EE631 Cooperating Autonomous Mobile Robots

Lecture: Multi-Robot Motion Planning

Prof. Yi Guo
ECE Department

Plan

- Introduction
- Premises and Problem Statement
- A Multi-Robot Motion Planning Algorithm
- Implementation Examples and Issues

Motion Planning

- Path planning considers mainly spatial information
- Motion planning considers both spatial and temporal information
- The problem is more challenging in multi-robot systems, where collisions between robots need to be resolved

Background

- Motion planning in dynamic environments with moving obstacles is NP-hard
- Simple reactive motion planning strategies cannot guarantee deadlock free and convergence
- Previous results either obtain optimal solutions through centralized and exhaustive computing, or achieve distributed implementations without considering optimization issues

- Distributed solutions (e.g., [Azarm & Schmidt, Carpin & Pagello]) use negotiation or insert random time delays to resolve conflicts
- Recent results (e.g., [LaValle & Hutchinson]) consider performance through centralized computing, not capable of real time re-planning

- Outdoor environment is more challenging with 3D terrain features and the requirement for online re-planning
- Need to deal with the constraint of computation expenses, the requirements of real time control and robust solutions

- Introduce a new multi-robot motion planning algorithm:
  - Distributed;
  - A global performance measurement defined and minimized;
  - Capable of operation in outdoor environments and real time re-planning.

Assumptions

- Each robot has an assigned goal, and knows its start and goal locations.
- Pre-defined map available
  - Indoor: static polygonal obstacles;
  - Outdoor: terrain elevation and traversability based on grid representation.
- Onboard sensors detect discrepancy and revise map online
- Communication devices broadcast messages
- Robots move at constant fixed speeds
- Robots switch instantaneously between fixed speed and halting.

Problem Statement

- Multi-robot motion planning problem:

  Find collision-free sequence of traverse states for each robot from its start to its goal, minimizing:

$$\Gamma = \gamma_1 \max(T_1, T_2, \cdots, T_N) + \gamma_2 \sum_{i=1}^{N} I_i$$

$T_1, T_2, \cdots, T_N$ are the time for each robot to reach its goal

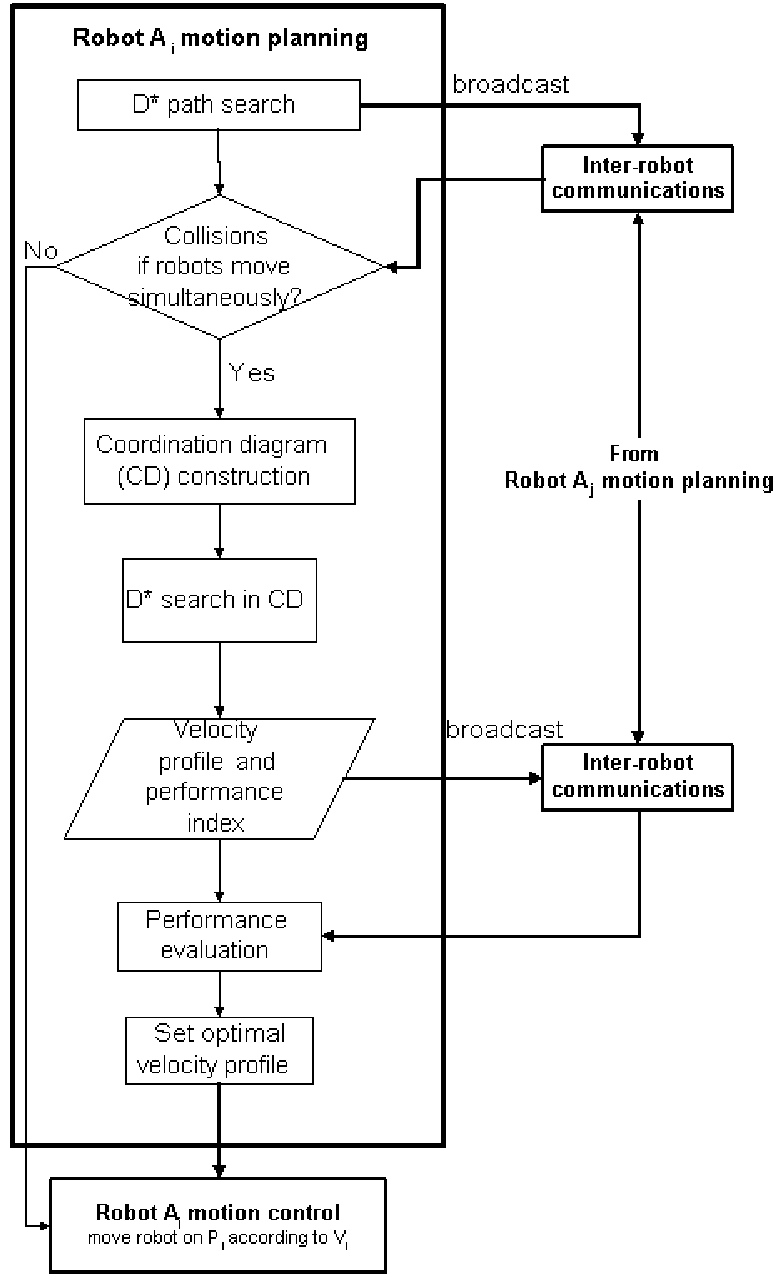
I_i is the idle time for robot i

γ_1, γ_2 are positive weighting constants

Multi-Robot Motion Planning Algorithm

- We propose to solve the computationally expensive problem by decomposing it into two modules: *path planning* and *velocity planning*.
- D* search method is applied in both modules, based on either geometric or schedule formulations.
- Optimization is achieved at the individual robot level by defining cost functions to minimize, and also at the team level by a global measurement function reflecting performance indices of interest as a team.
- Robustness design is incorporated by defining safety margins in both modules.

Flow chart diagram of algorithm:



Multi-Robot Motion Planning Algorithm

Step 1: Path planning:

D* search in free space $W^{e}_{free} = W \setminus W^{e}_{prohibit}$ produces optimal path P_i for each robot from the start to the goal minimizing cost function:

$$f_{pp} = \rho + \alpha_1 d + \alpha_2 s + \alpha_3 t$$

where: d is distance, s is slope,

t is penalty for turning, ρ is penalty on obstacles.

D* Paths in Outdoor Environment

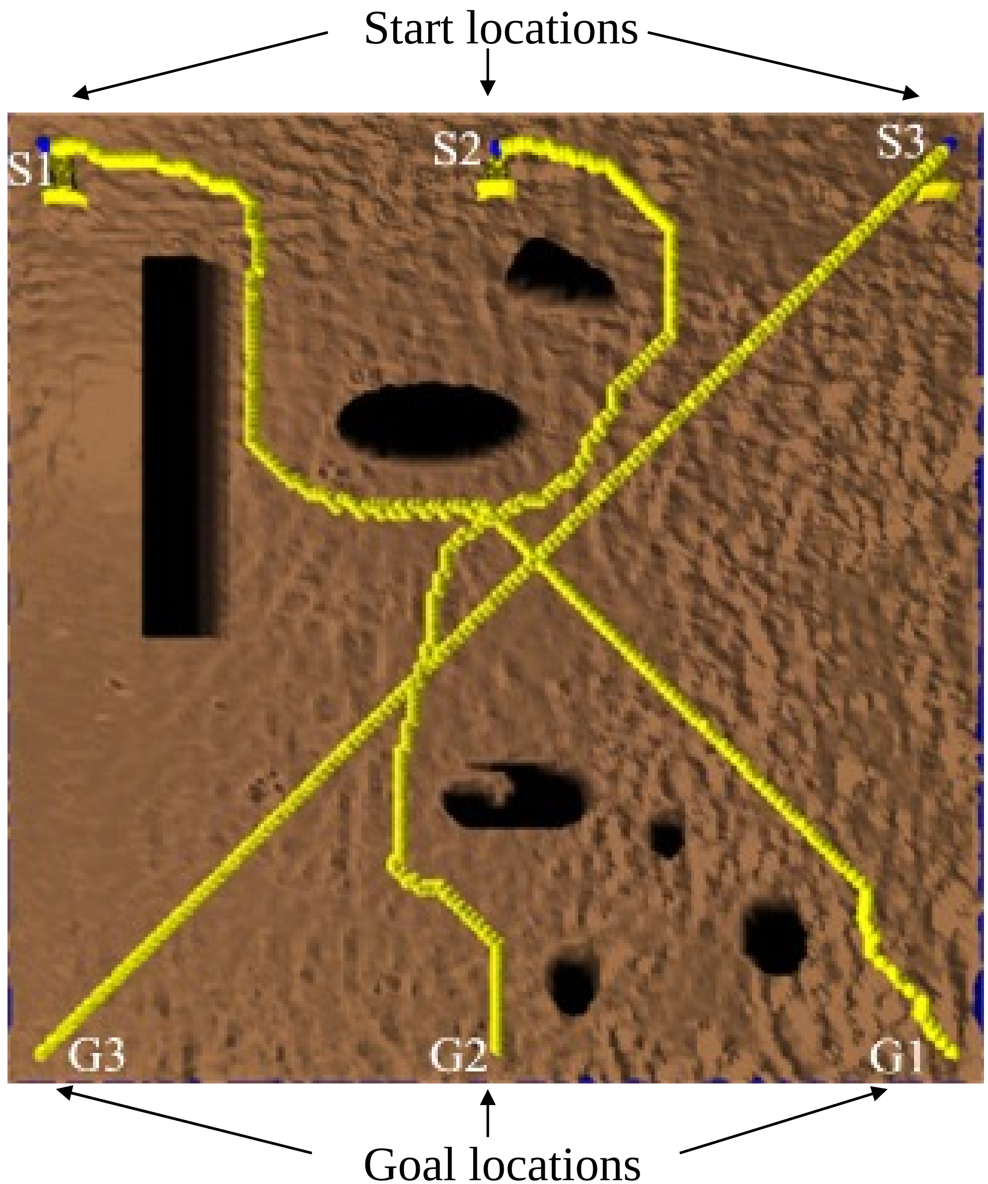
Cost function:

$f_{pp} = \mathrm{r} + \mathrm{a}_1 d + \mathrm{a}_2 s + \mathrm{a}_3 t$

d is distance, s is slope,

t is penalty on turning,

r is penalty on obstacles.

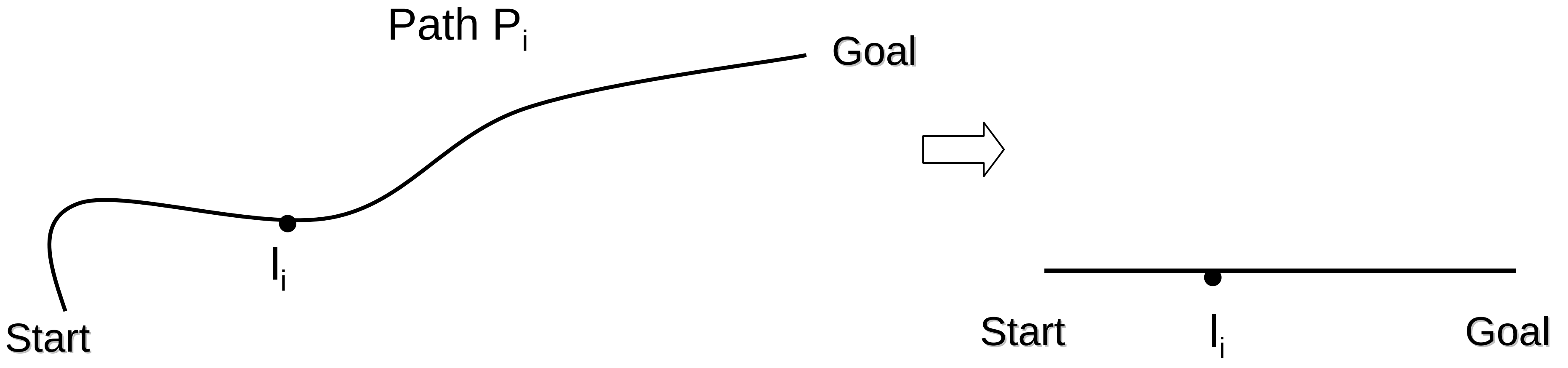


Step 2: Path is broadcast across robots; collision (time-space) check produces a set of collision regions;

Step 3: Coordination diagram constructed:

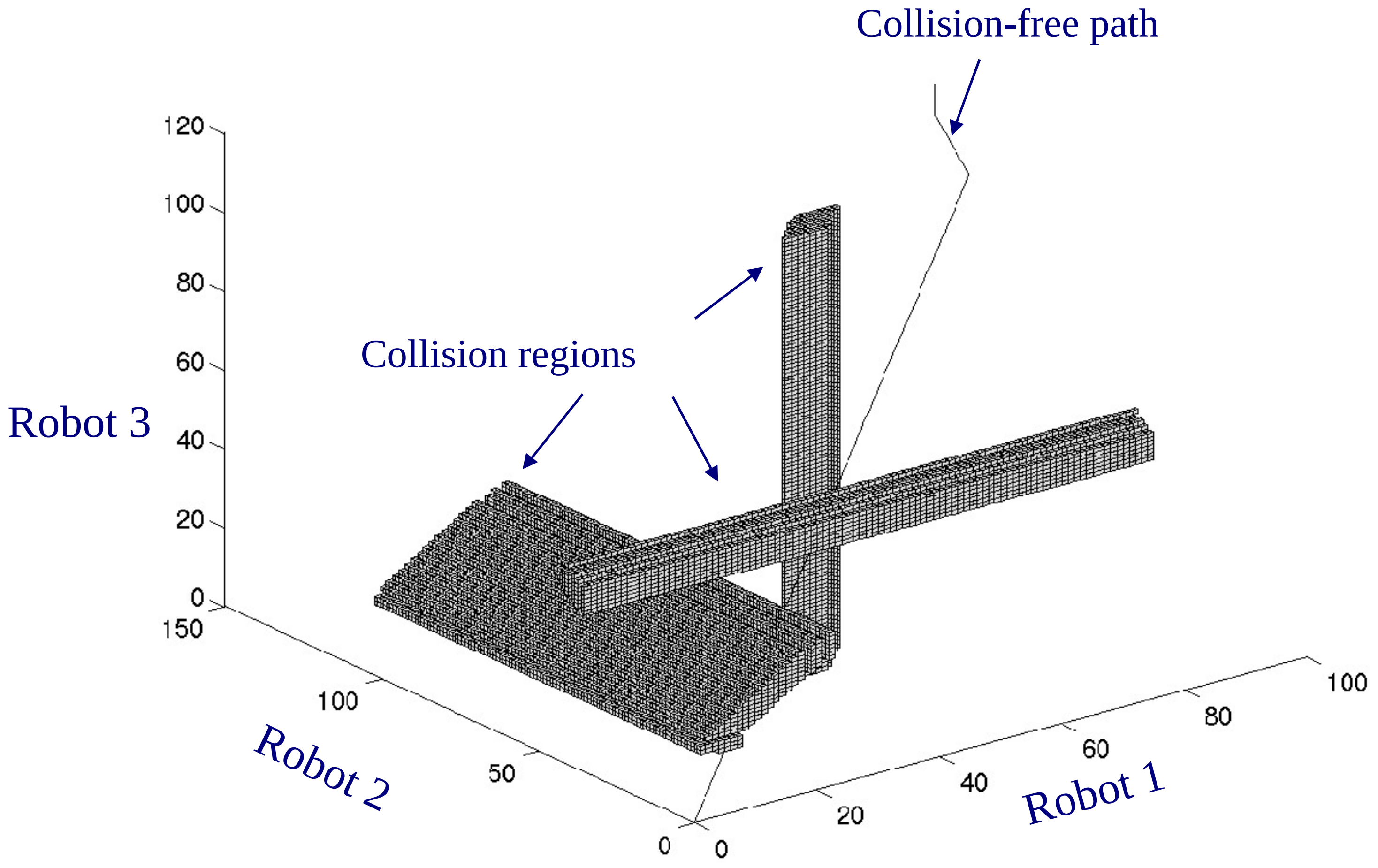
- Each path P_i is a continuous mapping $[0,l] \rightarrow W^{e}_{free}$;
- $S_i = [0,l]$ denotes the set of points that place robot along path P_i;
- Coordination space is defined $S = S_1 \times S_2 \times \cdots \times S_N$
- Collision regions marked as obstacles in coordination diagram.

Path Parameterization



Mapping each path P_i in W^{e}_{free} to a one-dimensional trajectory taking path length as parameter

Coordination Diagram (CD)



Step 4: Velocity planning:

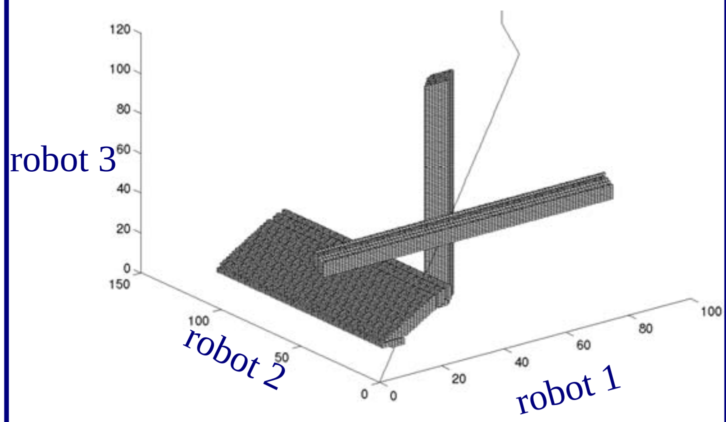
- D* search in coordination diagram, minimizing cost function:

$$f_{vp} = \text{d} + \text{a}_1 d + \text{a}_2 t_{idle} + \text{a}_3 p$$

where: d is distance, t_{idle} is idle time,

p is penalty for giving way

(different on each robot), d is penalty on obstacles;



Coordination Diagram

and producing velocity profile VP_i and performance index K_i (each robot evaluates a set of costs);

Velocity Profile

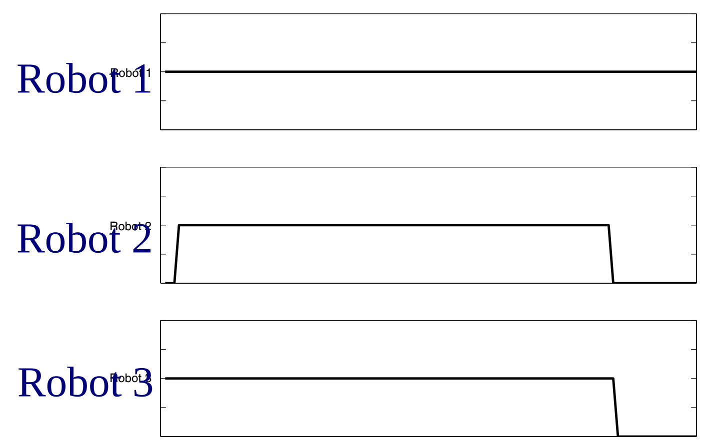
- Cost function for D* search in CD:

$$f_{vp} = \text{d} + \text{a}_1 d + \text{a}_2 t_{idle}$$

where: d is distance, t_{idle} is idle time,

d is penalty on obstacles;

- The returned path maps to velocity profile



Velocity over time

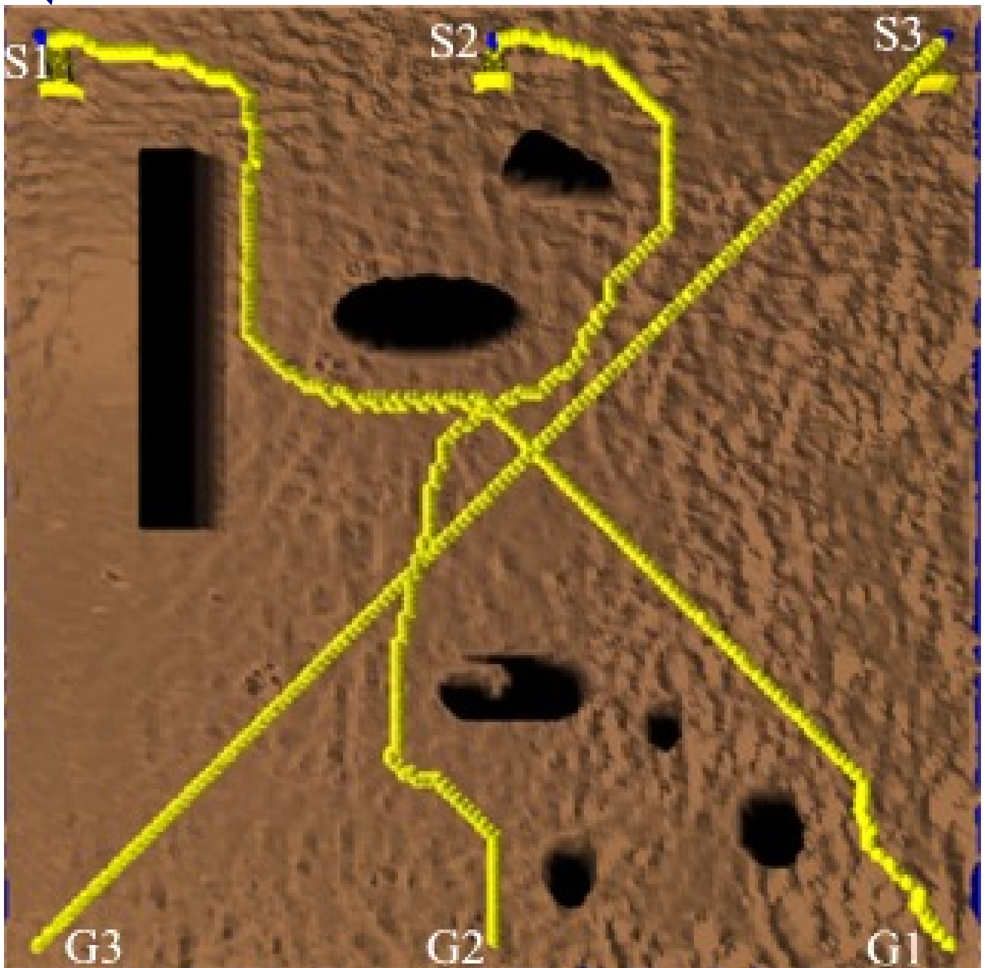
Step 5: Broadcasting VP_i and K_i across robots;

Step 6: Global performance evaluation

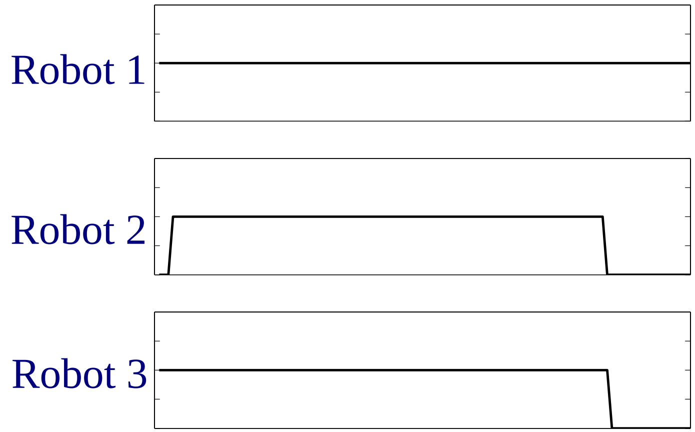
- Find the minimal K_l, select corresponding VP_l as the optimal solution for velocity.

Implementation: 3D Simulation

Start locations



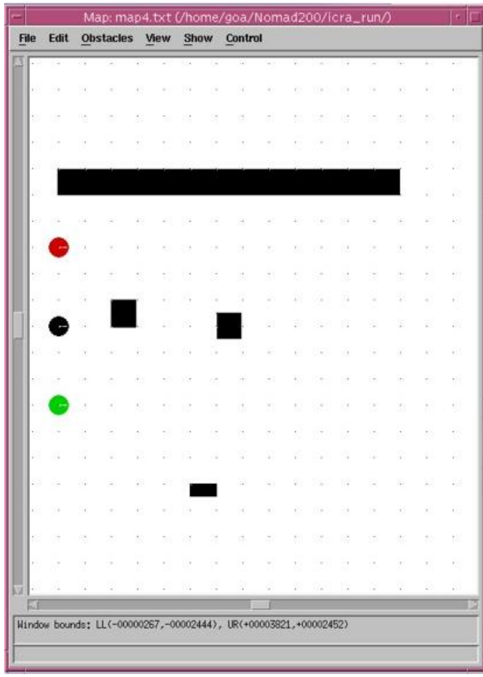
Goal locations



Velocity profile

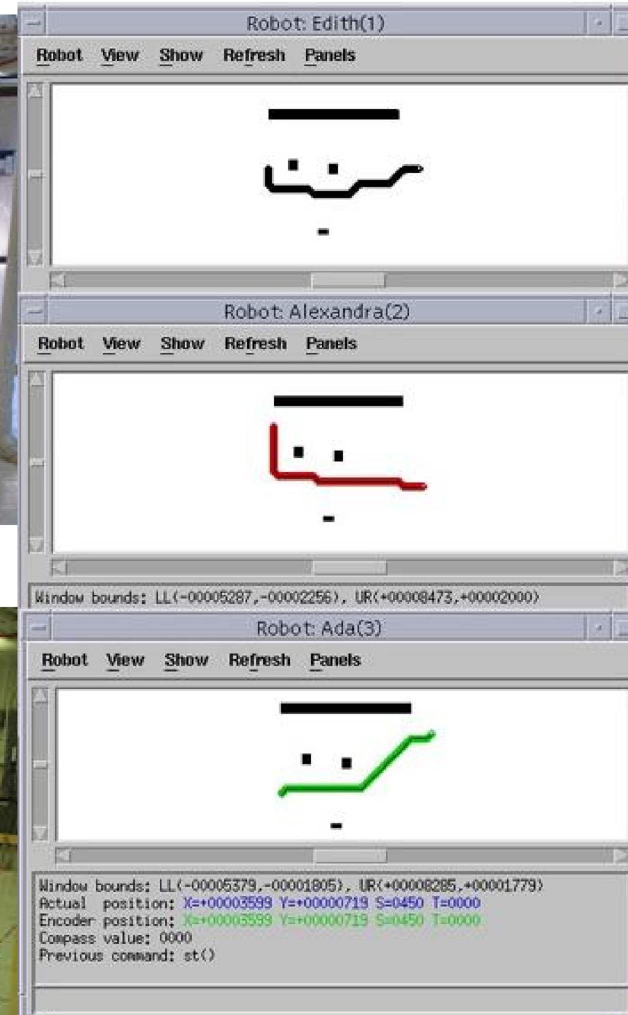
Multiple paths in Mars-like terrain environment

Nomad 200 Experiments

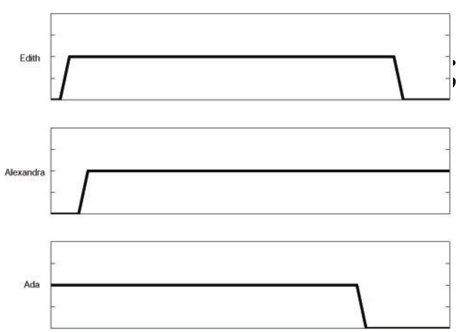


Environmental map

Robots at start positions



Encoder trajectory records



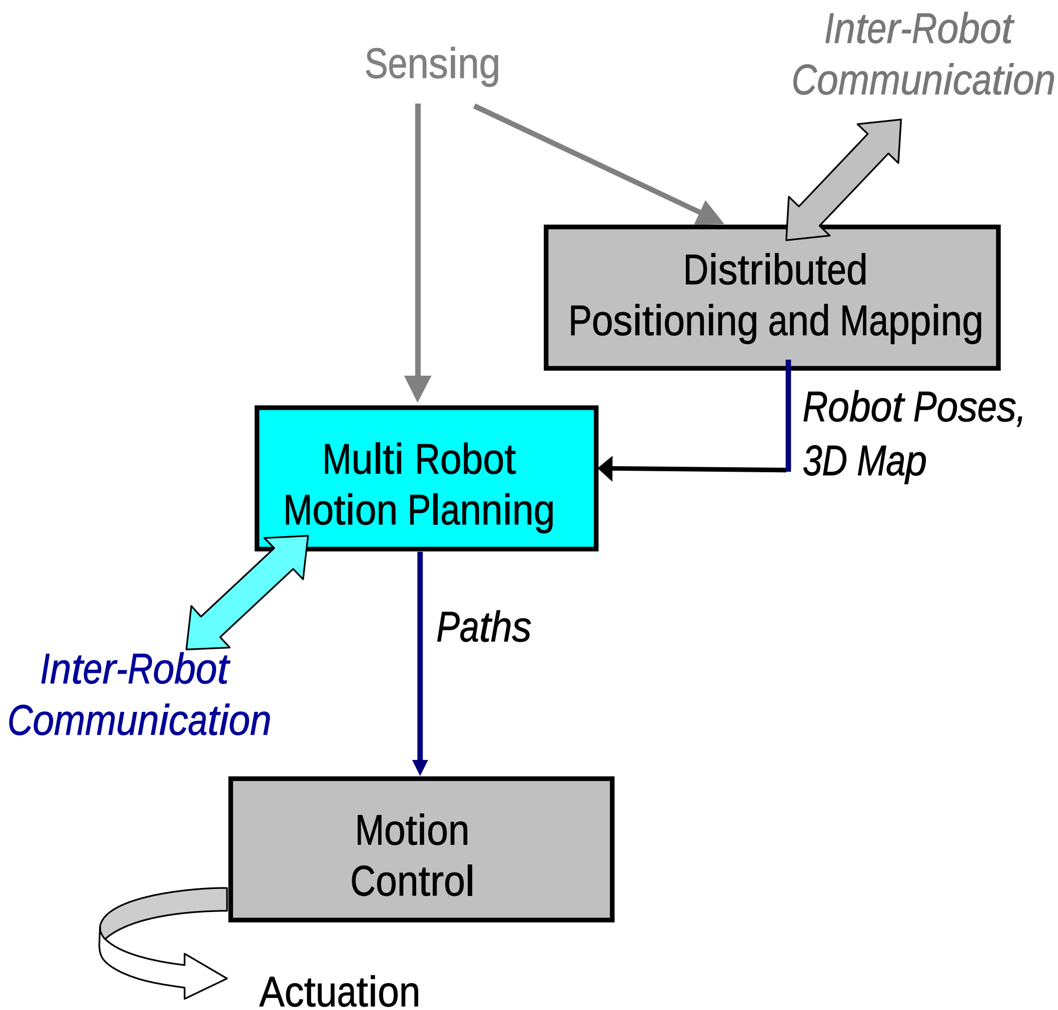
Velocity profile

Robots in motion

Issues in Design Robustness

- Observations:
  - Localization errors;
  - Motion uncertainties:
    - Robot does not take equal unit time to track a unit distance;
    - Robot does not switch instantaneously between moving and stopping.
- Robustness design:
  - Safety margin defined in path searching for localization errors;
  - Safety margin defined in velocity planning for motion uncertainties.

ATRV Experiments

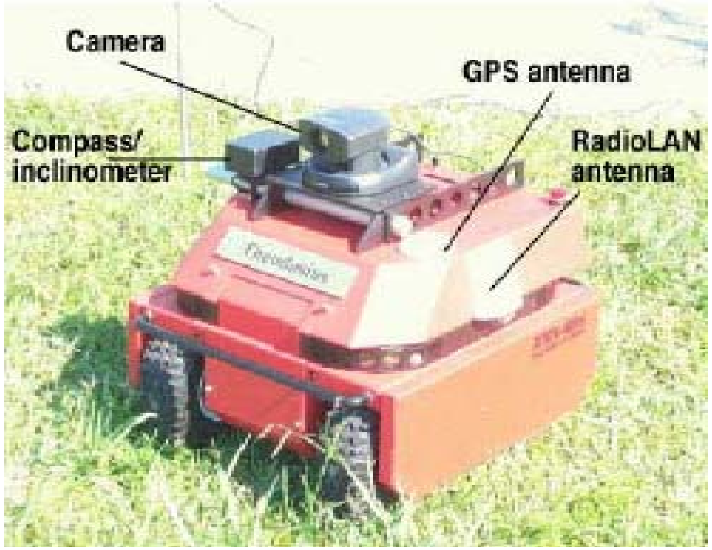


System integration issues:

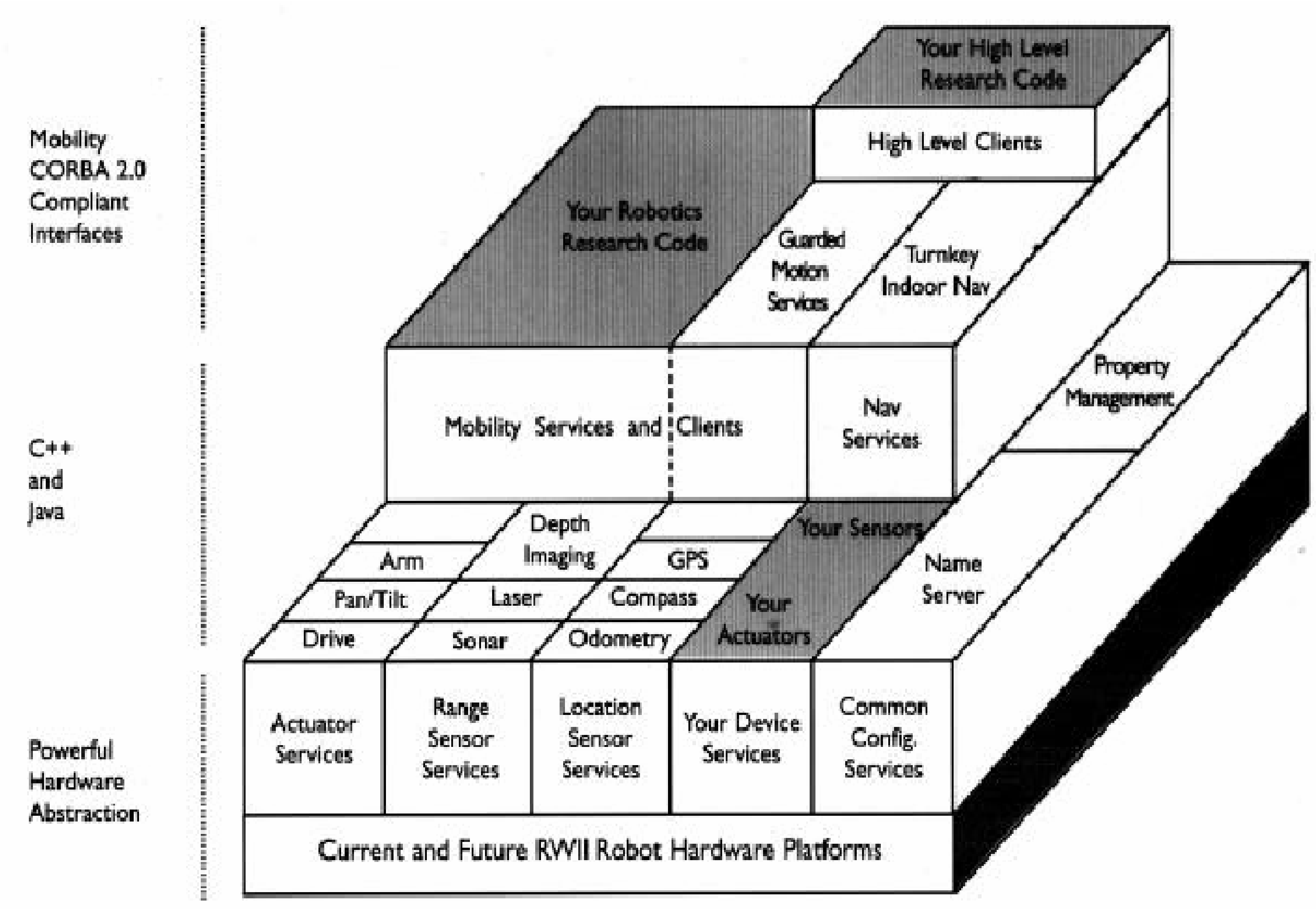
- Sensor selection and fusion
- Software platform
- Communications
- GUI/simulator

Heterogeneous sensor suites

- DGPS: upto 10-15cm accuracy; 1Hz
- PTZ CCD Camera: 120X160 pixels, 37°FOV; 10Hz
- Laser range scanner: 180° scan, 0.5° resolution, within 15m; 10Hz
- Sonar: measure distance to obstacle within 4m
- Compass: measure yaw angle
- Inclinometer: measure pitch and roll angles
- Encoder: internal measure of vehicle position; 50Hz

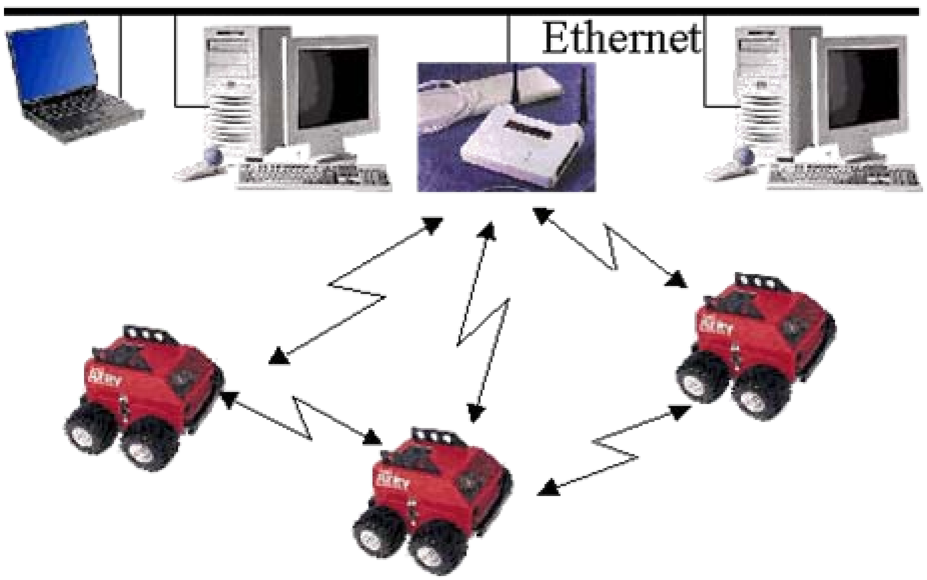


Software Platform on ATRV

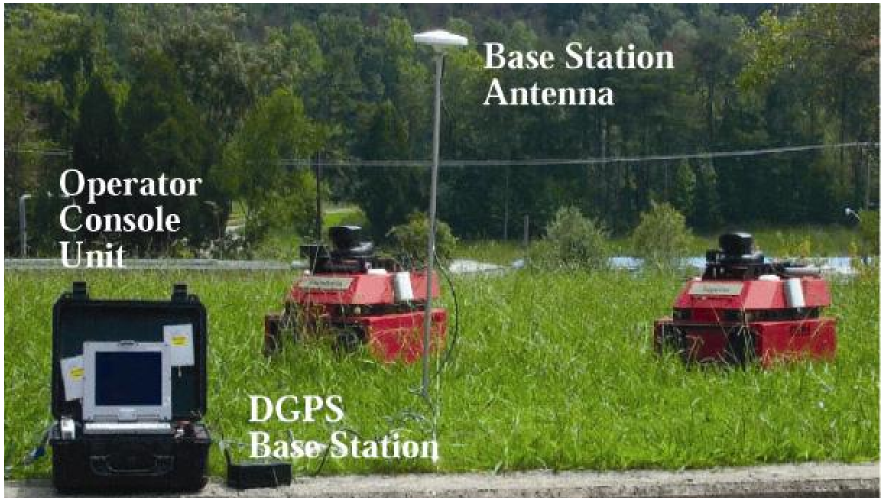


Robot integration software (Mobility) environment in context

Communications



Multipoint networking configuration



Wireless LAN setup outdoors

Summary

- A 3D multi-robot motion planning algorithm is presented that is distributed, optimal, and capable of real time re-planning in outdoor environment.
- The computationally expensive problem is decomposed into two modules: path planning and velocity planning.
- D* search method is applied in both modules, based on either geometric or schedule formulations.
- Robustness design is incorporated in the algorithm to overcome motion and sensor uncertainties.
- Implementation issues and experiments on ATRV-mini robots are discussed.

Reading

- Y. Guo and L. E. Parker, “A distributed and optimal motion planning approach for multiple mobile robots”, *Proceedings of the 2002 IEEE International Conference on Robotics and Automation (ICRA’02)*, pp. 2612-2619, Washington D.C., May, 2002.