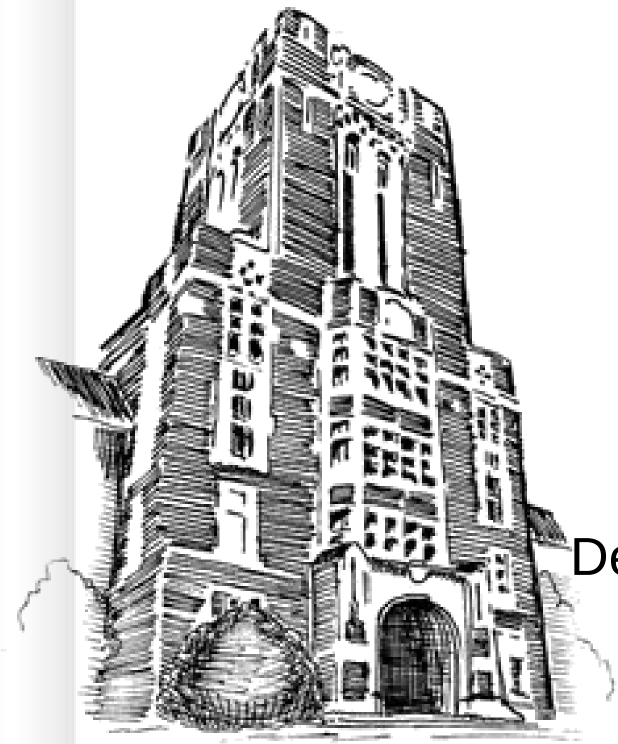


# Multi-Robot Path Planning and Motion Coordination

Dr. Lynne E. Parker
Professor and Associate Head

Dept. of Electrical Engineering & Computer Science
University of Tennessee, Knoxville USA

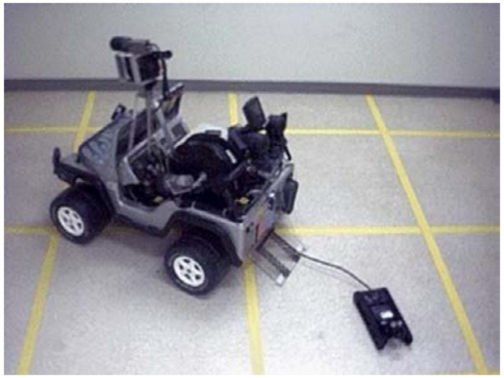
leparker@utk.edu

http://web.eecs.utk.edu/~parker

# Multi-Robot Motion Coordination

- Objective: enable robots to navigate collaboratively to achieve spatial positioning goals
- Issues studied:
  - Multi-robot path planning
  - Traffic control
  - Formation generation
  - Formation keeping
  - Target tracking
  - Target search
  - Multi-robot docking

Kumar (UPenn), Formations

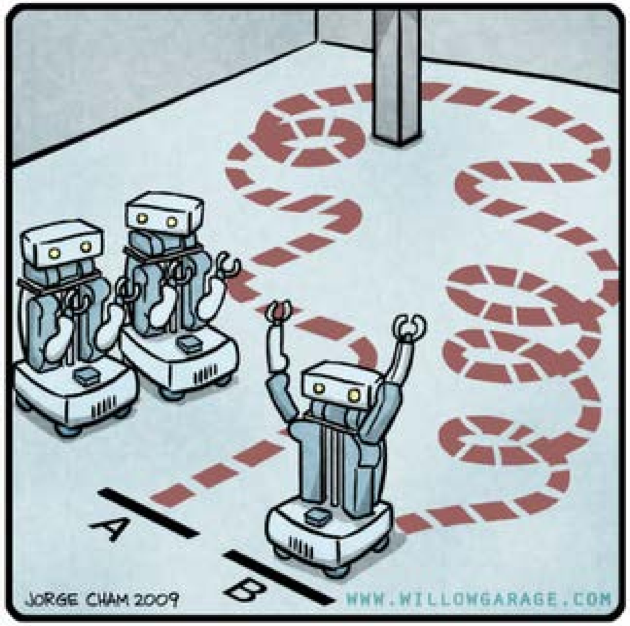


Murphy (USF), Docking

# Multi-Robot Path Planning – Problem Definition

- Given: *m* robots in *k*-dimensional workspace, each with starting and goal poses
- Determine path each robot should take reach its goal, while avoiding collisions other robots and obstacles
- Typical optimization criteria:
  - Minimized total path lengths
  - Minimized time to reach goals
  - Minimized energy to reach goals
- Unfortunately, this problem is PSPACE
  - Instead, opt for locally optimal portions of path planning problem

R.O.B.O.T. Comics



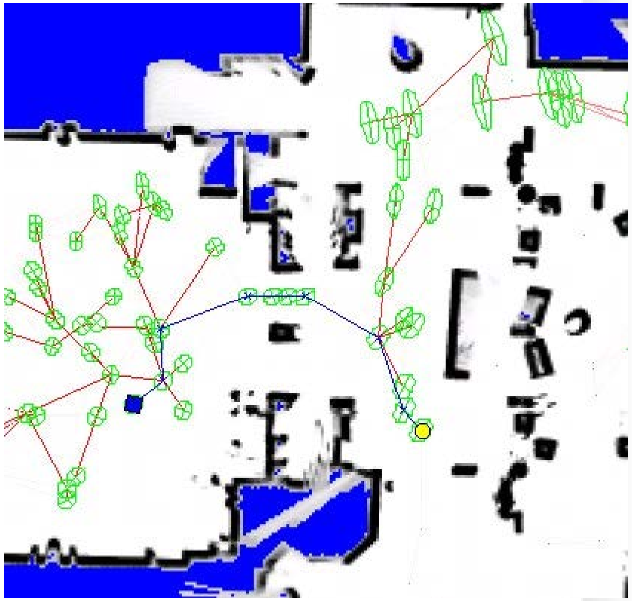

"HIS PATH-PLANNING MAY BE SUB-OPTIMAL, BUT IT'S GOT FLAIR."

# Taxonomy of Path Planning Techniques

1) Coupled, centralized approaches:
   - Plan directly in the combined configuration space of the entire robot team
   - Requires computational time exponential in the dimension of the configuration space
   - Thus, only applicable for small problems
2) Decoupled approaches:
   - Can be centralized or distributed
   - Divide problem into parts
     - E.g., plan each robot path separately, then coordinate
     - Or, separate path planning and velocity planning

# Coupled, Centralized Approaches

- Consider team a composite robot system
- Apply classical single-robot path planning algorithms, e.g.:
  - Sample-based planning
  - Potential-field techniques
  - Combinatorial methods



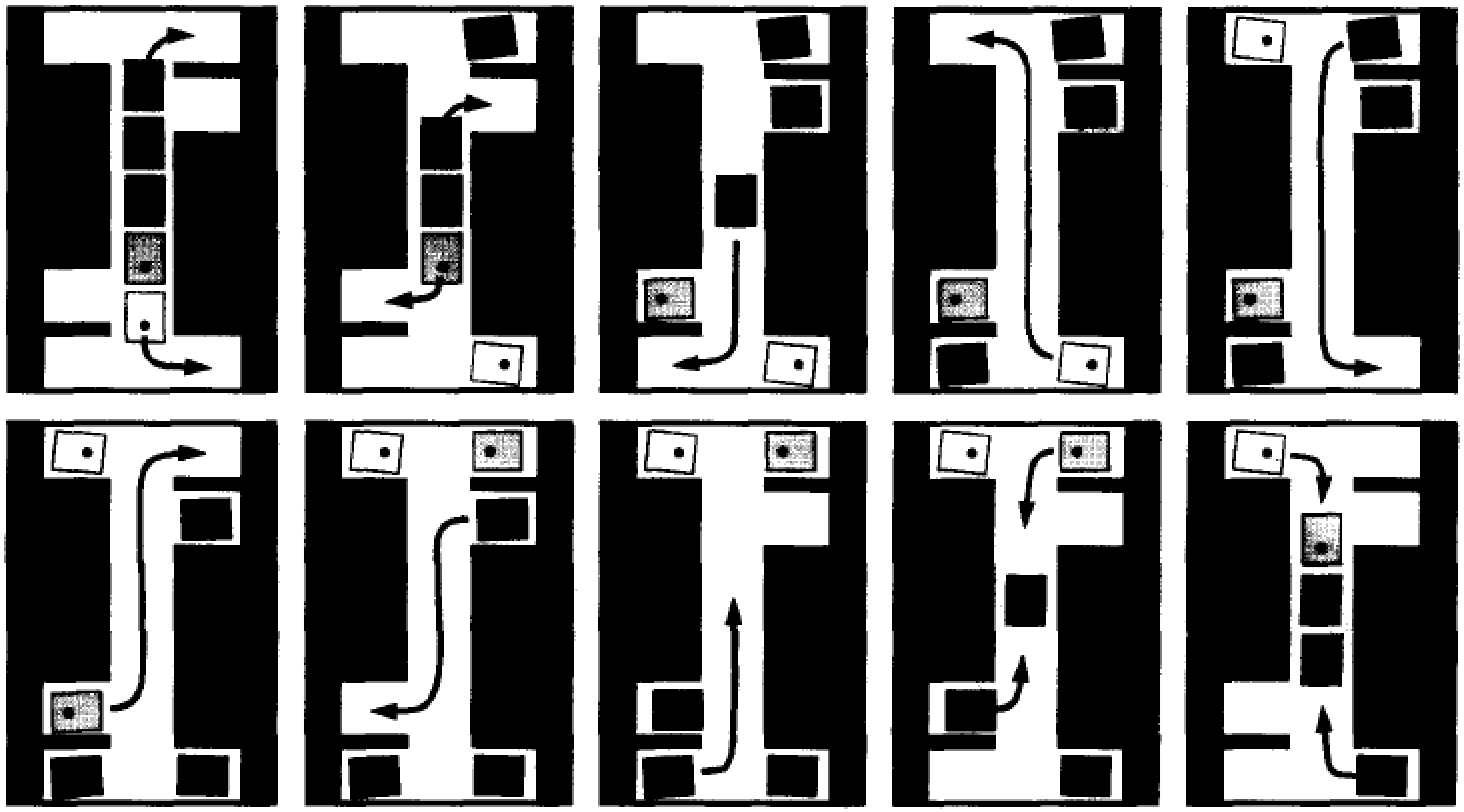
(from Prentice and Roy, MIT)

- Single-robot path planning:
  - In stationary environments: techniques such as graph searching are guaranteed to return optimal paths in polynomial time
  - In dynamic environments: Problem is PSPACE-hard, and not solvable in polynomial time

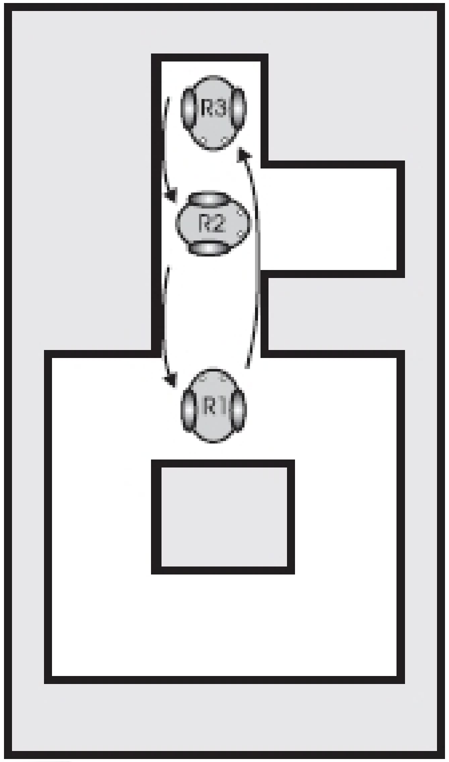
# Extending Problem to Multiple Robots

- Techniques become exponential in the number of robots
- Thus, centralized techniques are impractical except for small problems
- Better: reduce size of search space
  - Common technique: limit motion of robots to lie on *roadmaps* in the environment

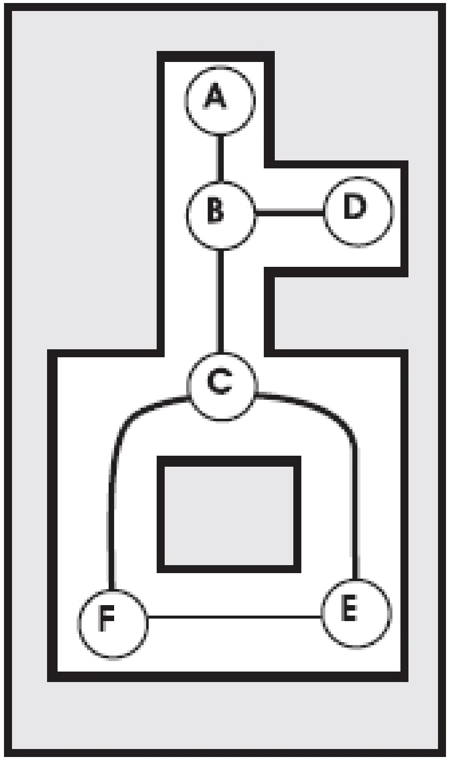
# Example Roadmap Method #1: Super-graph Method (Svestka and Overmars, 1998)



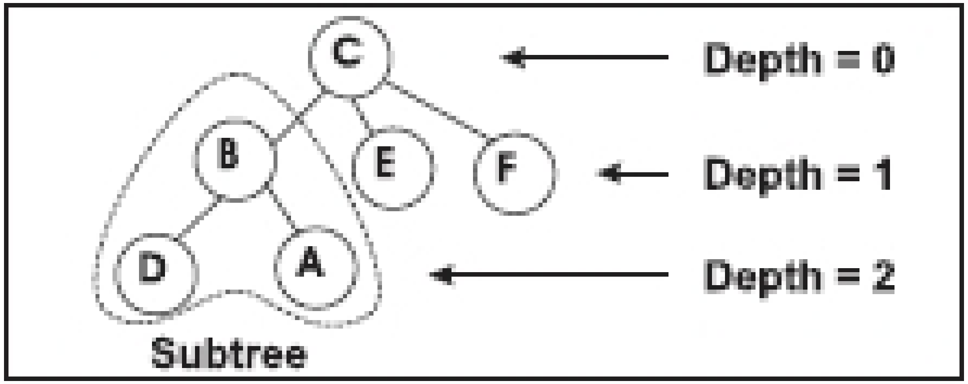
# Example Roadmap Method #2: Spanning Tree Method (Peasgood, et al., 2008)




Original planning problem

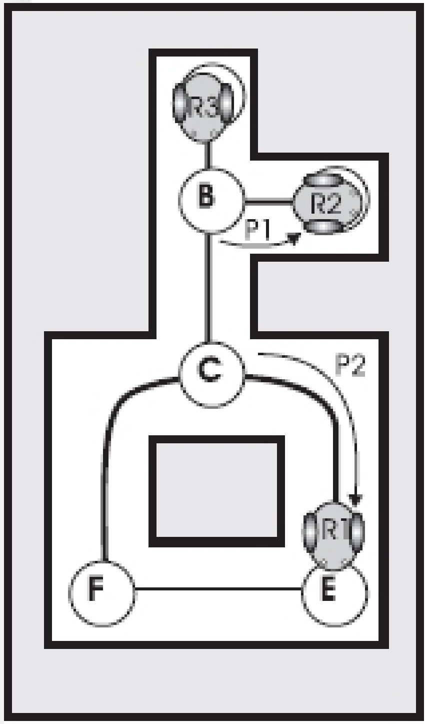



Graph-based map

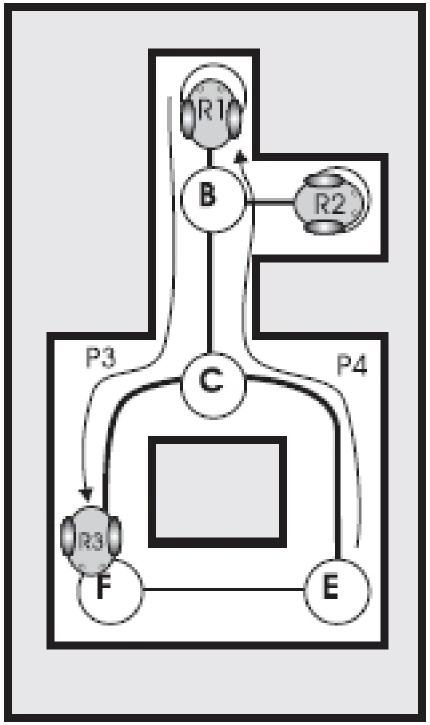



Spanning tree for the graph representation

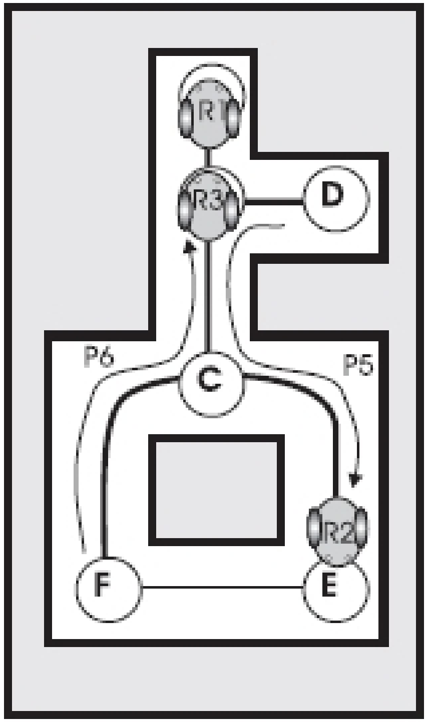
# Example Roadmap Method #2: Spanning Tree Method (con't.) (Peasgood, et al., 2008)



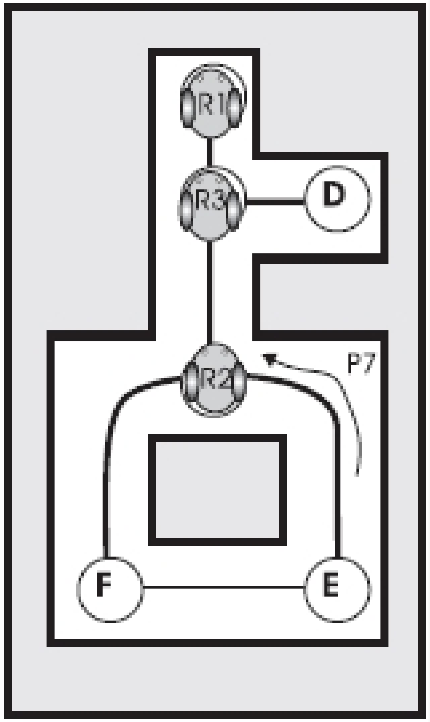

Phase 1




Phase 2a



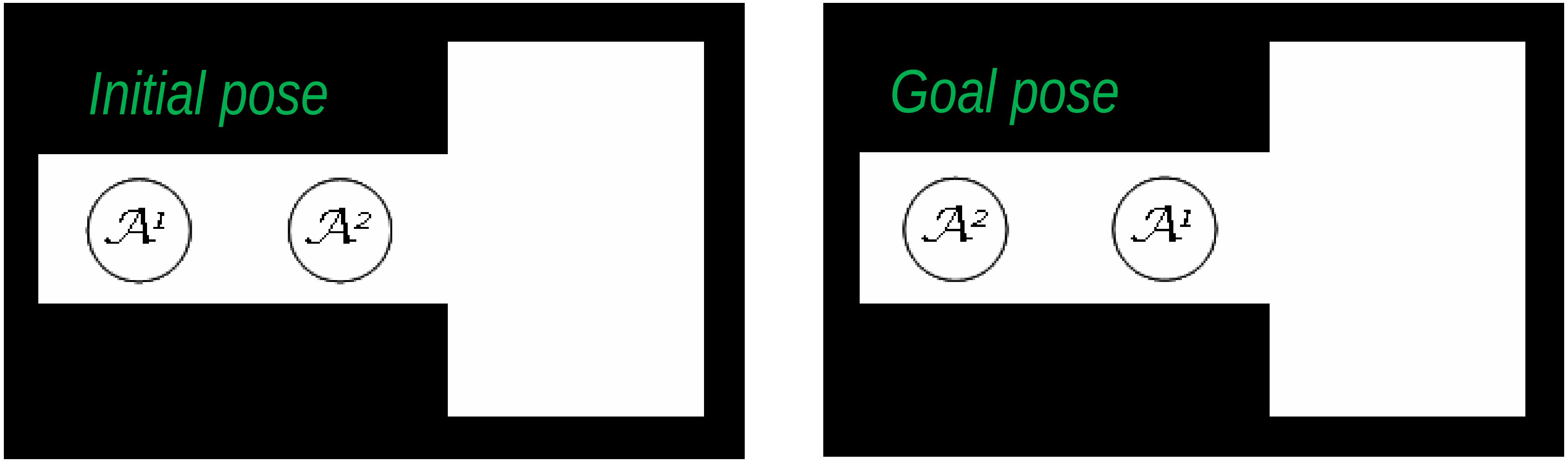

Phase 2b




Phase 3

# Decoupled Approaches

- Trade off solution quality for efficiency by solving parts of the problem independently
- Most common:
  - Plan individual paths for robots
  - Then, plan to avoid collisions
- Decoupled techniques lose completeness:




Situation that is hard for decoupled approaches to solve

# Two Types of Decoupled Approaches

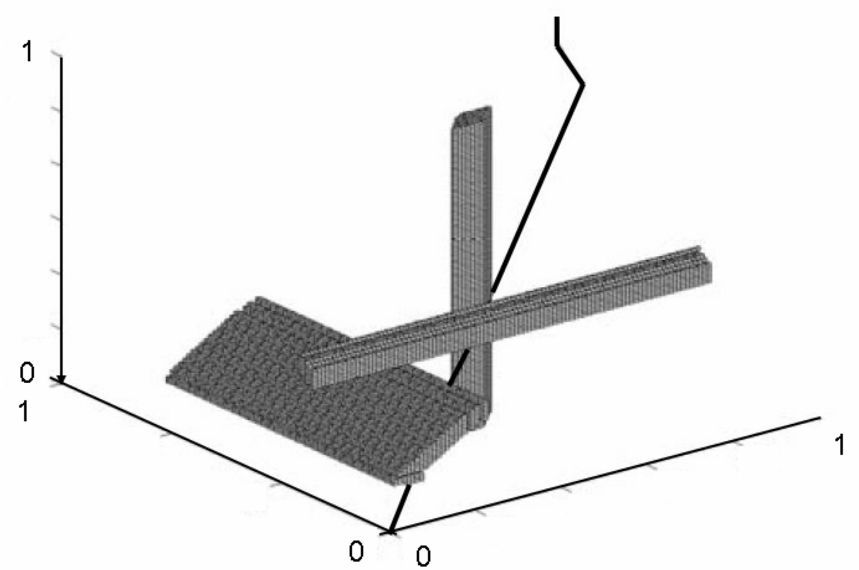
- Prioritized planning
  - Consider robots one at a time, in priority order
  - Plan for robot *i* by considering previous *i* –1 robots as moving obstacles
- Path coordination
  - Plan independent paths for each robot
  - Plan velocities to avoid collisions

# Prioritized Planning Approach

- Priorities assigned to robots
  - Randomly
  - Determined from motion constraints (i.e., more constrained robots have higher priority)
- Extend configuration space to account for time
- Plan path for first robot using any single-robot path planning approach
- Path for successive robots treats higher-priority robots as moving obstacles

# Path Coordination Approach

- Decouples problem into (1) path planning and (2) velocity planning
- First, generate individual robot paths independently, using any single-robot path planner
- Then, generate velocity profiles for each robot to ensure collisions avoided




(from Guo, Parker, 2002)

# Multi-Robot 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 …

# Multi-Robot 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 …

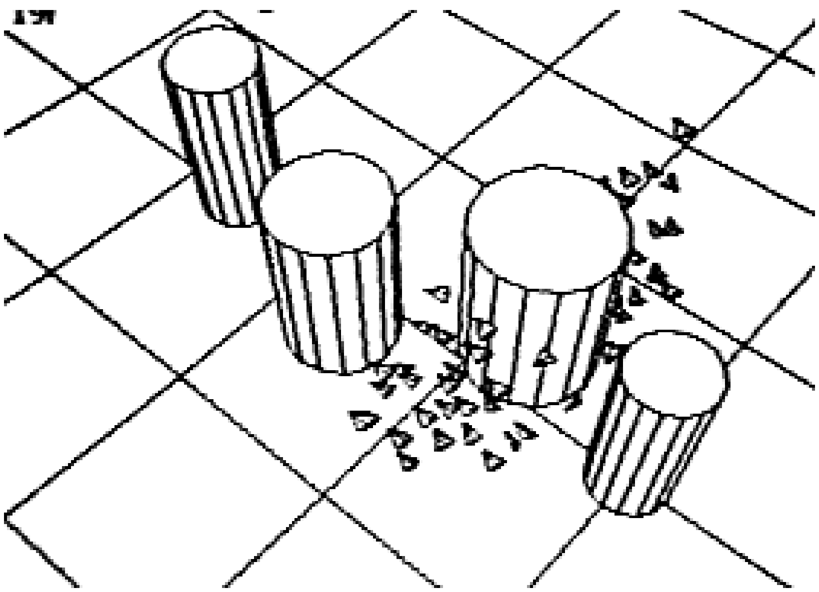
# Following / Swarming / Flocking / Schooling

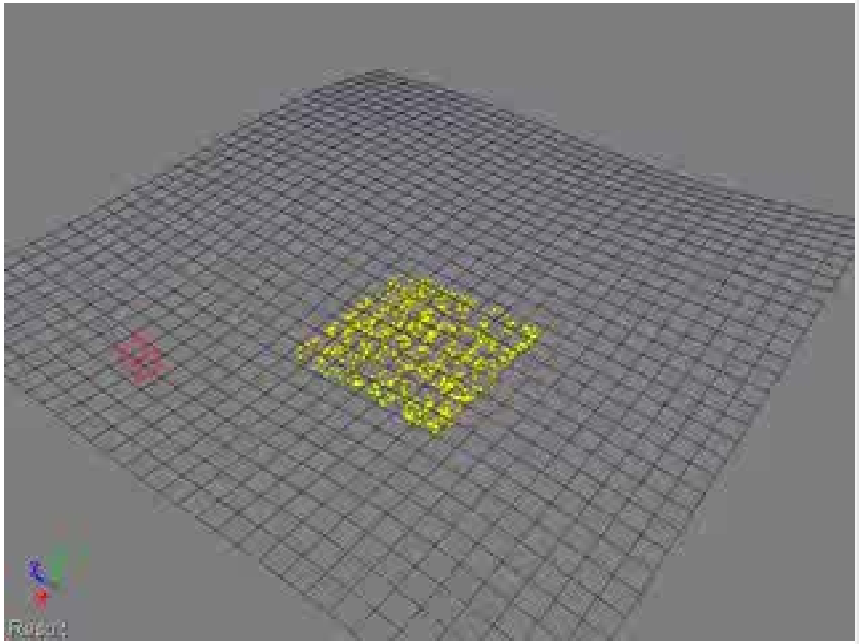
- Natural flocks consist of two balanced, opposing behaviors:
  - Desire to stay close to flock
  - Desire to avoid collisions with flock

- Why desire to stay close to flock?
  - In natural systems:
    - Protection from predators
    - Statistically improving survival of gene pool from predator attacks
    - Profit from a larger effective search pattern for food
    - Advantages for social and mating activities

# Craig Reynolds (1987) Developed Boids

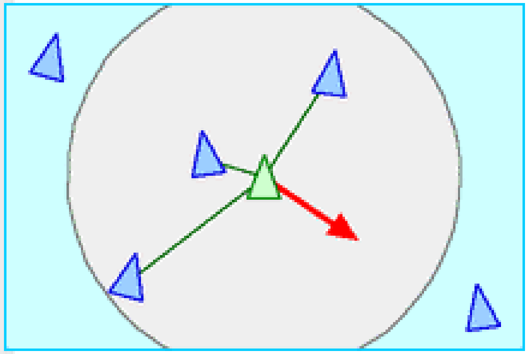
- “Flocks, Herds, and Schools: A Distributed Behavioral Model”, Craig Reynolds, *Computer Graphics*, 21(4), July 1987, pgs. 25-34.



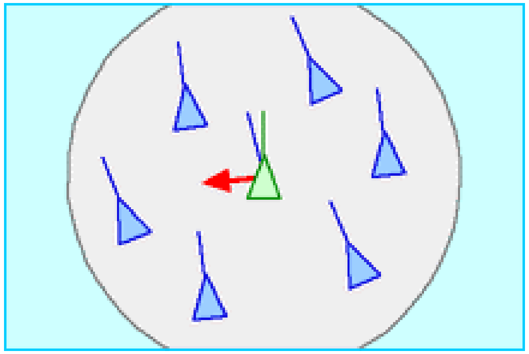


Simulated boid flock avoiding cylindrical obstacles

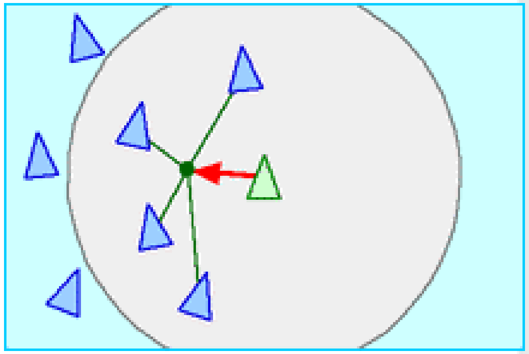
# How do Boids work?



Separation: steer to avoid crowding local flockmates

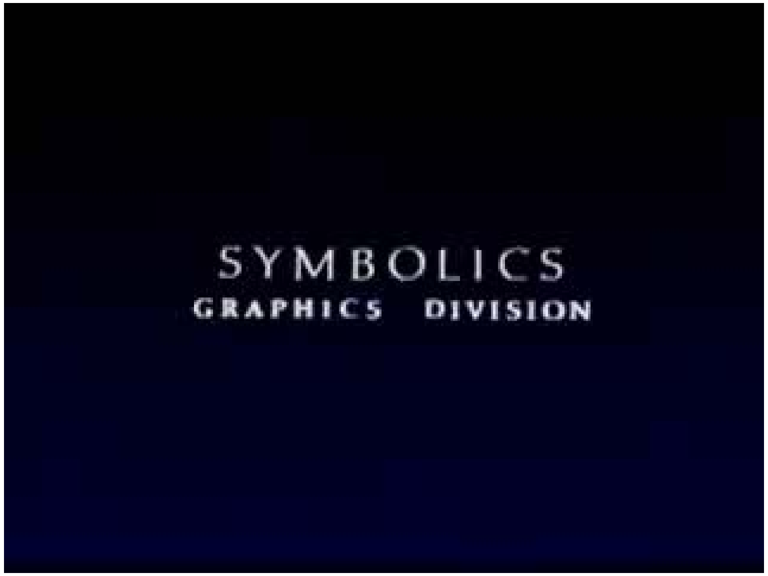


Alignment: steer towards average heading of local flockmates



Cohesion: steer to move Toward the average position of local flockmates

# Boids Movie
# "Stanley and Stella in Breaking the Ice"



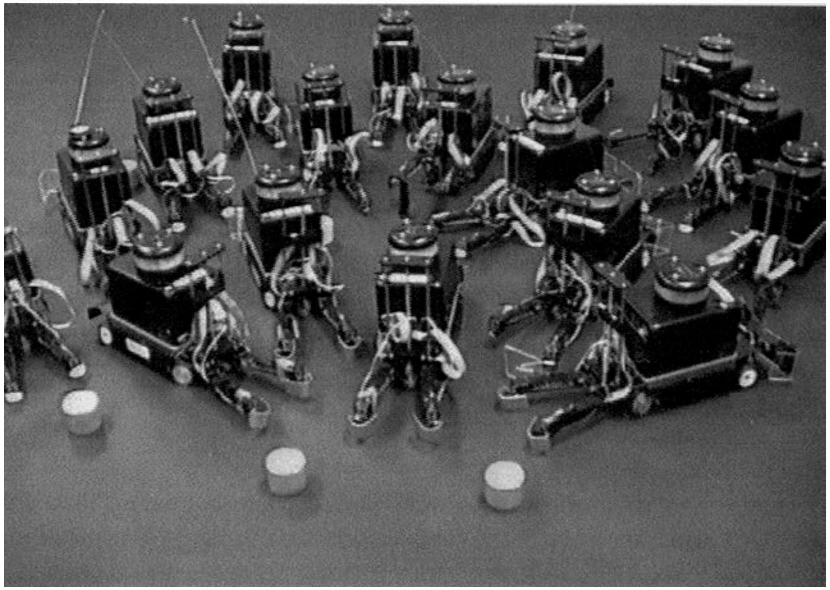

http://odyssey3d.stores.yahoo.net/comanclascli2.html

# Translating these Behaviors to Code on Robots

- Work of Mataric, 1994

- General Idea:
  - Use “local” control laws to generate desired “global” behavior

- The Robots:
  - 12” long
  - 4 wheels
  - Bump sensors around body
  - Radio system for:
    - Localization
    - Communication
    - Data collection
    - “Kin” recognition

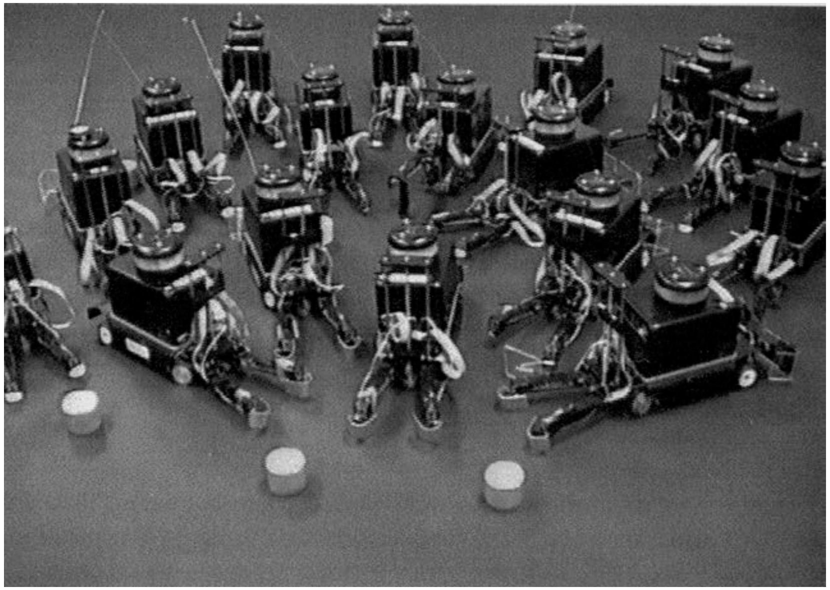


**The Nerd Herd: Mataric, MIT, 1994**

# The Nerd Herd Approach

- Fundamental principle: Define *basis behaviors* as general building blocks for synthesizing group behavior

- Set of basis behaviors proposed:
  - Avoidance
  - Save-wandering
  - Following
  - Aggregation
  - Dispersion
  - Homing



- Combine basis behaviors into higher-level group behaviors:
  - Flocking
  - Foraging

# Safe-Wandering Algorithm

- Avoid-Kin:
  - Whenever an agent is within d_avoid
    - If the nearest agent is on the left
      - Turn right
      - Otherwise, turn left
- Avoid-Everything-Else
  - Whenever an obstacle is within d_avoid
    - If obstacle is on right only, turn left
    - If obstacle is on left only, turn right
    - After 3 consecutive identical turns, backup and turn
    - If an obstacle is on both sides, stop and wait.
    - If an obstacle persists on both sides, turn randomly and back up
- Move-Around:
  - Otherwise move forward by d_forward, turn randomly

# Following Algorithm

```
Follow:
  – Whenever an agent is within d_follow
      • If an agent is on the right only, turn
        right
      • If an agent is on the left only, turn
        left
```

If sufficient robot density, safe_wandering + follow yield more complex behaviors:

- e.g., osmotropotaxic behavior of ants: unidirectional lanes

# Dispersion Algorithm

**Dispersion:**

- Whenever one or more agents are within d_disperse
  - Move away from Centroid_disperse

# Aggregation Algorithm

Aggregate:

- Whenever nearest agent is outside d_aggregate
  - Turn toward the local centroid_aggregate, go.
- Otherwise, stop.

# Homing Algorithm

```
Home:
  – Whenever at home
      • Stop
  – Otherwise, turn toward home, go.
```

# Generating Flocking Through Behavior Combinations

- **Flock:**
  - Sum weighted outputs from Safe-Wander, Disperse, Aggregate, and Home

Movie of Nerd Herd (~1994)

# More 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

SwarmBots

# McLurkin's Robot Swarms

- Approach to generating behaviors is similar to Mataric's, in principle

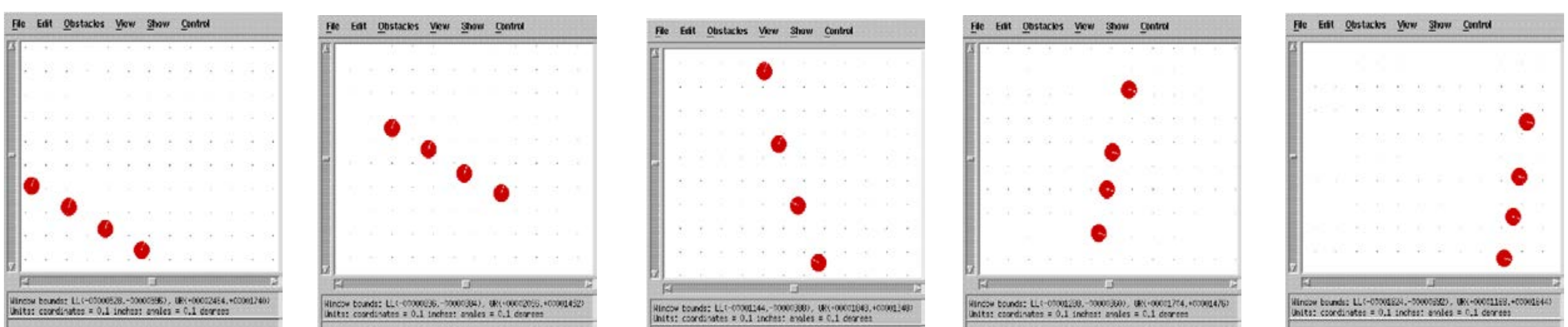
- Primary differences:
  - Algorithms more tuned to the SwarmBot
  - More exhaustively tested
  - Parameters explored,
  - More kinds of behaviors,
  - etc.

# SwarmBots in Action

# Motion Coordination: Formation-Keeping

- Objective:
  - Robots maintain specific formation while collectively moving along path
- Examples:
  - Column formation:

  - Line formation:



# Formations

Key Issues:

- What is desired formation?
- How do robots determine their desired position in the formation?
- How do robots determine their actual position in the formation?
- How do robots move to ensure that formation is maintained?
- What should robots do if there are obstacles?
- How do we evaluate robot formation performance?

# Issue in Formation Keeping: Local vs. Global Control

- Local control laws:
  - No robot has all pertinent information
  - Appealing because of their simplicity and potential to generate globally emergent functionality
  - But, may be difficult to design to achieve desired group behavior

- Global control laws:
  - Centralized controller (or all robots) possess all pertinent information
  - Generally allow more coherent cooperation
  - But, usually increases inter-agent communication

# Descriptions: Global Goals, Global Knowledge, Local Control

- Global Goals:
  - Specify overall mission the team must accomplish
  - Typically imposed by centralized controller
  - May be known at compile time, or only at run-time
- Global Knowledge:
  - Additional information needed to achieve global goals
  - E.g., information on capabilities of other robots, on environment, etc.
- Local Control:
  - Based upon proximate environment of robot
  - Derived from sensory feedback
  - Enables reactive response to dynamic environmental changes

# Tradeoffs between Global and Local Control

- Questions to be addressed:
  - How static is global knowledge?
  - How difficult is it to obtain reliable global knowledge?
  - How badly will performance degrade without use of global knowledge?
  - How difficult is it to use global knowledge?
  - How costly is it to violate global goals?
- In general:
  - The more unknown the global information is, the more dependence on local control

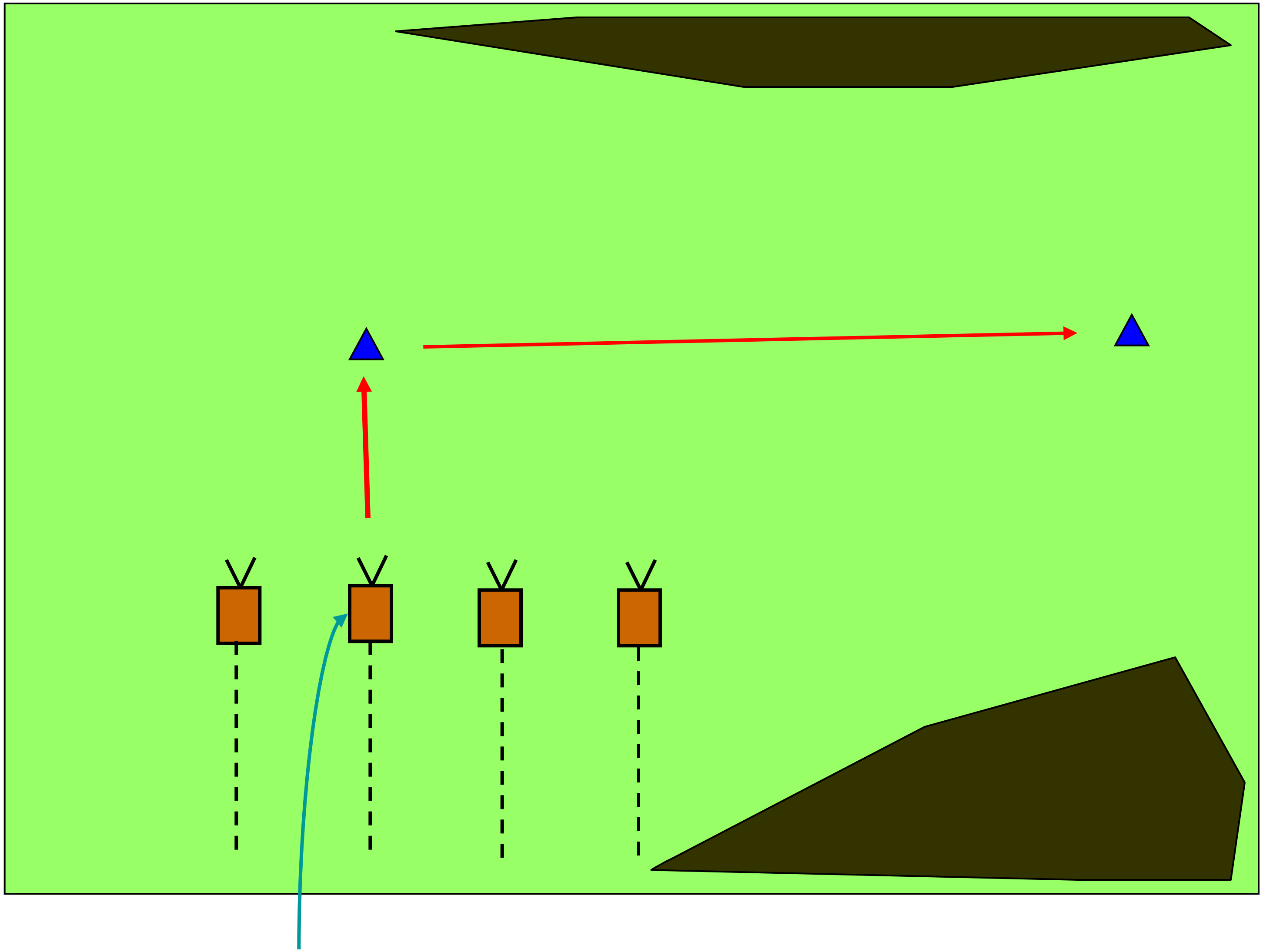
# Demonstration of Tradeoffs in Formation-Keeping

- Measure of performance: Cumulative formation error:

$$\sum_{t=0}^{t_{\max}} \sum_{i \neq leader} d_i(t)$$ where $d_i(t)$ = distance robot $i$ is from ideal formation position at time $t$

- Strategies to investigate:
  - Local control alone
  - Local control + global goal
  - Local control + global goal + partial global knowledge
  - Local control + global goal + more complete global knowledge

# Formation Keeping Objective

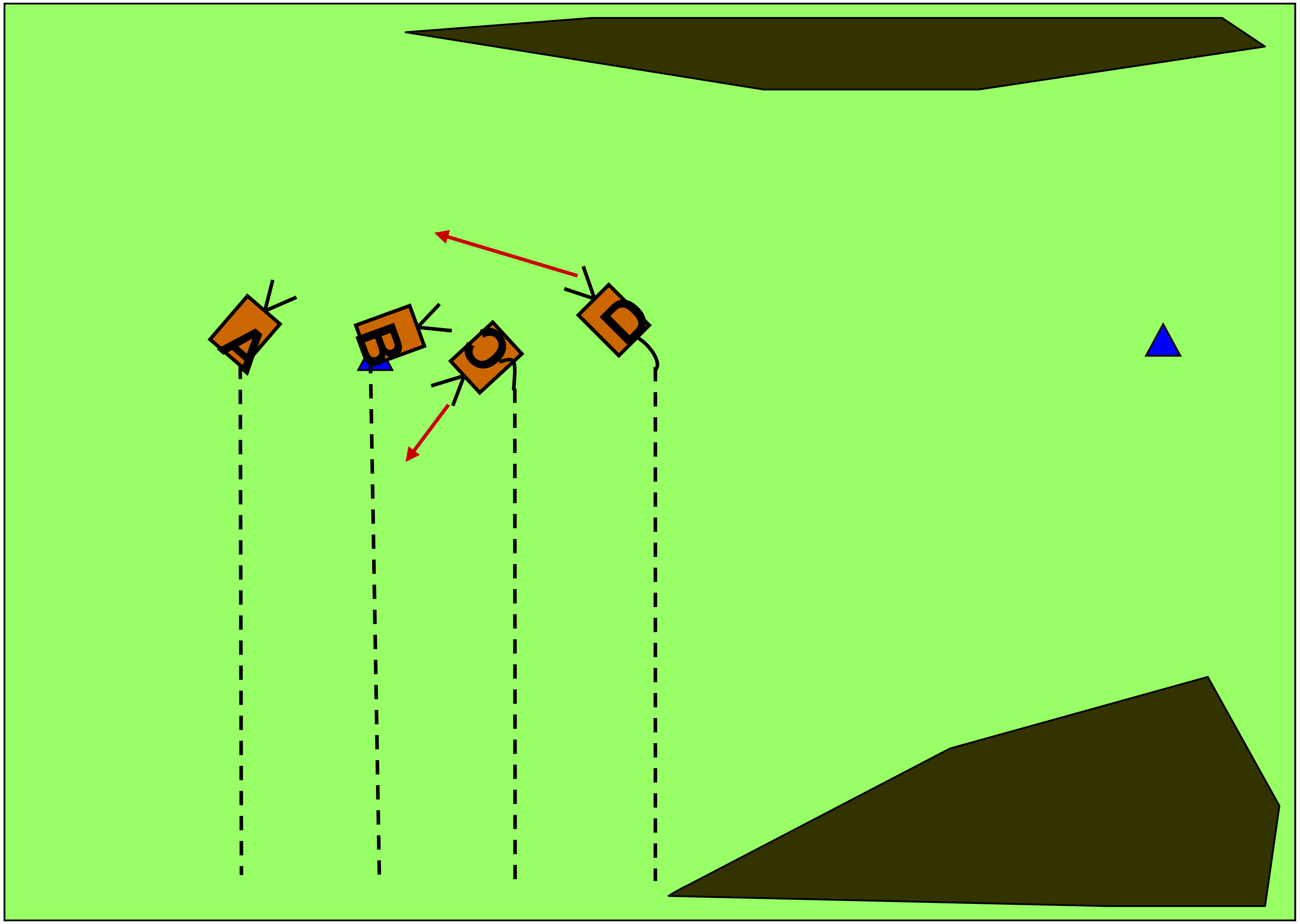


Leader

# Strategy I: Local Control

- Group leader knows path waypoints
- Each robot assigned local leader + position offset from local leader
- As group leader moves, individual robots maintain relative position to local leaders

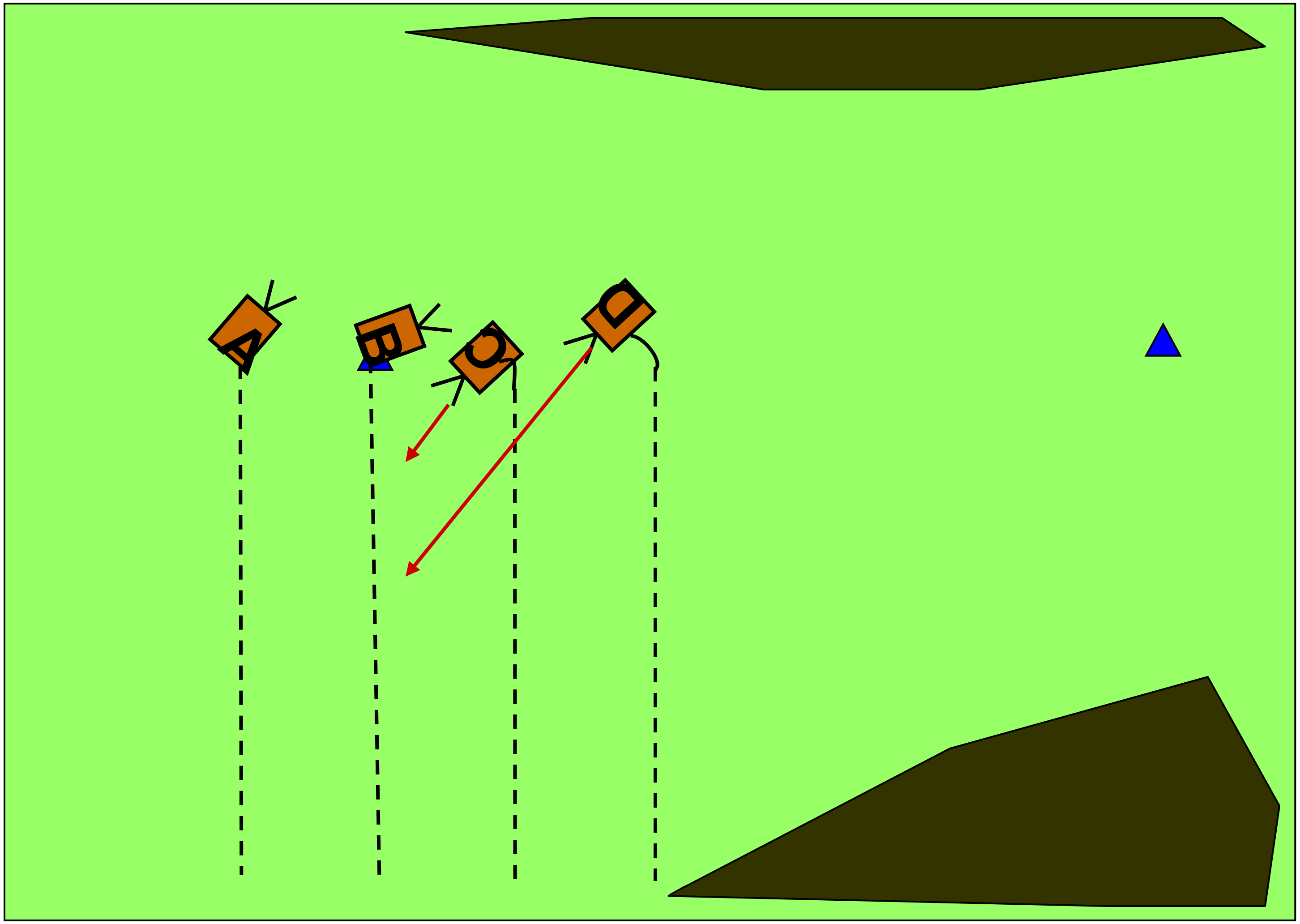
# Results of Strategy I



# Strategy II: Local Control + Global Goal

- Group leader knows path waypoints
- Each robot assigned global leader + position offset from global leader
- As group leader moves, individual robots maintain relative position to global leader

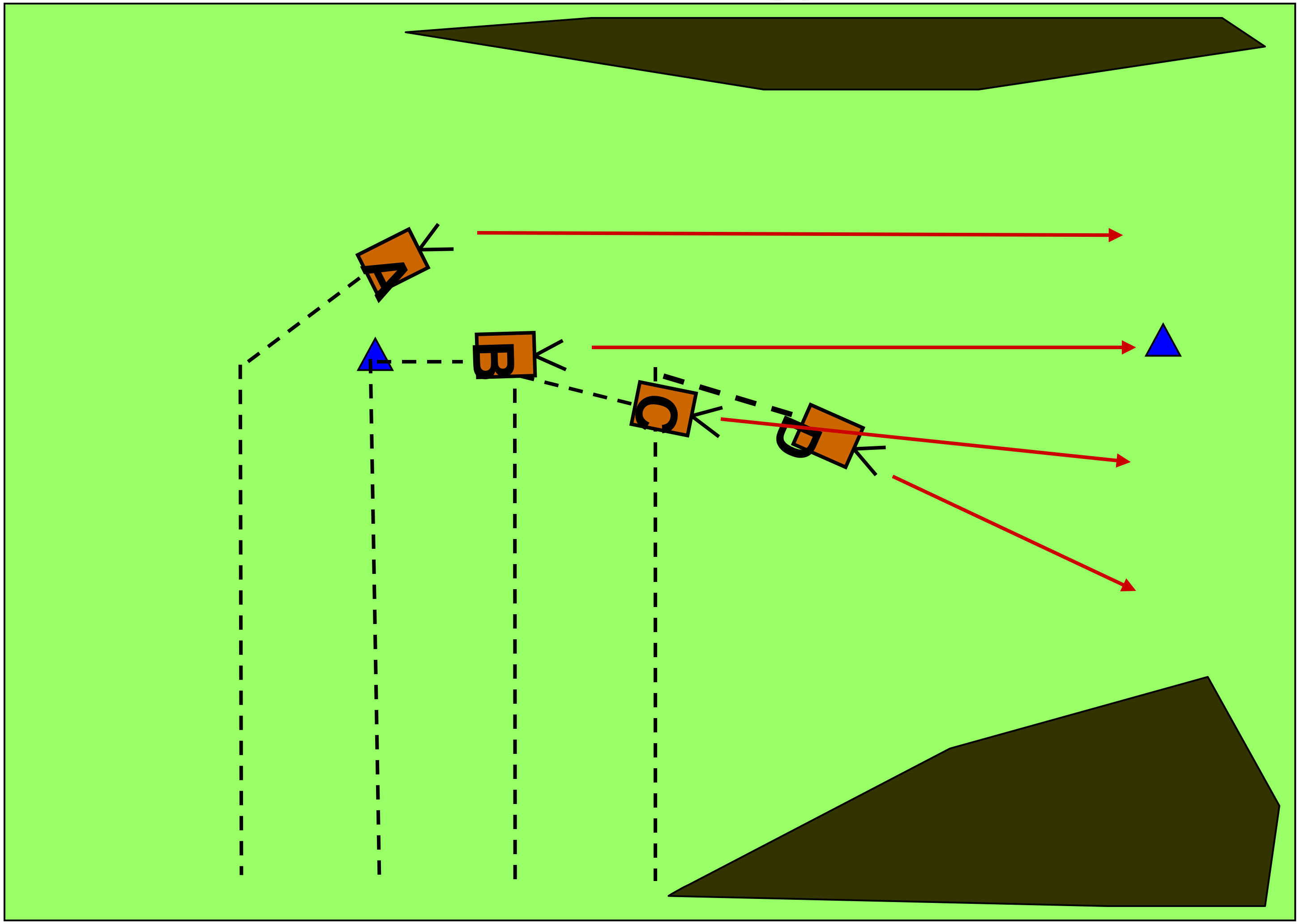
# Results of Strategy II



# Strategy III: Local Control + Global Goal + Partial Global Knowledge

- Group leader knows path waypoints
- Each robot assigned global leader + position offset from global leader
- Each robot knows next waypoint
- As group leader moves, individual robots maintain relative position to global leader

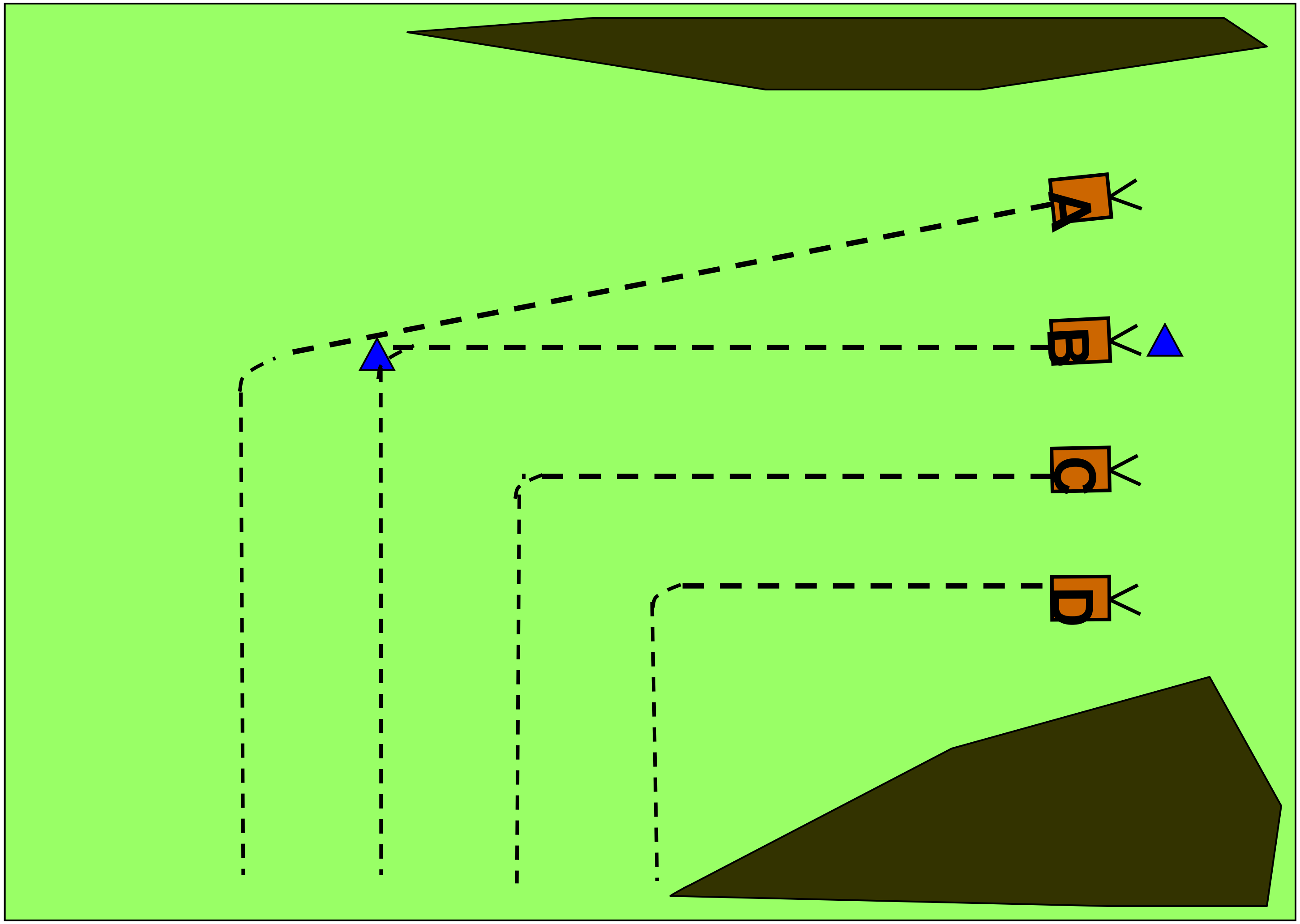
# Results of Strategy III



# Strategy IV: Local Control + Global Goal + More Complete Global Knowledge

- Group leader knows path waypoints
- Each robot assigned global leader + position offset from global leader
- Each robot knows current and next waypoints
- As group leader moves, individual robots maintain relative position to global leader

# Results of Strategy IV



# Time and Cumulative Formation Error Results

## Time Required to Complete Mission

Strategy IV *
Strategy III *
Strategy II ******** ****
Strategy I ********* *

0 10 20 30 40 50 Time

## Normalized Cumulative Formation Error

Strategy IV ***
Strategy III ***
Strategy II ******** **
Strategy I ** **** ** *** **

0 50 100 150 200 250 300 Error

# Summary of this Formation-Keeping Control Case Study

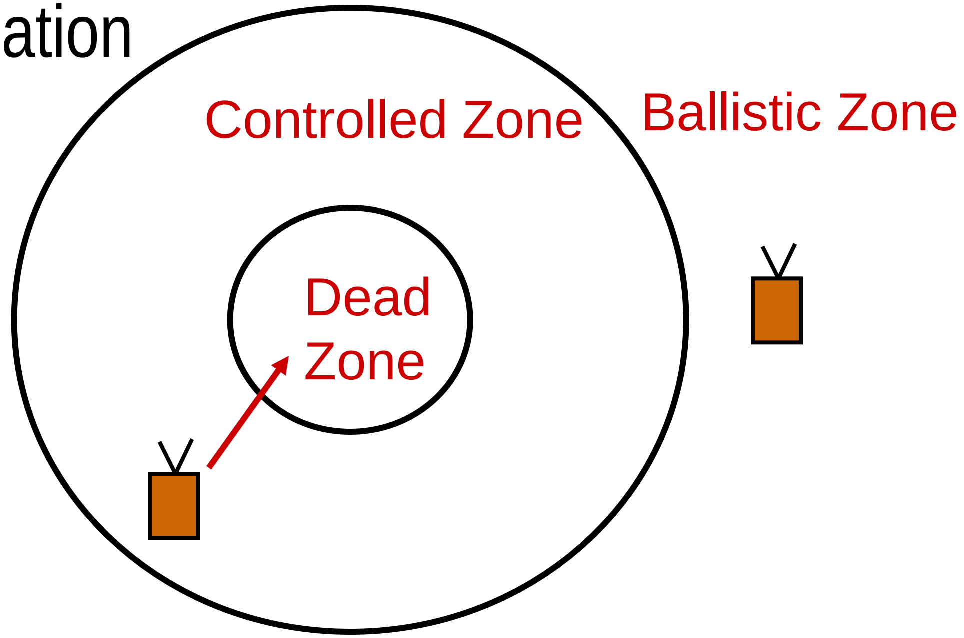
- Important to achieve proper balance between local and global knowledge and goals
- Static global knowledge ==> easy to use as global control law
- Local knowledge ==> appropriate when can approximate global knowledge
- Local control information should be used to ground global knowledge in the current situation.

# Another Case Study for Formation-Keeping: Balch & Arkin's Behavior-Based Control

- Applications:
  - Automated scouting (military)
  - Search and rescue
  - Agricultural coverge
  - Security patrols
- Approach:
  - Motor schemas
  - Fully integrated obstacle avoidance

# Motor Schemas Used for Formation-Keeping

- Move-to-goal
- Avoid-static-obstacle
- Avoid-robot
- Maintain-formation

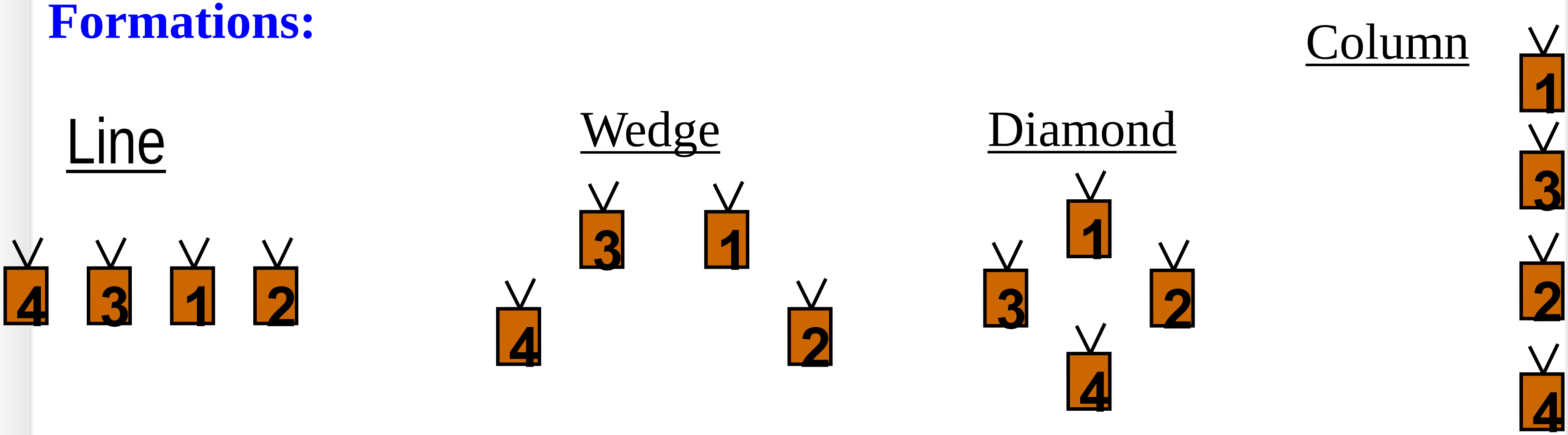


# Formation and Obstacle Avoidance

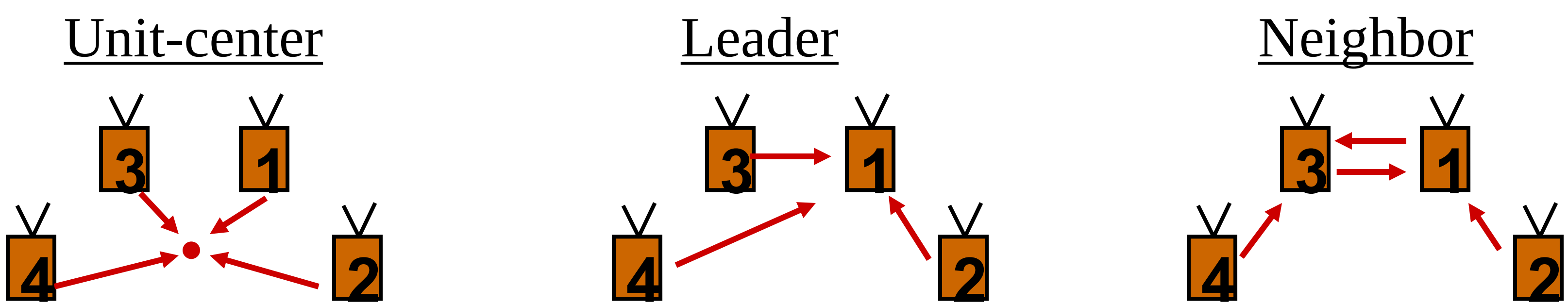
- Barriers -- choices for handling include:
  - Move as a unit around barrier
  - Divide into subgroupcs
- Choice depends upon relative strengths of behaviors

# Balch's Formation Types and Position Determination

## Formations:




## Position Determination:



# Balch's Formation Results

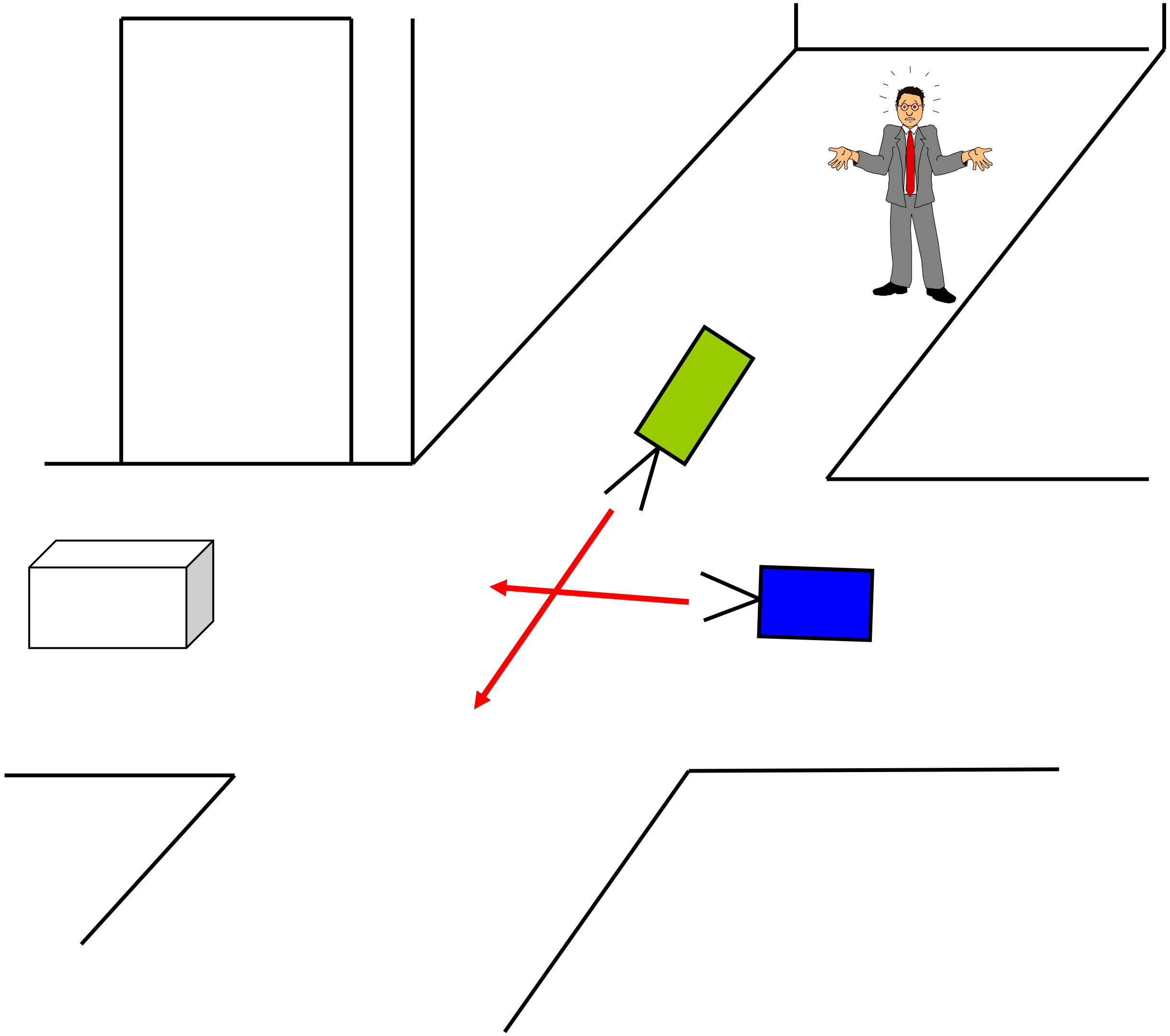
- For 90 degree turns:
  - Diamond formation best with unit-center-reference
  - Wedge, line formations best with leader-reference
- For obstacle-rich environments:
  - Column formation best with either unit-center or leader-reference
- Most cases:
  - Unit-center better than leader-center
  - Except:
    - If using human leader, not reasonable to expect to use unit-center
    - Unit-center requires transmitter and receiver for all robots, whereas leader-center only requires transmitter at leader plus receivers for all robots
    - Passive sensors are difficult to use for unit-center

# Coordinating Multiple Robots Through Traffic Rules (Kato et al, Japan)

- Issues:
  - Collisions
  - Deadlocks
  - Congestion

- Possible approaches:
  - Communication
  - Local collision avoidance
  - Traffic rules

# Typical Problem Situation for Traffic Rules



# Traffic Rule Application System (TRAS)

- "Traffic Rule": imposes a certain level of order on mobile objects, such as mobile robots and people, and work environments
- Rules constructed by considering:
  - Work environment
  - Performance of mobile objects
  - Quantity of mobile objects
- Robots must know:
  - Current position
  - Current sensory information
  - Global map information

# Traffic Rules

- Keep sufficient space in front
- Keep sufficient side space
- Maintain passage zone
- Intersection crossing:
  - Preference to right turn
  - Preference toward a right-side mobile object
  - Collision avoidance
- Deadlock avoidance:
  - Preference at intersections
  - Replan if route blocked

# Control of Robots in Traffic Management

1. Plan shortest route to goal
2. Extract local maps from global map for route and intersections
3. Move along planned path
4. Determine sensor-detecting range re: traffic rules
5. Observe workspace, using sensors
6. Detect obstacles
7. Judge, according to traffic rules, whether collision will occur
8. Decide how to act
9. Move or stop
10. Return to step 2

# Multi-Robot 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 …

# Cooperative Tracking (CMOMMT)

## Cooperative Multi-robot Observation of Multiple Moving Targets

Definition:

Given: $S$ : 2-D bounded, enclosed spatial region
$V$ : team of $m$ robot vehicles, $v_i$, $i = 1, 2, \ldots, m$, with 360° FOV sensors
$O(t)$: set of $n$ targets, $o_j(t)$, $j = 1, 2, \ldots, n$, such that target $o_j(t)$ is in $S$ at $t$

Define $m$ x $n$ matrix $B(t)$:

$$B(t) = [b_{ij}(t)]_{mxn} \text{ such that } b_{ij}(t) = \begin{cases} 1 & \text{if robot } v_i \text{ is observing target } o_j(t) \text{ in } S \text{ at time } t \\ 0 & \text{otherwise} \end{cases}$$

Goal:

$$\text{Maximize:} \quad A = \sum_{t=1}^{T} \sum_{j=1}^{n} \frac{g(B(t),j)}{T}$$

$$\text{where } g(B(t),j) = \begin{cases} 1 & \text{if there exists an } i \text{ such that } b_{ij}(t) = 1 \\ 0 & \text{otherwise} \end{cases}$$

# Motivation for Studying Cooperative Observation

- Automatic location/tracking of:
  - Other mobile robots
  - Items in a warehouse or factory that might move during search
  - People in a search/rescue effort
  - Adversarial targets in surveillance and reconnaissance
- Monitoring automated processes:
  - In assembly workcell
  - Verifying parts or subassembly configurations
- Medical applications:
  - Moving cameras to keep designated areas (e.g. particular tissue) in continuous view

# Cooperative Observation Research Issues

- Physical, sensor-based tracking
- Prediction of object movements
- Sensor fusion across robots
- Multi-robot communication
- Selection of object to track
- Distributed navigation
- Achieving adequate terrain coverage

Many possible problem variations:

- Relative numbers and speeds of robots
- Limited FOV sensors
- Availability of communication
- Robots heterogeneous in sensing and movement capabilities

# Cooperative Observation Approaches

- Art Gallery Theorems -- O'Rourke, 1987; Briggs, 1995

  *Works for static sensor placements*
- Searchlight Scheduling and Polygon Search -- Sugihara *et al.*, 1990; Suzuki and Yamashita, 1992; Crass *et al.*, 1995

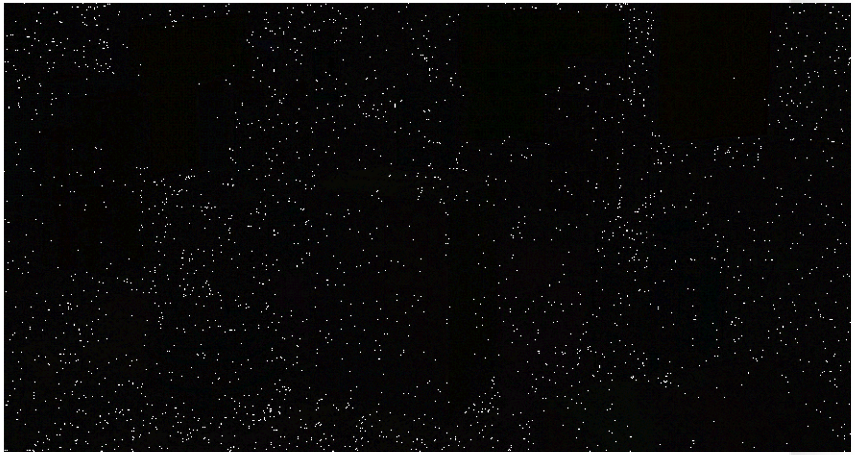
  *Addresses fixed sensor placements; often assume one searcher*
- Visibility-Based Motion Planning -- Lavalle *et al.*, 1997

  *Focuses on single robots and targest*
- Multi-target tracking and/or weapons assignment -- Bar-Shalom, 1978, 1990; Blackman, 1986; Fox *et al.*, 1994

  *Focuses on target trajectory derivation*
- Multi-Robot Surveillance -- Everett *et al.*, 1993; Durfee *et al.*, 1987; Wesson *et al.*, 1981

  *Works for static sensor placements*
- CMOMMT– Parker, 1999

  *Uses weighted local force vectors*



# Summary of Motion Coordination Research

- Many issues studied by the field:
  - Multi-robot path planning
  - Traffic control
  - Formation generation
  - Formation keeping
  - Target tracking
  - Target search
  - Multi-robot docking

- Approaches are usually specific to given application

# Open Issues in Multi-Robot Path Planning and Motion Coordination

- Scaling to larger numbers of robots (i.e., thousands)
- Extensions to 3 dimensions (i.e., for aerial robots)
- Handling highly stochastic environments
- Dealing with dynamic, online replanning
- Creating provably correct interaction strategies
- Incorporating practical motion and sensing constraints
- Integrating onto physical robots

# For more information on multi-robot path planning and motion coordination

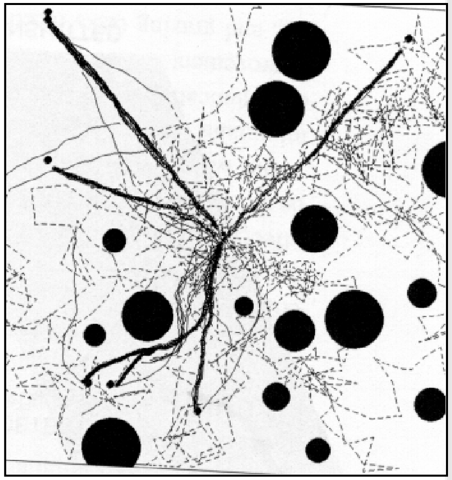
- Lynne E. Parker, “Path planning and motion coordination in multiple mobile robot teams”, in *Encyclopedia of Complexity and System Science*, Robert A. Meyers, Editor-in-Chief, Springer, 2009.

# Multi-Robot Communication

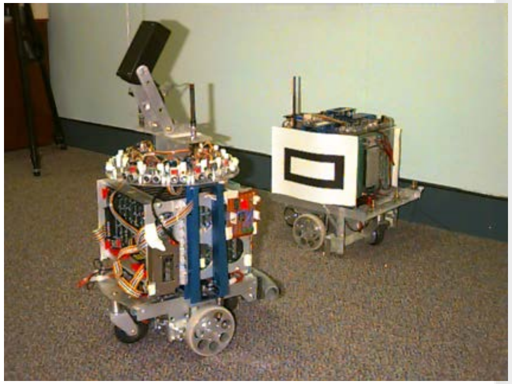
Objective of communication: Enable robots to exchange state and environmental information with a minimum bandwidth requirement

Issues of particular importance:

- Information content
- Explicit vs. Implicit
- Local vs. Global
- Impact of bandwidth restrictions
- "Awareness"
- Medium: radio, IR, chemical scents, "breadcrumbs", etc.
- Symbol grounding



Balch and Arkin



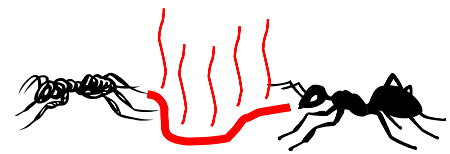
Jung and Zelinsky

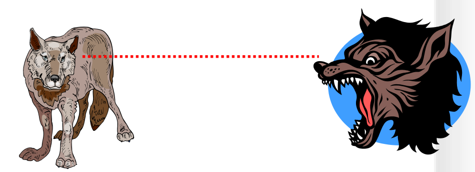
# The Nature of Communication

One definition of communication:

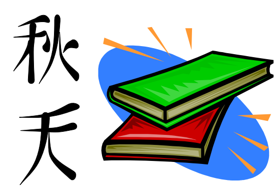
"An interaction whereby a signal is generated by an *emitter* and 'interpreted' by a *receiver*"

- Emission and reception may be separated in space and/or time.
- Signaling and interpretation may innate or learned (usually combination of both)





- Cooperative communication examples:
  - Pheromones laid by ants foraging food
    - Time delayed, innate
  - Posturing by animals during conflicts/mating etc.
    - Separated in space, learnt with innate biases
  - Writing
    - Possibly separated in space & time,
      mostly learned with innate support and scaffolding



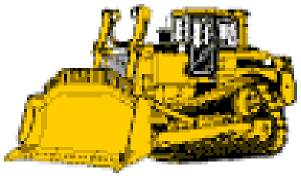
# Multi-Robot Communication Taxonomy

Put forth by Dudek (1993) (this is part of larger multi-robot taxonomy):

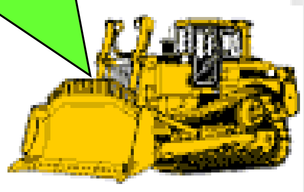
- Communication range:
  - None
  - Near
  - Infinite
- Communication topology:
  - Broadcast
  - Addressed
  - Tree
  - Graph
- Communication bandwidth
  - High (i.e., communication is essentially "free")
  - Motion-related (i.e., motion and communication costs are about the same)
  - Low (i.e., communication costs are very high
  - Zero (i.e., no communication is available)

# Explicit Communication

- Defined as those actions that have the express goal of transferring information from one robot to another
- Usually involves:
  - Intermittent requests
  - Status information
  - Updates of sensory or model information



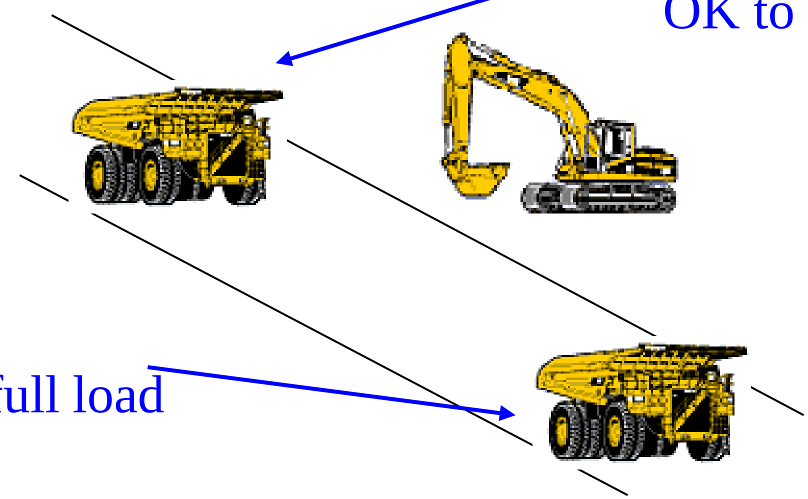
- Need to determine:
  - What to communicate
  - When to communicate
  - How to communicate
  - To whom to communicate




- Communications medium has significant impact
  - Range
  - Bandwidth
  - Rate of failure

# Implicit Communication

- Defined as communication “through the world”
- Two primary types:
  - Robot senses aspect of world that is a side-effect of another’s actions
  - Robot senses another’s actions



# Three Key Considerations in Multi-Robot Communication

- Is communication needed at all?
- Over what range should communication be permitted?
- What should the information content be?

# Is Communication Needed At All?

- Keep in mind:
  - Communication is not free, and can be unreliable
  - In hostile environments, electronic countermeasures may be in effect
- Major roles of communication:
  - Synchronization of action: ensuring coordination in task ordering
  - Information exchange: sharing different information gained from different perspectives
  - Negotiations: who does what?
- Many studies have shown:
  - Significantly higher group performance using communication
  - However, communication does not always need to be explicit

# Over What Range Should Communication Be Permitted?

- Tacit assumption: wider range is better
- But, not necessarily the case
- Studies have shown: higher communication range can lead to decreased societal performance

- One approach for balancing communication range and cost (Yoshida '95):
  - Probabilistic approach that minimizes communication delay time between robots
  - Balance out communication flow (input, processing capacity, and output) to obtain optimal range

# What Should the Information Content Be?

- Research studies have shown:
  - Explicit communication improves performance significantly in tasks involving little implicit communication
  - Communication is not essential in tasks that include implicit communication
  - More complex communication strategies (e.g., goals) often offer little benefit over basic (state) information → “display” behavior is a rich communication method

# Summary of Multi-Robot Communication

- Many types:
  - Implicit vs. explicit
  - Local vs. global
  - Iconic vs. symbolic
  - General “awareness”

- Proper approach to communication dependent upon application:
  - Communication availability
  - Range of communication
  - Bandwidth limitations
  - Language of robots
  - Etc.