CAP6671 Intelligent Systems

Lecture 16:

Multi-robot Coordination (part 2)

Instructor: Dr. Gita Sukthankar
Email: gitars@eecs.ucf.edu
Schedule: T & Th 9:00-10:15am
Location: HEC 302
Office Hours (in HEC 232):
T & Th 10:30am-12

Relevant subproblems?

Relevant subproblems?

- Task allocation
- Scheduling
- Information/map fusion
- Path planning

General Approaches

- Heartbeat approach (Gerkey & Mataric)
- Social potential fields (Balch & Arkin)
- Stigmergic approach (Shen)
- Locker-room agreement (Stone)
- Token-based architecture (Yu, Scerri, Sycara)
- Team-oriented planning (Tambe)
- Decentralized MDP (Lesser)

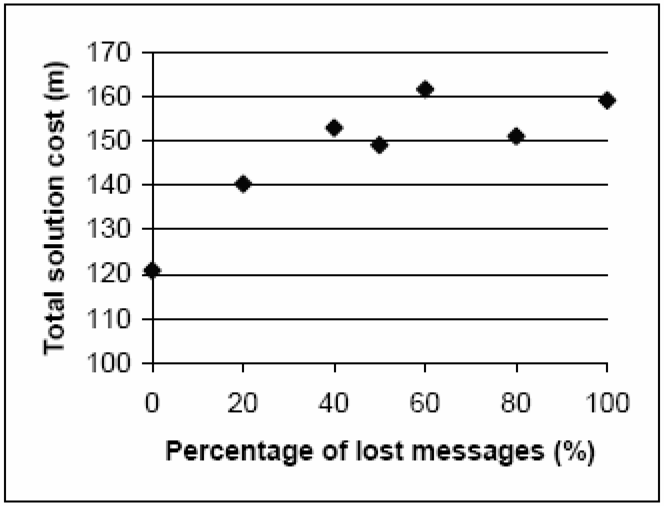
Market-based Approaches

- Formulate the robot task allocation problem as a group auction
- Robots bid their cost for completing a task
- Central auctioneer injects tasks into the market
- Robots can re-auction tasks to other robots
- Market can be made increasingly robust through:
  - Distributed auctions
  - Monitoring awarded tasks and reauctioning them if not completed within a certain period of time

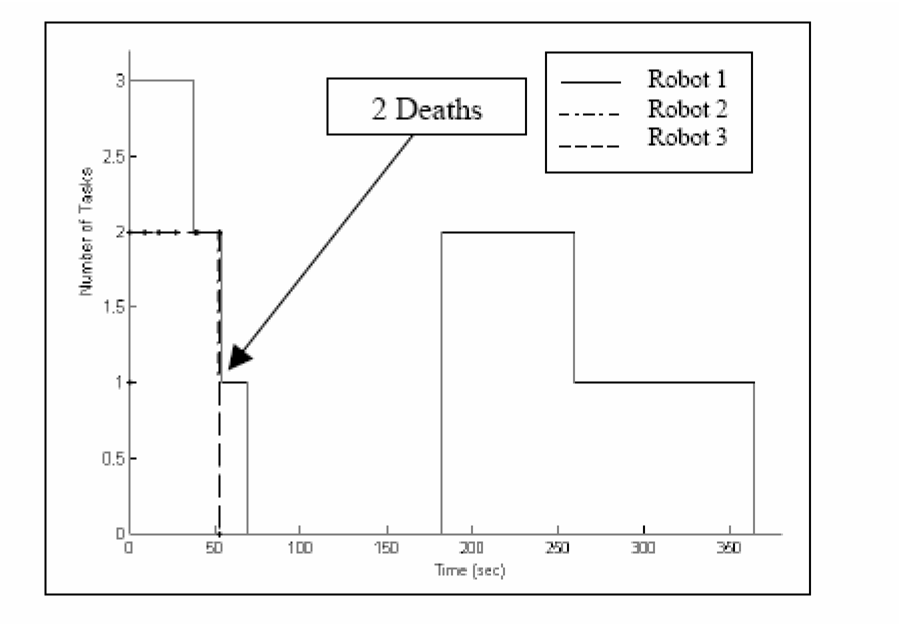
TraderBots (Dias, Stentz, et al)

- Demonstrated system robust to
    - simulated message loss
    - partial robot failures
    - robot death
    - intermittent robot function

Communication Loss



Robot Death


2 Deaths
Robot 1
Robot 2
Robot 3
Number of Tasks
Time (sec)
0
0.5
1
1.5
2
2.5
3
0
50
100
150
200
250
300
350

Problems?

COCOA

- Constraint Optimization Coordination Architecture
- Capable of reasoning about joint goals
- Outperforms token-based coordination and market-based architecture under certain conditions
- Problem formulation is similar to deliberate team planning framework but solution method uses constraint optimization

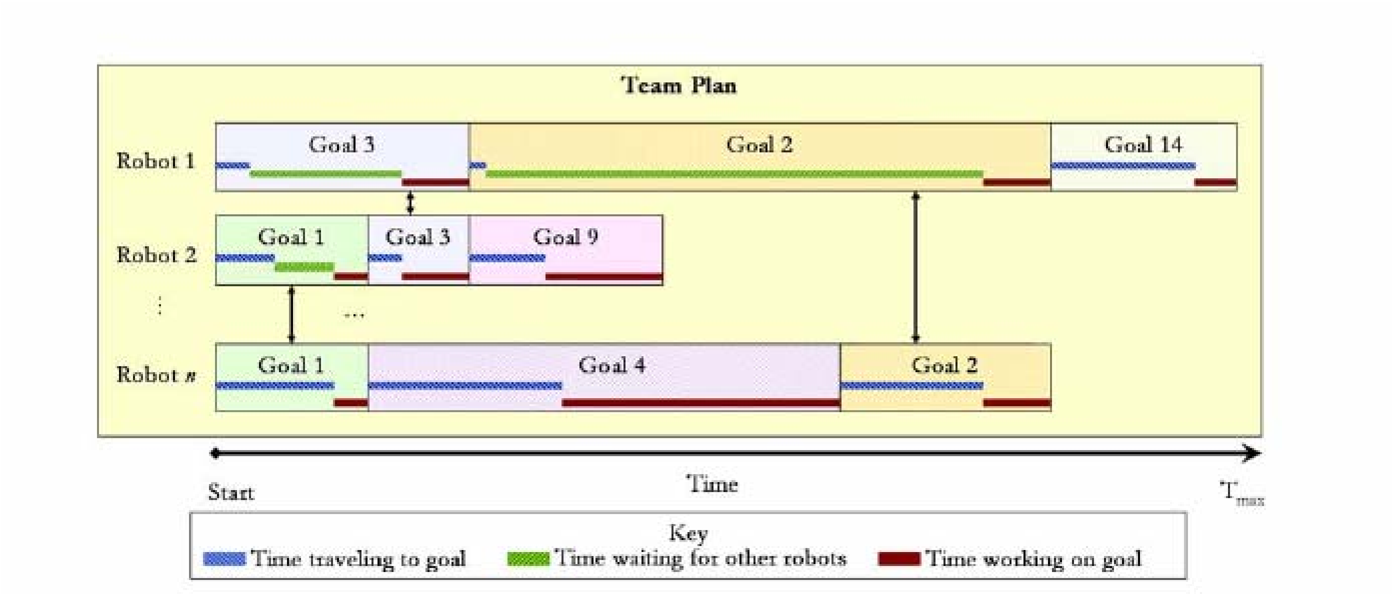
MILP

- Stands for **M**ixed-**i**nteger **L**inear **P**rogramming
- Linear programming
  - Optimizing a linear objective function subject to linear equality and inequality constraints
  - Used to pose questions like “how can I mazimize my utility subject to certain constraints”
  - Simplex method
- Mixed-integer linear programming
  - Certain variables are required to have integer values (e.g. number of robots)
  - Solved by relaxing constraint that variables be integer

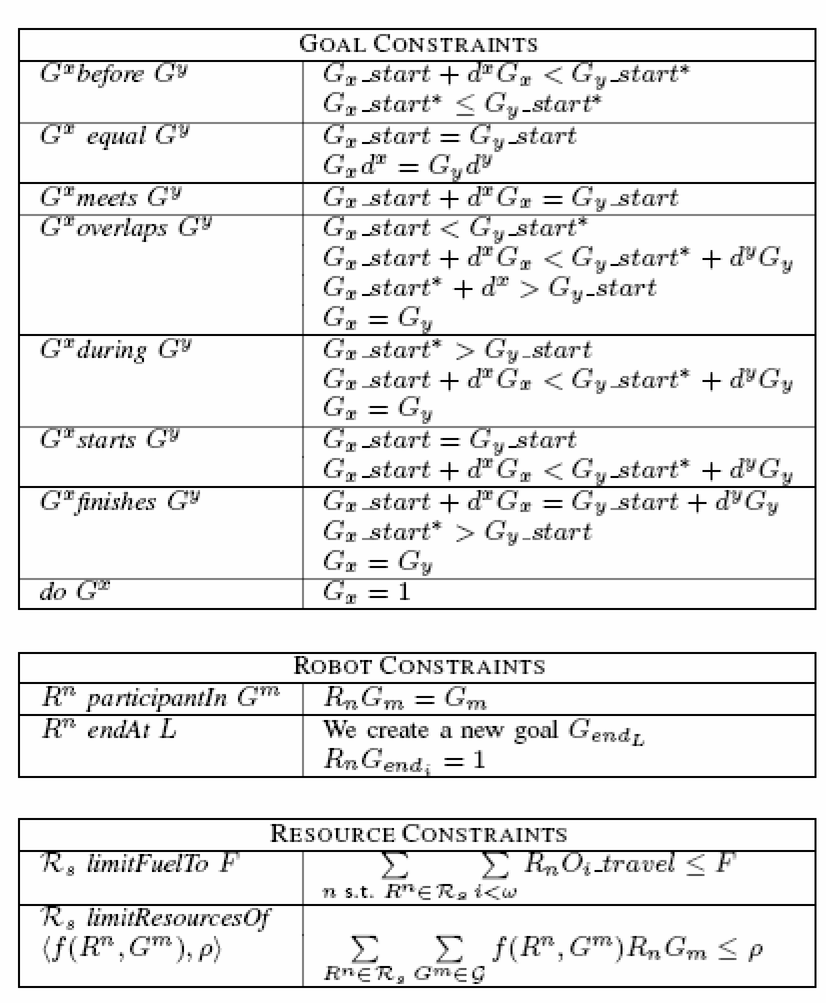
Constraints

- Goal constraints:
  - Allen’s 13 temporal relationships (before, equal, meets)
- Robot constraints
  - ParticipateIn (robot must be part of mission)
  - EndAt (robot must end at a certain location)
- Resource constraints
  - LimitFuelTo
  - LimitResourceTo
- Constraints can be combined using 1st order logic and converted to numeric constraints using binary constraint variables

Example Plan



Constraint Formulation



GOAL CONSTRAINTS	
G^x*before* G^y	$G_x\_start + d^x G_x < G_y\_start^*$ $G_x\_start^* \leq G_y\_start^*$
G^x *equal* G^y	$G_x\_start = G_y\_start$ $G_x d^x = G_y d^y$
G^x*meets* G^y	$G_x\_start + d^x G_x = G_y\_start$
G^x*overlaps* G^y	$G_x\_start < G_y\_start^*$ $G_x\_start + d^x G_x < G_y\_start^* + d^y G_y$ $G_x\_start^* + d^x > G_y\_start$ $G_x = G_y$
G^x*during* G^y	$G_x\_start^* > G_y\_start$ $G_x\_start + d^x G_x < G_y\_start^* + d^y G_y$ $G_x = G_y$
G^x*starts* G^y	$G_x\_start = G_y\_start$ $G_x\_start + d^x G_x < G_y\_start^* + d^y G_y$
G^x*finishes* G^y	$G_x\_start + d^x G_x = G_y\_start + d^y G_y$ $G_x\_start^* > G_y\_start$ $G_x = G_y$
do G^x	$G_x = 1$

ROBOT CONSTRAINTS	
R^n *participantIn* G^m	$R_n G_m = G_m$
R^n *endAt* L	We create a new goal G_{end_L} $R_n G_{end_i} = 1$

RESOURCE CONSTRAINTS	
$\mathcal{R}_s$ *limitFuelTo* F	$\sum_{n \text{ s.t. } R^n \in \mathcal{R}_s} \sum_{i<\omega} R_n O_i\_travel \leq F$
$\mathcal{R}_s$ *limitResourcesOf* $\langle f(R^n, G^m), \rho \rangle$	$\sum_{R^n \in \mathcal{R}_s} \sum_{G^m \in \mathcal{G}} f(R^n, G^m) R_n G_m \leq \rho$

Making MILP Real-time

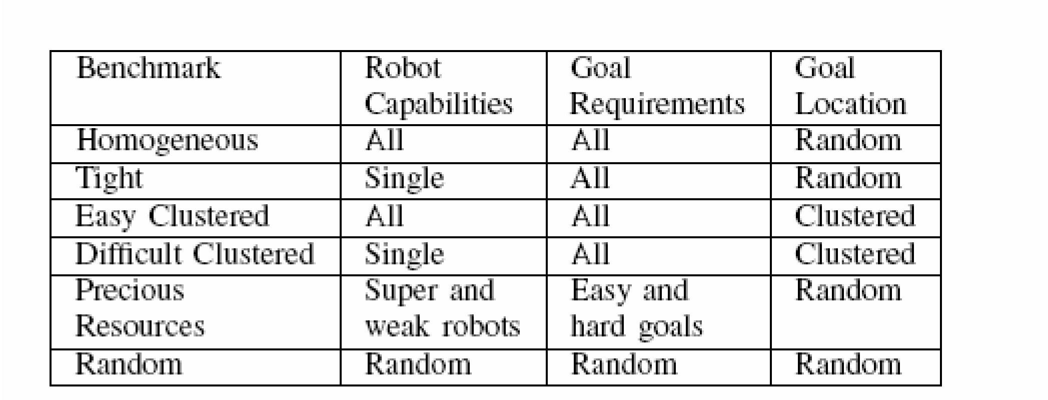
- Generate solution using heuristic and use it as a starting point for MILP solver
- Use solutions for subsets of goals to build up to full solution
- Use solutions for shorter time-horizons to build up to complete time horizon

Heuristics

- Myopic
  - Robots recursively schedule goals over an increasing planning horizon
  - Problem size grows exponentially with planning horizon so subproblem is much smaller than original problem
- Greedy goal
  - Similar to market-based task allocation
  - Sort goals in decreasing order of reward
  - Robots bid cost
  - Lowest bidder assigned to goal capability
  - No goal reauctioning

Evaluation

Benchmark Problems

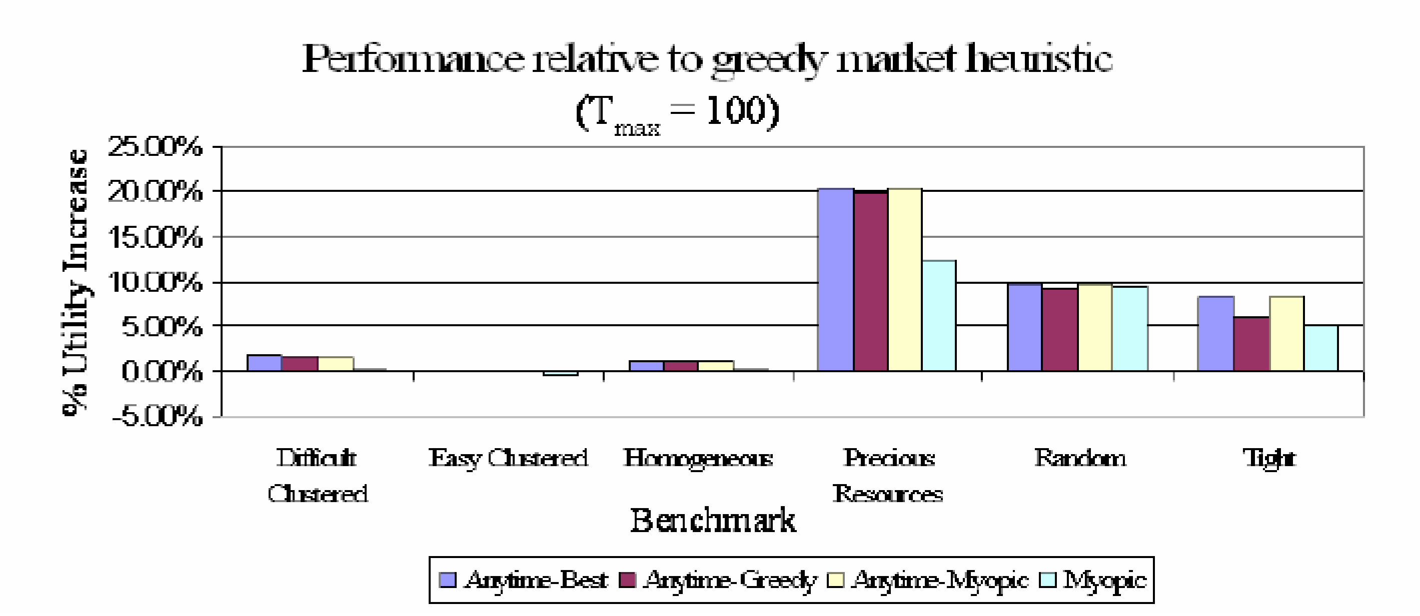


Benchmark	Robot Capabilities	Goal Requirements	Goal Location
Homogeneous	All	All	Random
Tight	Single	All	Random
Easy Clustered	All	All	Clustered
Difficult Clustered	Single	All	Clustered
Precious Resources	Super and weak robots	Easy and hard goals	Random
Random	Random	Random	Random

- Compared myopic heuristic, greedy heuristic, anytime algorithm with myopic, anytime with greedy, anytime with best

Results

- Showed significant performance improvements over the greedy algorithm (which is comparable to a market-based strategy)



Problems?