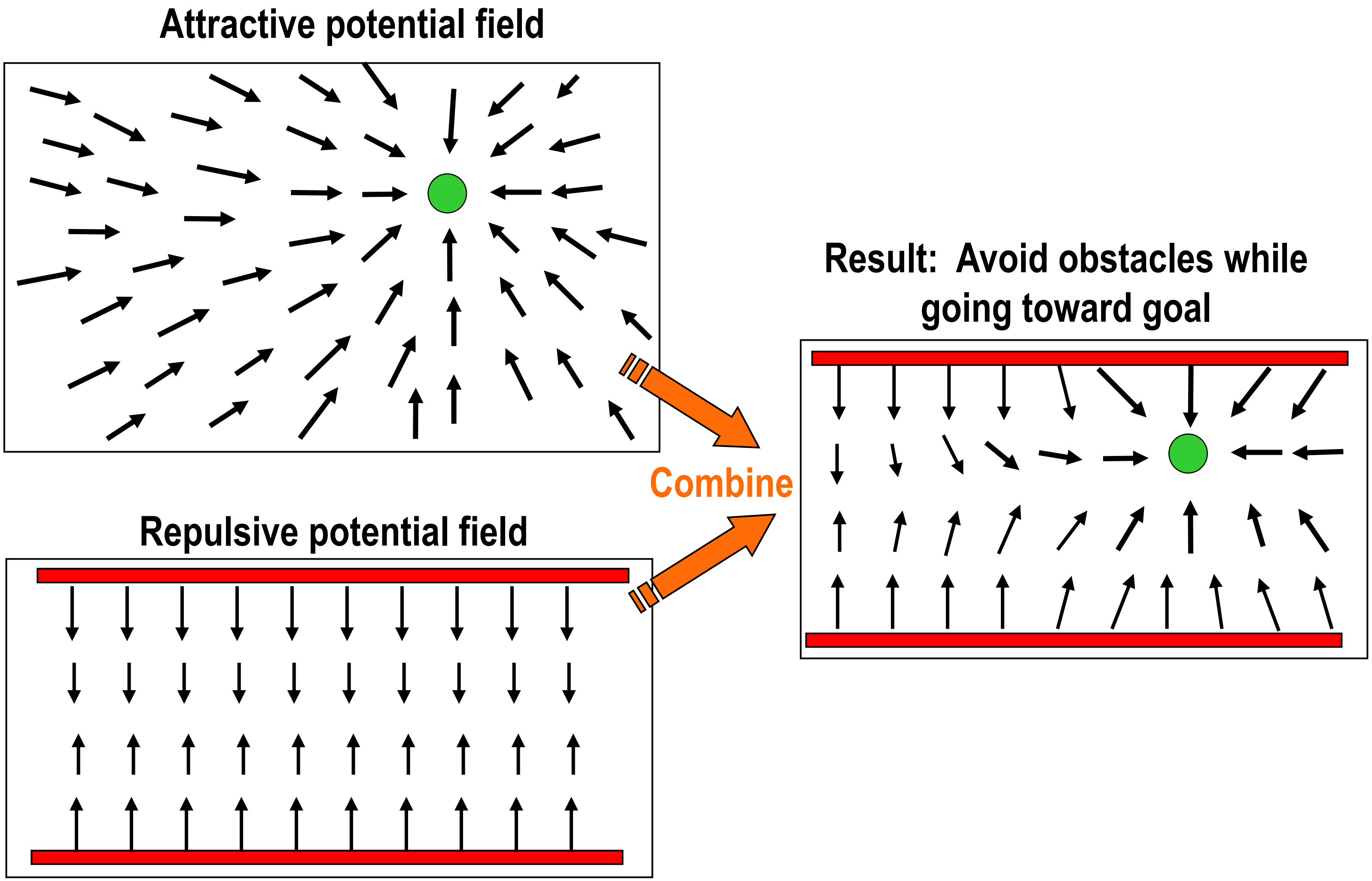
or you don't know what to do,

or by the time you've figured out what to do, the world has changed,

or the building catches on fire,

or, … ?

How does a robot "act"? Can use potential fields…



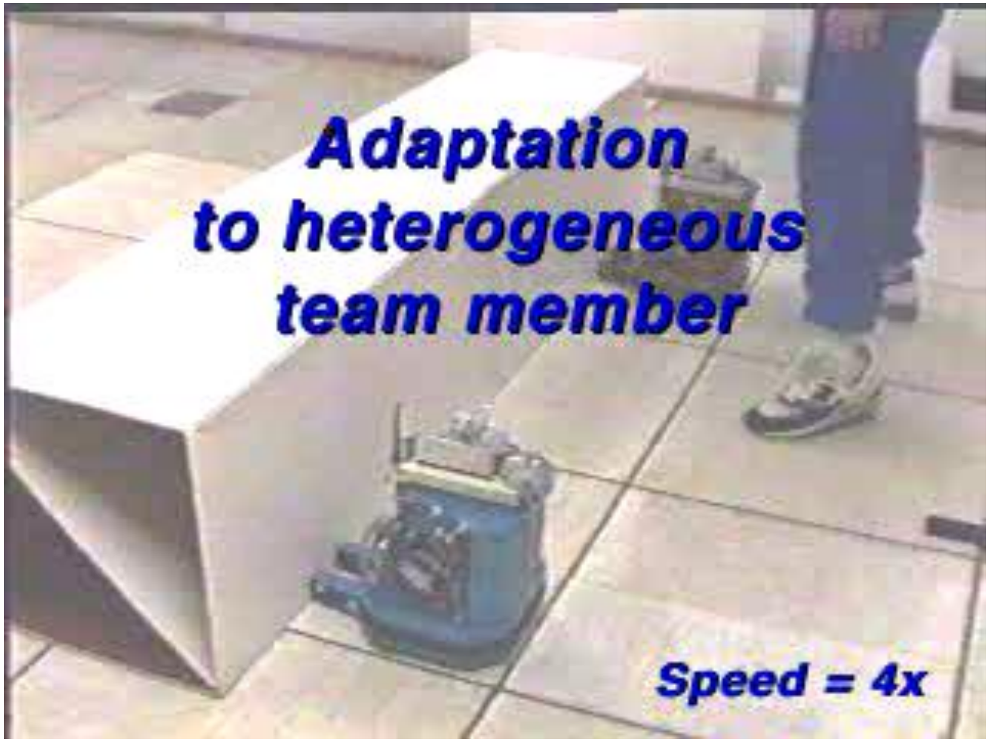
How Does a Robot Convert "Potential Fields" to Motion? Can use wheels…

- "Go forward at 1 meter per second"
  - `SetMotorState(1)`
  - `SetSpeed(1000,0)`
- "Steer hard to the right"
  - `SetMotorState(1)`
  - `SetSpeed(50,90)`
- "Stop!"
  - `SetSpeed(0,0)`
  - `SetMotorState(0)`

And, there are lots of other ways to move…

- Legs
- Tracks
- Arms
- Wings…

Example of Robots Pushing Boxes



L. Parker, MIT, 1994

But it's not so easy for a robot to act…

What if your wheels get stuck,

or your gripper breaks,

or you collide with something,

or your battery gets too low,

or you try to grip something, but it doesn't work,

or you fall into a hole,

or a car is coming,

or, …?

How do Robots Cooperate?
Can communicate local info to each other…

- Share and compare local sensor data:
  - Acoustics
  - Chemical concentrations
  - Visual tracks, …

Acoustic sensor network:

Parker's DI Lab, UTK, 2004

Example of Sensor Network Cooperating with Mobile Robot

Y. Li, UTK, 2007

But, it's not so easy for robots to cooperate…

What if some of the robots fail,

or the wireless communication goes down,

or robots can't find each other,

or one robot un-does what another robot just did,

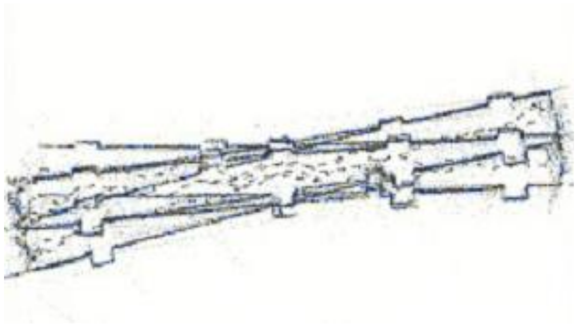
or one robots thinks push while the other thinks pull,

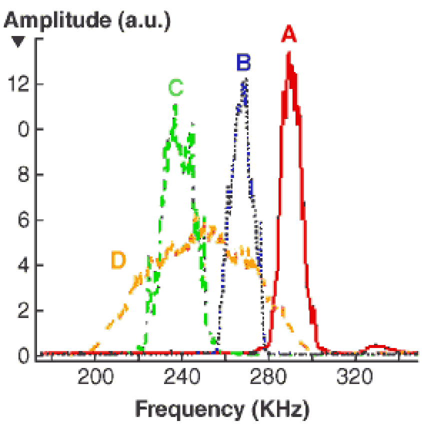
or one robot refuses to help another,

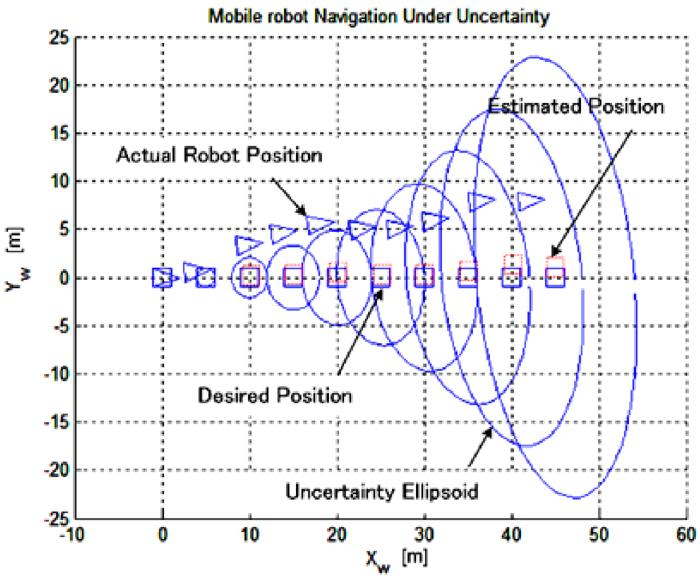
or, … ?

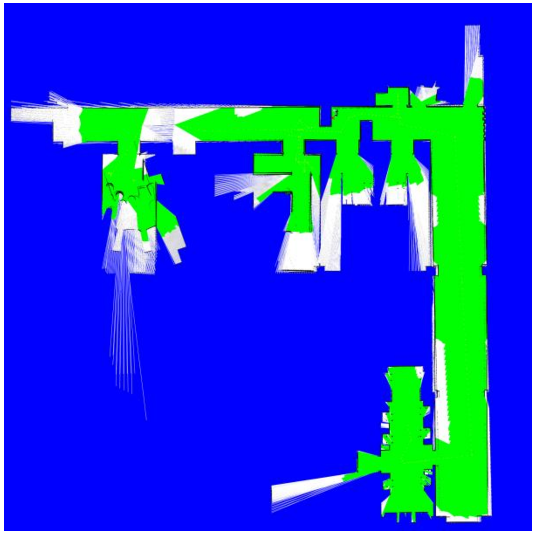
Primary Research Challenge is Dealing with Uncertainty

- **Uncertainties abound in:**
  - Sensing
  - Reasoning
  - Acting
  - Communicating/Cooperating



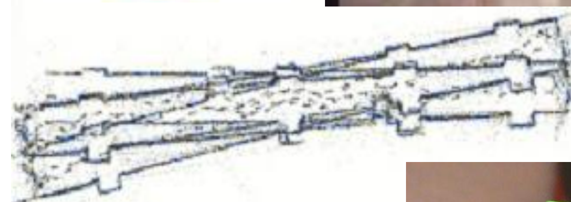


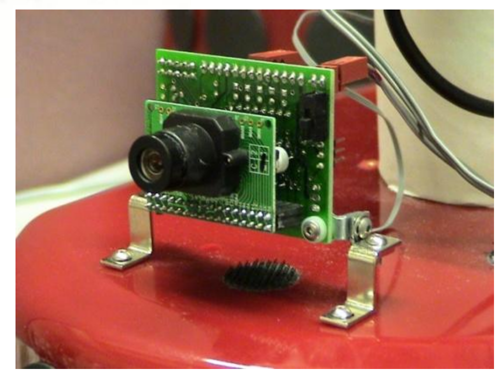




Sensing Uncertainties Abound!

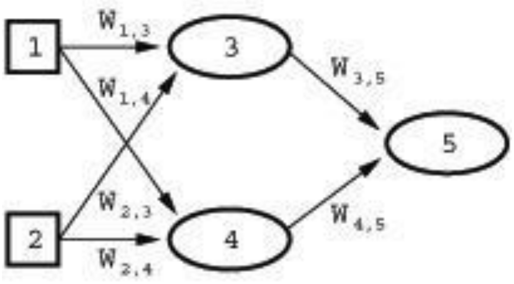
- Sensor failures
- Noisy data
- Conflicting data from multiple sensors
- Specular reflection
- Poor operating conditions
- Lack of calibration
- (etc.)

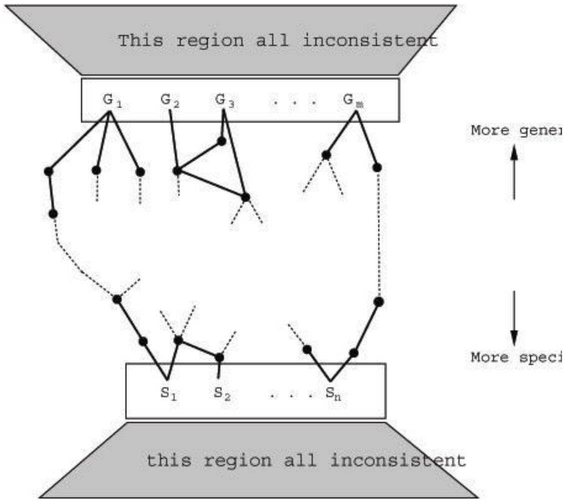




Reasoning Uncertainties Abound!

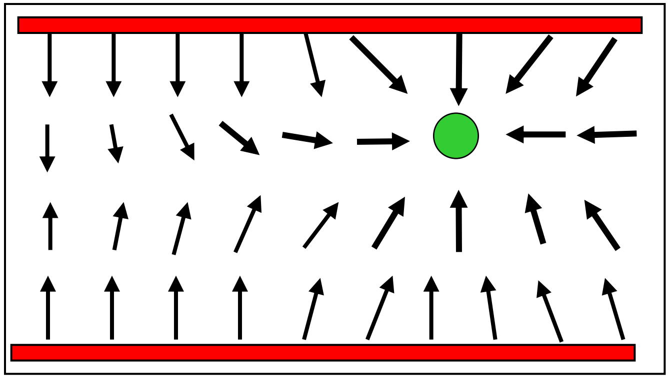
- Incomplete (often only local) information
- "Non-Markovian" environments
- Incomplete models of the world
- Dynamic environments
- Unexpected events
- NP-hard problems require approximate solutions
- Lack of common sense reasoning
- (etc.)





Acting Uncertainties Abound!

- Wheels/legs/etc. do not execute perfectly
- Slipping, sliding, friction
- Collisions
- Battery levels
- Mechanisms degrade or fail
- Robot localization difficult
- Poor repeatability
- (etc.)



Communication/Cooperation Uncertainties Abound!

- Noisy wireless communications
  - Lost messages
  - Delayed messages
  - Signal interference
- Unknown state of other robots
- Robots may not recognize each other
- Inter-robot interference/collisions
- Competing priorities
- Heterogeneous robots
- (etc.)

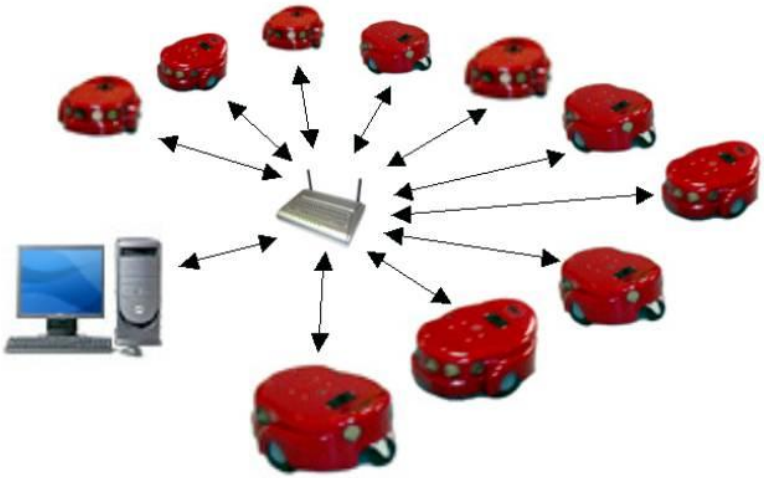
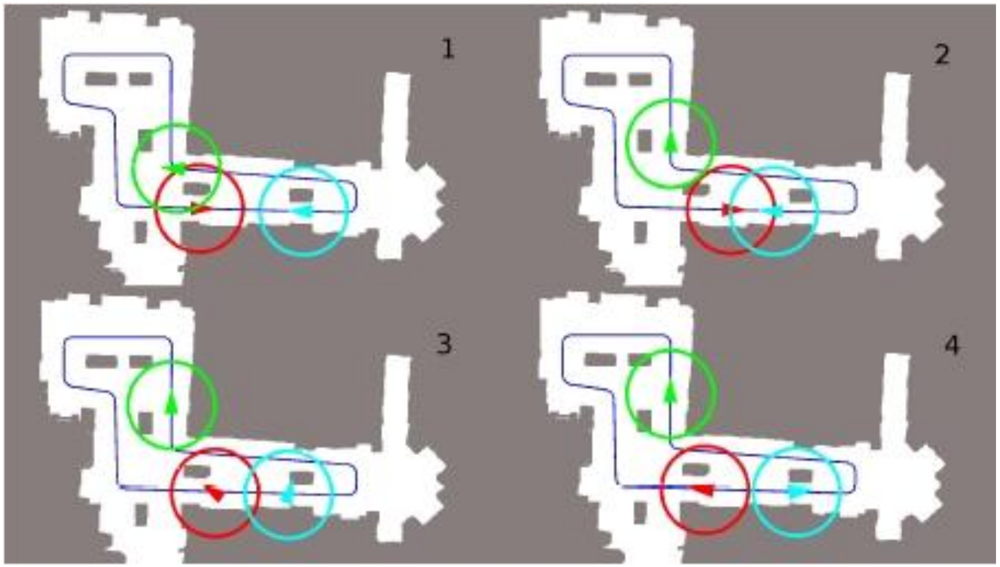


Illustration of Uncertainty

Parker, et al., 2004

UT Collaborative Project with Univ. degli Studi della Basilicata (Italy): Multi-Robot Perimeter Patrol



Impact:

- Allows robots to patrol border in fully decentralized fashion, with no communication
- Can enable dynamic reaction in patrolling due to robot failures, or to allow a "friend" robot to pass through

For more information:

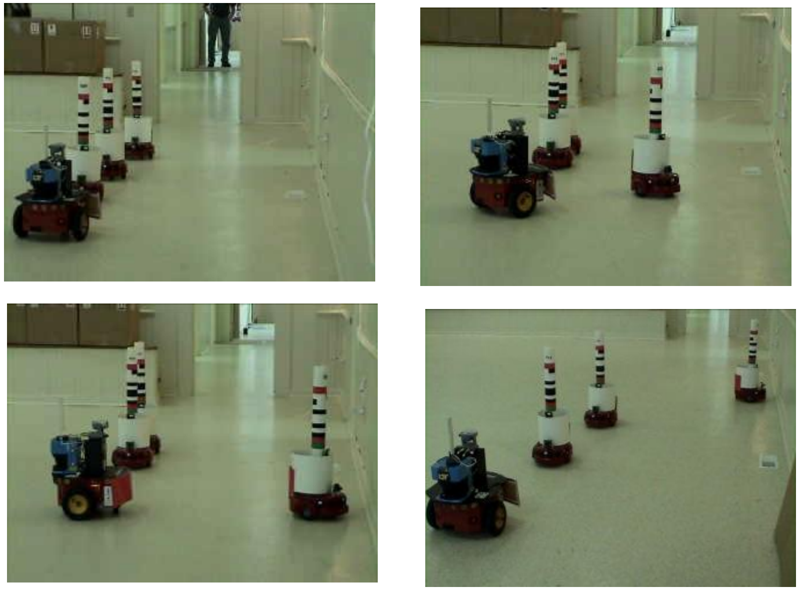
Marino, Parker, Antonelli, Caccavale, "Behavioral Control for Multi-Robot Perimeter Patrol: A Finite State Automata approach", ICRA 2008.

New Ideas:

- Defines behavioral control based on the concept of *action* obtained by combining elementary Null-Space-Behavioral Framework
- Uses a finite state automata as a supervisor in charge of selecting

Movies of Perimeter Patrol

UT Project in Heterogeneity: *ASyMTRe* – Automated Synthesis of Task Solutions for Teams



Impact:

- Enables robots to share sensory/perceptual resources across the network
- Enables robots to determine appropriate teaming behaviors when *how* to solve a task is dependent on team capabilities

New Ideas:

- Changes fundamental abstraction from *task* to *schemas* which can be recombined in multiple ways to solve the same task in different ways
- Find solutions based on *information flow* through network

For more information:

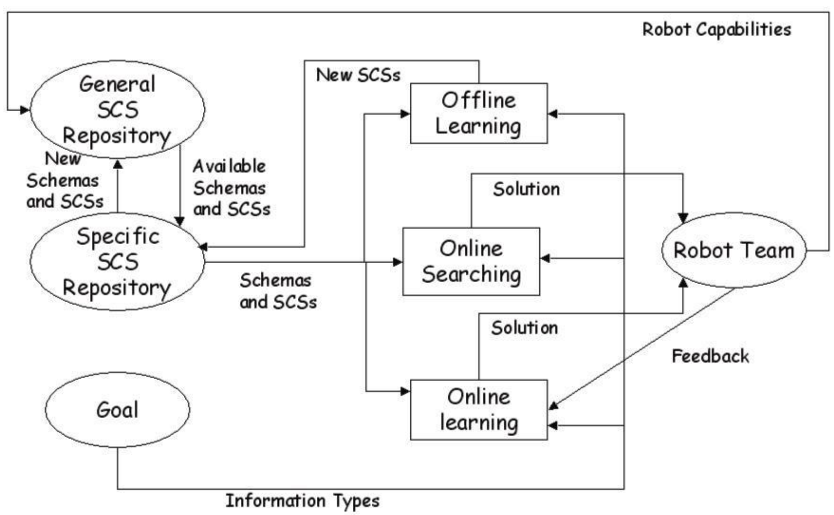
Parker and F. Tang, "Building Multi-Robot Coalitions through Automated Task Solution Synthesis", *Proceedings of the IEEE*, Special Issue on Multi-Robot Systems, 2006

F. Tang and Parker, "Layering ASyMTRe-D with Task Allocation for Multi-Robot Tasks", in ICRA 2007.

Movie of Heterogeneous Sensor Sharing

UT Project in Learning: Schema-Based Constructivist Robot Learning via "Chunking"

SB-CoRLA Architecture



Impact:

- Provides a robot learning architecture for continuous online and offline learning
- Enables schema based learning via "chunking" (building higher hierarchical schemas as a congregation of lower hierarchical schemas)

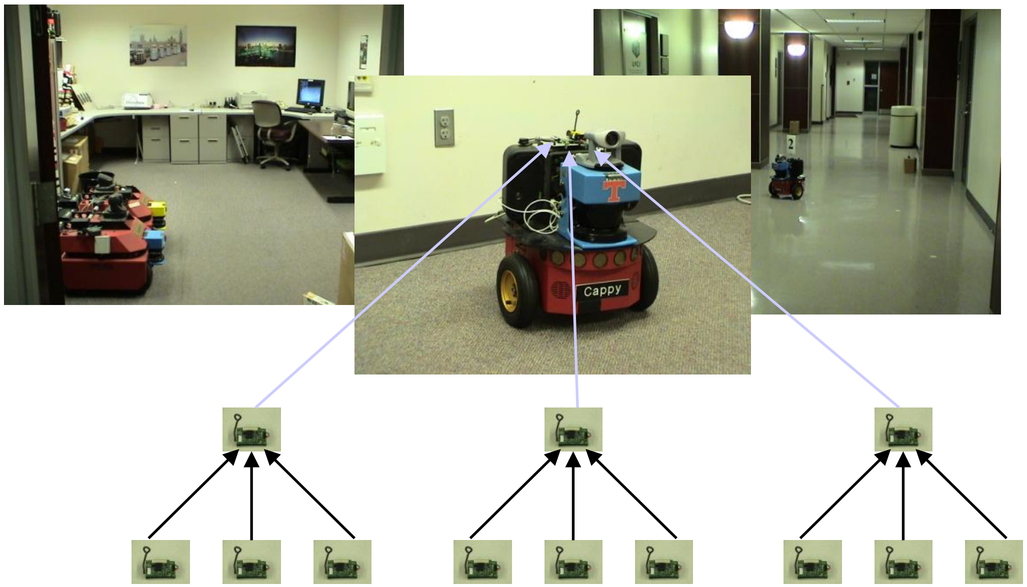
New Ideas:

- Uses evolutionary learning to generate highly fit partial solutions
- Extracts "chunks" of schemas for hierarchical learning and future online search
- Extends ASyMTRe search algorithm to employ "chunks" of schemas in the online search process

For more information:

Y. Tang and L. E. Parker, "Towards Schema-Based, Constructivist Robot Learning: Validating an Evolutionary Search Algorithm for Schema Chunking", in ICRA 2008.

UT Project in Learning: Anomaly Detection in Mobile Sensor Networks



Impact:

- Provides general approach to anomaly detection
- Approach applicable to many applications (e.g. intruder detection, environmental monitoring, etc.)

New Ideas:

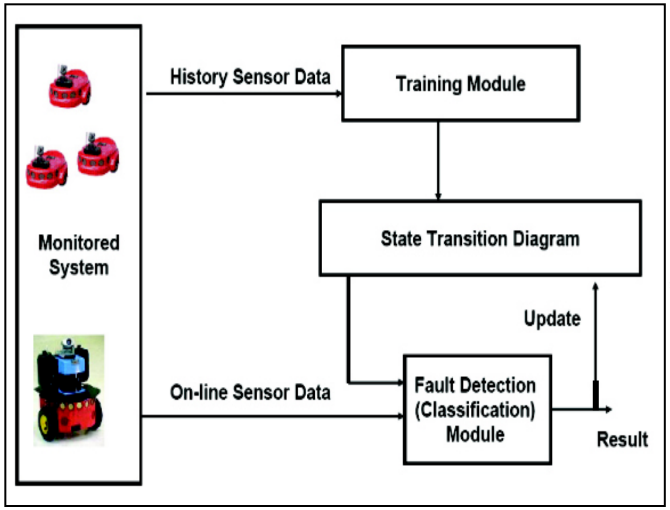
- Make use of FuzzyART (Adaptive Resonance Theory) for learning   normal" and detecting anomalies
- Create new category when sensor input is significantly different from what has been seen before
- Mobile nodes (robots) respond to anomalies in sensor network

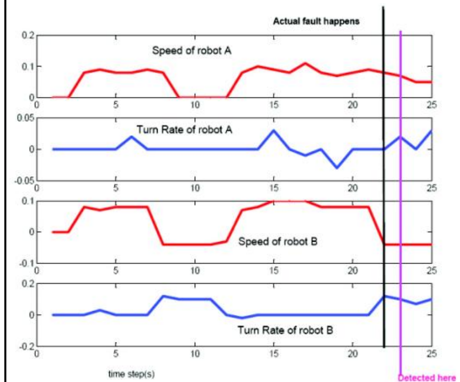
For more information:

Parker and Li, "Detecting and monitoring time-related abnormal events using a wireless sensor network and mobile robot", in *Proc. of IEEE Int'l. Conf. on Intelligent Robots and System*, 2008.

Li and Parker, "A spatial-temporal imputation technique for classification with missing data in a wireless sensor network", in *Proc. of IEEE Int'l. Conf. on Intelligent Robots and System*, 2008.

UT Project in Fault Detection – *SAFDetection*: Cooperative behavior monitoring for fault detection





Impact:

- **General approach** to cooperative fault detection that does not require extensive modeling of control
- **Increases robustness** of team through easier detection of tightly-coupled task problems

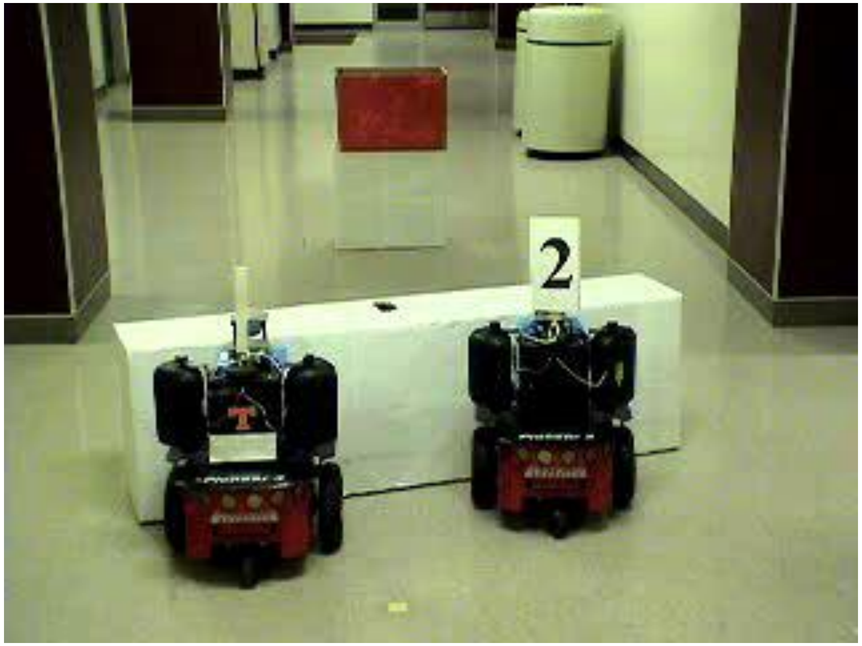
New Ideas:

- Monitor sensor data in multi-robot coalitions to **learn model of expected behavior**
- **"Black box" sensor model** makes use of feature selection, fuzzy c-means clustering, HMMs to learn "normal"
- Use **online monitoring** to detect errors in cooperation – either hard faults, logic faults, or coalition faults

For more information:

- Li and Parker, "Sensor Analysis for Fault Detection in Tightly-Coupled Multi-Robot Team Tasks", in *Proc. of IEEE Int'l. Conf. on Robotics and Automation*, 2007.
- Li and Parker, "Distributed Sensor Analysis for Fault Detection in Tightly-Coupled Multi-Robot Team Tasks", *Proc. of IEEE Int'l. Conf. on Robotics and Automation*, 2009.

Movies of SAFDetection



UT Project in
Peer-to-Peer Human-Robot Teaming

New Ideas:

- Humans and robots work on shared cooperative activity in same physical space
- Robot and human work as equal peers with robot being attentive to goals and intentions of human
- No explicit communication between human and robot

Impact:

- Should be more natural to humans
- Could enable human-robot coordination in communications-limited environments
- Could allow humans and robots to work together in a manner similar to human-only teams today

***Example of human-only peer-to-peer team*:**

Fireteam of soldiers clearing a building; through training, they understand well how to interact and work as a team, with minimal need for explicit communication

Our objective is to add robots to this type of team, while maintaining the same "natural" interaction capabilities within the team

Movie of HRI

- **P2P-Robot-Right-2x.mpeg**

Some Unifying Themes

- Make use of sensor-based machine learning to learn relevant patterns:
  - E.g., patterns of “normal” behavior; or patterns of “useful” behavior
  - Examples of techniques employed:
    - Fuzzy Adaptive Resonance Theory Neural Networks
    - Hidden Markov Models, extended with time duration in state
    - Conditional Random Fields
    - Case-based reasoning
    - Fuzzy C-Means clustering
    - Principal components analysis
- Use learned models:
  - To detect “off-normal” events, for fault detection
  - To recognize current state of cooperation, for proper action selection
  - To extend robot capabilities
    - Via constructivist learning
    - Via genetic algorithms
    - Via cross-application transfer of learning

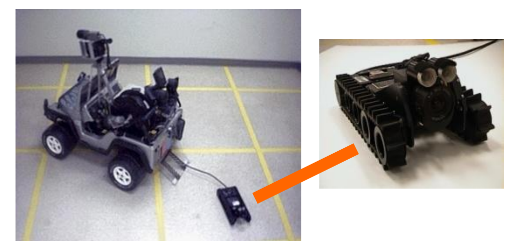
Overview of Multi-Robot Systems

- Applications
- Communication
- Swarm Robots
- Heterogeneity
- Task Allocation
- Learning
- Architectures for Multi-Robot Systems

Categorizing Multi-Robot Systems

- Cooperative robotics field is often divided according to a number of criteria:
  - Collective (swarm) cooperation
    - Many robots; sub-symbolic communication (possibly implicit)
    - Typically uses insect society cooperation model

  - Homogeneous vs. Heterogeneous systems
    - Sensors, actuators and behavior
    - Affects communication possibilities

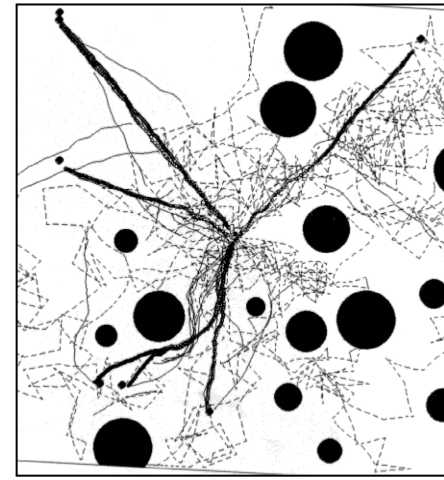


  - Centralized vs. Distributed
    - Centralized systems typically use classical-AI planning, rather than being “behavior-based” (new AI)

Commonly Studied Tasks for Multi-Robot Teams

- **Foraging and Coverage:** collection of randomly placed items
- **Formations and flocking:** team maintains a geometric pattern while moving
- **Box pushing / cooperative manipulation:** team collectively moves object
- **Multi-target observation:** team maintains targets within field of view
- **Traffic control / multi-robot path planning:** coordinating actions in shared space
- **Multi-robot soccer:** game that incorporates challenging aspects of cooperation

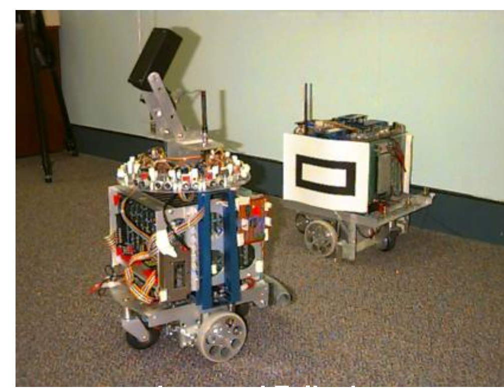
Multi-Robot Communication



Objective of communication: Enable robots to obtain "enough" information about teammates' states/actions to allow team to achieve globally coherent solution.

Three (3) most common techniques:

- Implicit communication *through the world* (stigmergy)
- Passive action recognition, which uses sensors to observe actions/state of teammates
- Explicit communication, involving active, intentional transmission of signals (e.g., wireless comms or flashing lights)



Swarm-Type Motion Coordination

- Lots of types of motion coordination:
  - Relative to other robots:
    - E.g., formations, flocking, aggregation, dispersion…
  - Relative to the environment:
    - E.g., search, foraging, coverage, exploration …
  - Relative to external agents:
    - E.g., pursuit, predator-prey, target tracking …
  - Relative to other robots and the environment:
    - E.g., containment, perimeter search …
  - Relative to other robots, external agents, and the environment:
    - E.g., evasion, soccer …

Some recent "swarm" robotics (2004)

- James McLurkin, MIT and iRobot
- Developed libraries of "swarm" behaviors, such as:
  - avoidManyRobots
  - disperseFromSource
  - disperseFromLeaves
  - disperseUniformly
  - computeAverageBearing
  - followTheLeader
  - navigateGradient
  - clusterIntoGroups
  - …

- For more information: "Stupid Robot Tricks: A Behavior-Based Distributed Algorithm Library for Programming Swarms of Robots, James McLurkin, Master's thesis, M.I.T., 2004.

http://people.csail.mit.edu/jamesm/McLurkin-SM-MIT-2004(72dpi).pdf

McLurkin's Robot Swarms

- Approach to generating behaviors is similar to early work of Reynolds (Boids) and Mataric (Nerd Herd), in principle
- Primary differences:
  - Algorithms more tuned to the SwarmBot
  - More exhaustively tested
  - Parameters explored,
  - More kinds of behaviors,
  - etc.

Movies of SwarmBots

Movies of SwarmBots (con'.t)

Movies of SwarmBots (con'.t)

Movies of SwarmBots (con'.t)

Additional Topics to be Discussed in Next Talk

- Heterogeneity
- Task Allocation
- Architectures for Multi-Robot Systems