Team Task Allocation and Routing under Human Guidance

David Castañón

Darin Hitchings

Boston University

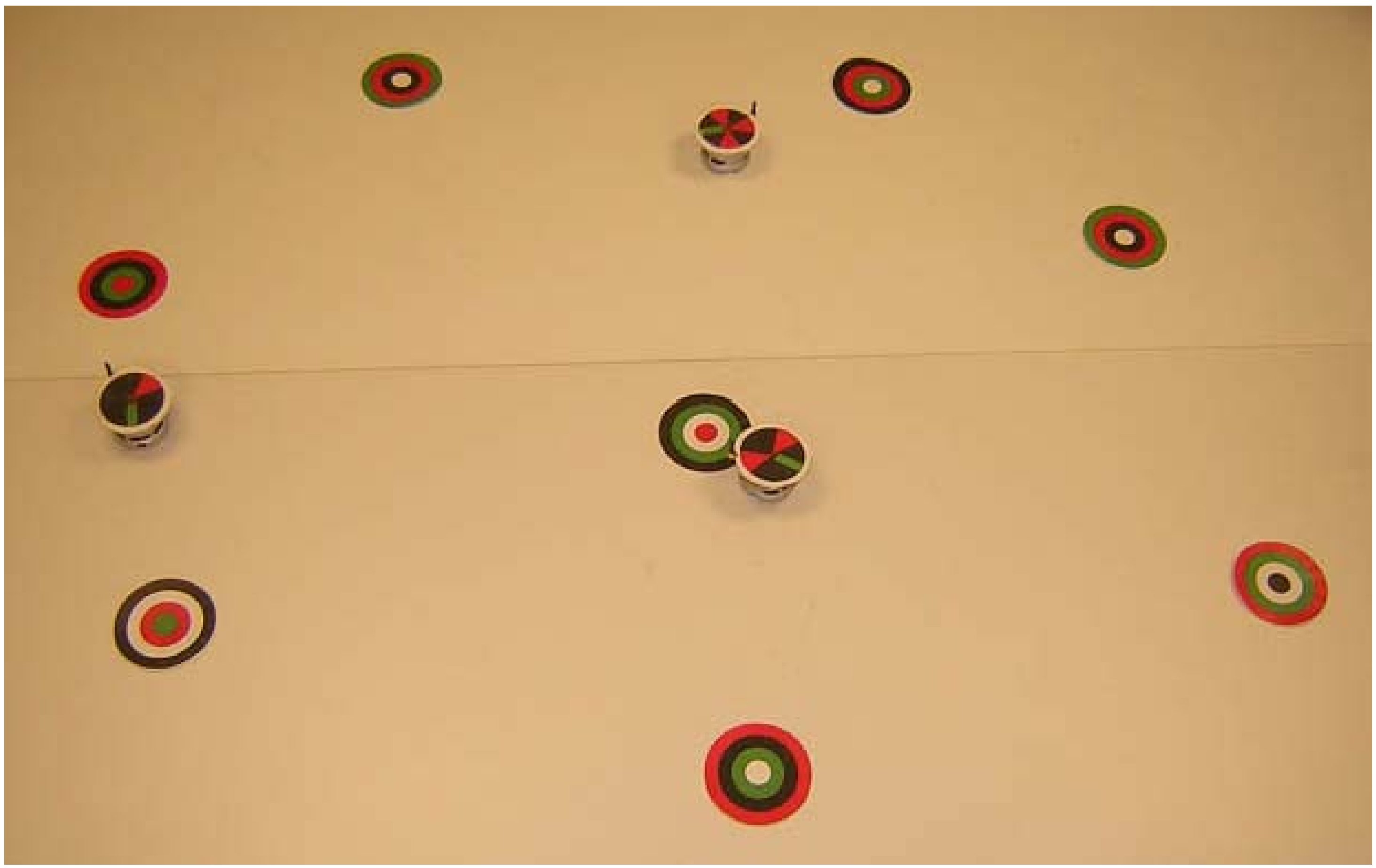
Center for Information and Systems Engineering

Introduction

- Objective: Study relative advantages of alternative human control approaches in problems involving teams of autonomous vehicles
- Paradigm: teams execute diverse spatially distributed tasks in uncertain environments
  - Uncertain nature, number of tasks
  - Risk of vehicle loss
- Combine aspects of exploration and exploitation
  - Must trade off searching for potential tasks versus exploiting known tasks
- Focus: Develop vehicle control algorithms under varying levels of control

Experiment Facility

- Multiple robots search for and perform tasks at BU's Mechatronics Lab



Initial Problem: Task Allocation

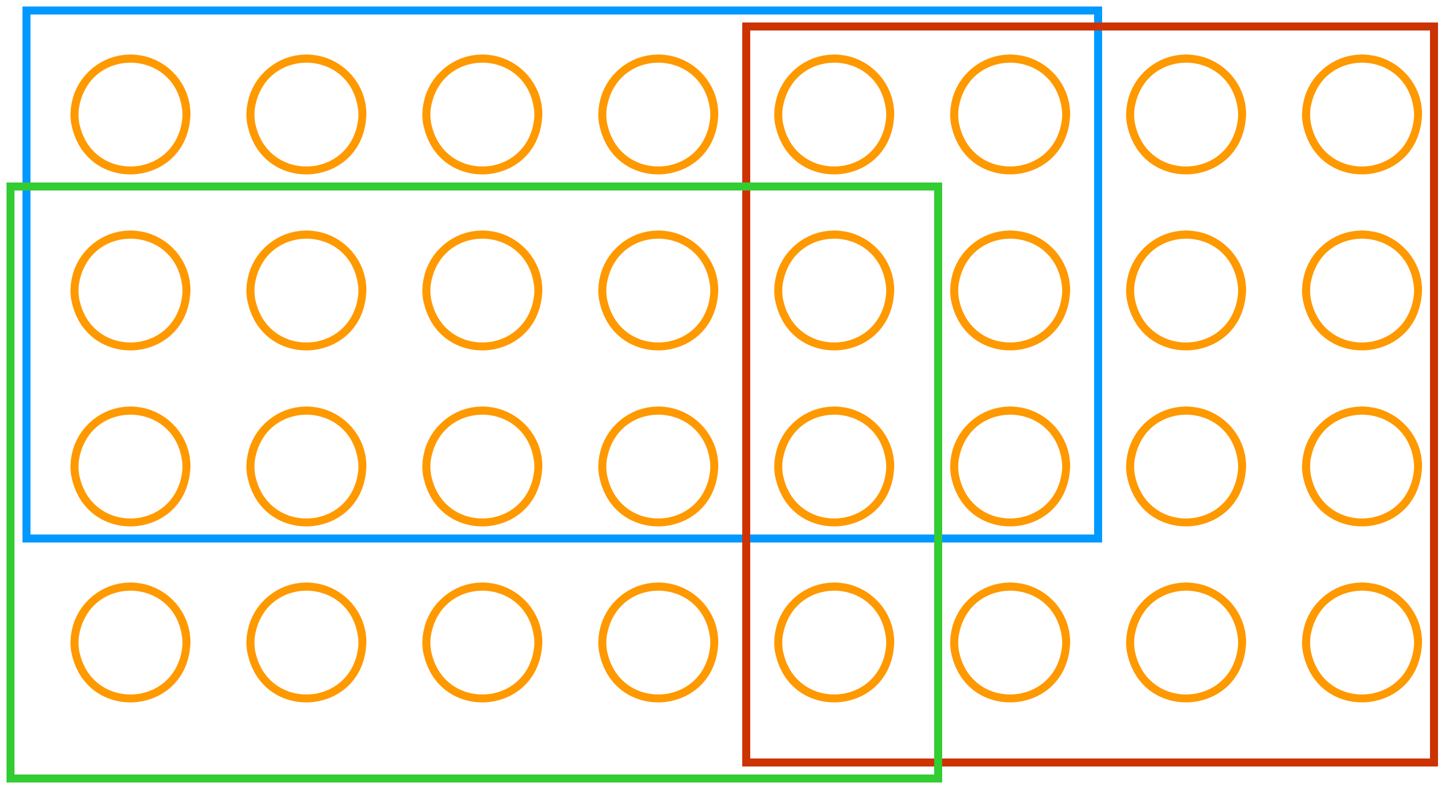
- Problem paradigm: Find and correctly classify objects in field of interest
  - Finite number of areas that may contain objects
  - Multiple actions possible per area
    - Obtain different quality of information: search, image at different resolutions
    - Quality of action increases with time used in action
  - Multiple agents in team, with overlapping fields of regard
- Objective: adaptive scheduling of team activities to find and correctly classify objects of interest
  - Team member action: select area and mode to observe, collect and communicate information to rest of team members
  - Trade off search for new objects versus obtaining high quality information on known objects
  - No risk of platform loss

Nature of Team Decision Problem

- Control information dynamics
  - Control flow of information on objects by selecting actions
  - Process information in Bayesian setting using statistical models
  - Dynamics: Bayesian inference
- Sequential decision problem: select next actions based on collected information
- Objective: Bayes classification cost
  - After fixed amount of sensing resource, minimize expected classification error cost (terminal cost only)
  - Related to Cohen-Holmes inferencing paradigm, but without time penalty
    - Some differences: multiple actions, potentially multiple classes of objects, search

Illustration of Problem

- 3 Agents with different fields of regard (different colors)
- Multiple sites to search and classify objects
- Initial focus: no motion (static field of regard, sites)



Mathematical Representation

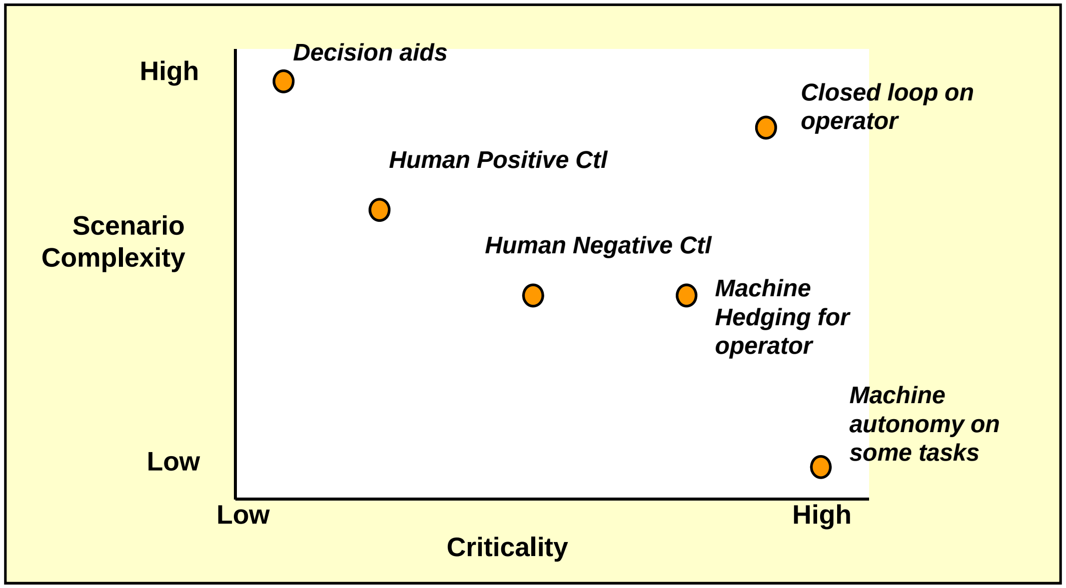
- N sites, each possibly containing an object with S possible types
- Underlying state at each site: x_i in {0, …, S} where 0 is empty
- Information state at site n: probability of site content π_n
- Multiple agents K, M observation modes per agent
- Mode m from sensor k on site i requires R_{ikm} time
- Decisions: $u_{ikm}(t) = 1$: mode m, agent k to site i
  - Consumes resource R_{ikm}
- Finite total observation resource per agent k: C_k
- Finite-valued observation y_{jkm} for site i:
  - Likelihood $P(y_{ikm}|x_i, u_{ikm})$ known
- Assumption: Conditional independence of observations across agents, time, modes

Variations on Human Control

1) Control by objective
   - Provide Bayes' objective in terms of cost of classification errors
   - Agent control algorithms seek to minimize expected Bayes cost
2) Control by geographical partitioning and local objectives
   - Partition site responsibility among agents, adapting site allocation in response to progress and workload
3) Control by functional partitioning:
   - Assign specific functions (modes of observation) to agents
4) Control by action: Select activities of agents adaptively based on observations

AFRL Notional Diagram for Alternative Human Control

- Murphey 2007



System-Level Performance Measures

- Success:
  - Target Value Destroyed
  - False Target Rate, …
- Clarity of intent: human, machine
- Quality of decisions: Workload Margin
- Stability → Criticality
- Flexibility/robustness → Scenario complexity

…

$$\textit{Criticality} \sim \frac{\textit{Cost of wrong decision}}{\textit{System/decision time constant}}$$

Autonomous Control Algorithms: Theory Overview

- Theorem: Under given assumptions, a sufficient statistic is $\Pi(t) = \{\pi_1(t), \ldots, \pi_N(t)\}$, where $\pi_i \in S_k$ is conditional probability of site i's content given past information measured on site i only
  - NOTE:→ Joint conditional probability is product of marginals
- Information Dynamics (discrete event system): Bayes' Rule
  - Act locally on objects: only measured sotes change information state
  - Similar to multi-armed bandit problem

$$\begin{aligned} \pi_i^s(\tau+1) &\equiv P(x_i = s | Y_i(\tau+1)) \\ &= \frac{\pi_i^s(\tau) \prod_{k,m} P(y_{ikm} | x_i = s, u_{ikm}(\tau))}{\sum_{s'} \pi_i^{s'}(\tau) \prod_{k,m} P(y_{ikm} | x_i = s', u_{ikm}(\tau))} \end{aligned}$$

Resource Constraints

- Constraints: *for all observation sample paths*
  - Cannot exceed total sensor resource

$$\sum_{\tau=0}^{T-1} \sum_{i=1}^{N} \sum_{m=1}^{M} R_{ikm} u_{ikm}(\tau) \leq C_k \text{ for all } k \in K$$

  - Lots of these: one constraint per sample path

  - Only one action per sensor at each event time

$$\sum_{i=1}^{N} \sum_{m=1}^{M} u_{ikm}(\tau) \leq 1 \quad \text{(sensor timeline constraint)}$$

Objective

- Goal: accurate classification with given resources
  - Cost: Minimize expected Bayes classification error as a final action at random stopping time T
    - Classification decision for object i: $v_i(T)$

$$J = \sum_{i=1}^{N} E\{\min_{v_i} c(x_i(T), v_i(T))\}$$

- Result: Partially Observed Markov Decision Problem (POMDP) with sample path constraints (product state space)
  - Extension of classical POMDP (Smallwood-Sondik, …) with constraint states
  - Solvable by DP recursion
  - Too cumbersome!

Approximate Control Algorithm

- Relax sensor resource constraints to average value:

$$\sum_{\tau=0}^{T} \sum_{i=1}^{N} \sum_{m=1}^{M} E\{R_{ikm} u_{ikm}(\tau)\} \leq C_k$$

  - Single constraint per sensor, averaged across sample paths
  - Chen-Blankenship model
  - Expands admissible strategies, *yields lower bound*
- Allow each sensor to act on multiple objects per event time

$$\sum_{m=1}^{M} u_{ikm}(\tau) \leq 1$$

- Allow for mixed strategies
  - Simplifies the integer programming nature of the relaxed problem
  - Convexifies problem and *maintains lower bound*

Lower Bound POMDP

- Minimize $$J = \sum_{i=1}^{N} E\{\min_{v_i} c(x_i(T), v_i(T))\}$$

- Subject to constraints

$$\sum_{\tau=0}^{T} \sum_{i=1}^{N} \sum_{m=1}^{M} E\{R_{ikm} u_{ikm}(\tau)\} \leq C_k$$

$$\sum_{m=1}^{M} u_{ikm}(\tau) \leq 1$$

$$\pi_i^s(\tau+1) \quad = \quad \frac{\pi_i^s(\tau) \prod_m P(y_{ikm} | x_i = s, u_{ikm}(\tau))}{\sum_{s'} \pi_i^{s'}(\tau) \prod_m P(y_{ikm} | x_i = s', u_{ikm}(\tau))}$$

$$u_{ikm}(\tau) : [\pi_1(\tau) \ldots \pi_N(\tau)] \rightarrow \{0, 1, \ldots, M\}$$

Weak Duality

- Use Lagrange multipliers to incorporate relaxed resource constraints into objective: Lagrangian, for λ >= 0:

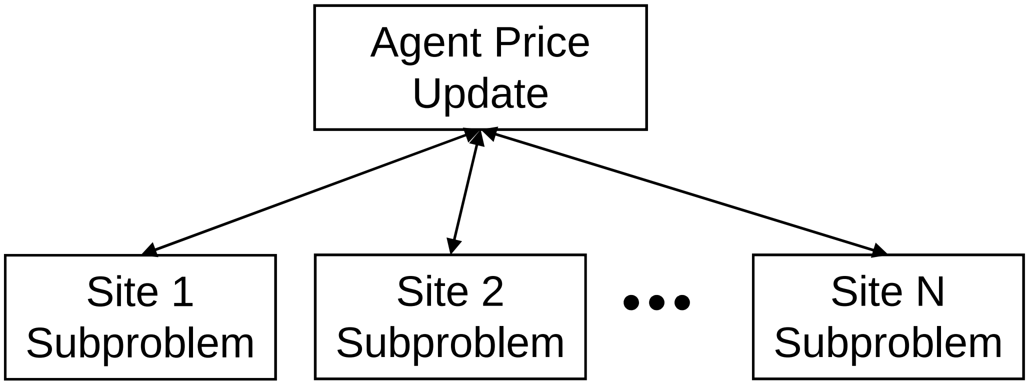
$$J(\lambda,\gamma) = E_\gamma\{\sum_{i=1}^{N}[c(v_i,x_i)+\sum_k \lambda_k \sum_{\tau=0}^{T-1}\sum_{m=1}^{M} R_{ikm}u_{ikm}(\tau))\}-\sum_k \lambda_k C_k$$

- Lower bounds given by weak duality

$$\min_\gamma J(\lambda,\gamma) \le \max_{\lambda\ge 0}\min_\gamma J(\lambda,\gamma) \le \min_\gamma J(\gamma)$$

- Lagrangian problem is almost separable over objects
  - Coupled only by feedback strategies!
  - THEOREM: Can decouple bound computation across objects given dual variables
  - For every coupled strategy, there is an equivalent random decoupled strategy that achieves the same performance

Hierarchical Pricing of Agent Time



$$\min_p L(p,\lambda) = \sum_i \min_{p_i} p_i(\gamma_i)(J_i^{\gamma_i} - \sum_j \lambda_j R_{ij}^{\gamma_i}) + \sum_j C_j \lambda_j$$

Note: minimum is achieved in pure strategies for each price vector λ

- Agent prices: dual variables for consuming sensor time for different sensors
  - Subproblems solved optimally using small POMDP single object algorithms
  - NS-dimensional POMDP reduced to N single object S-dimensional POMDPs + dual

Extension of Algorithms for Different Human Control Approaches

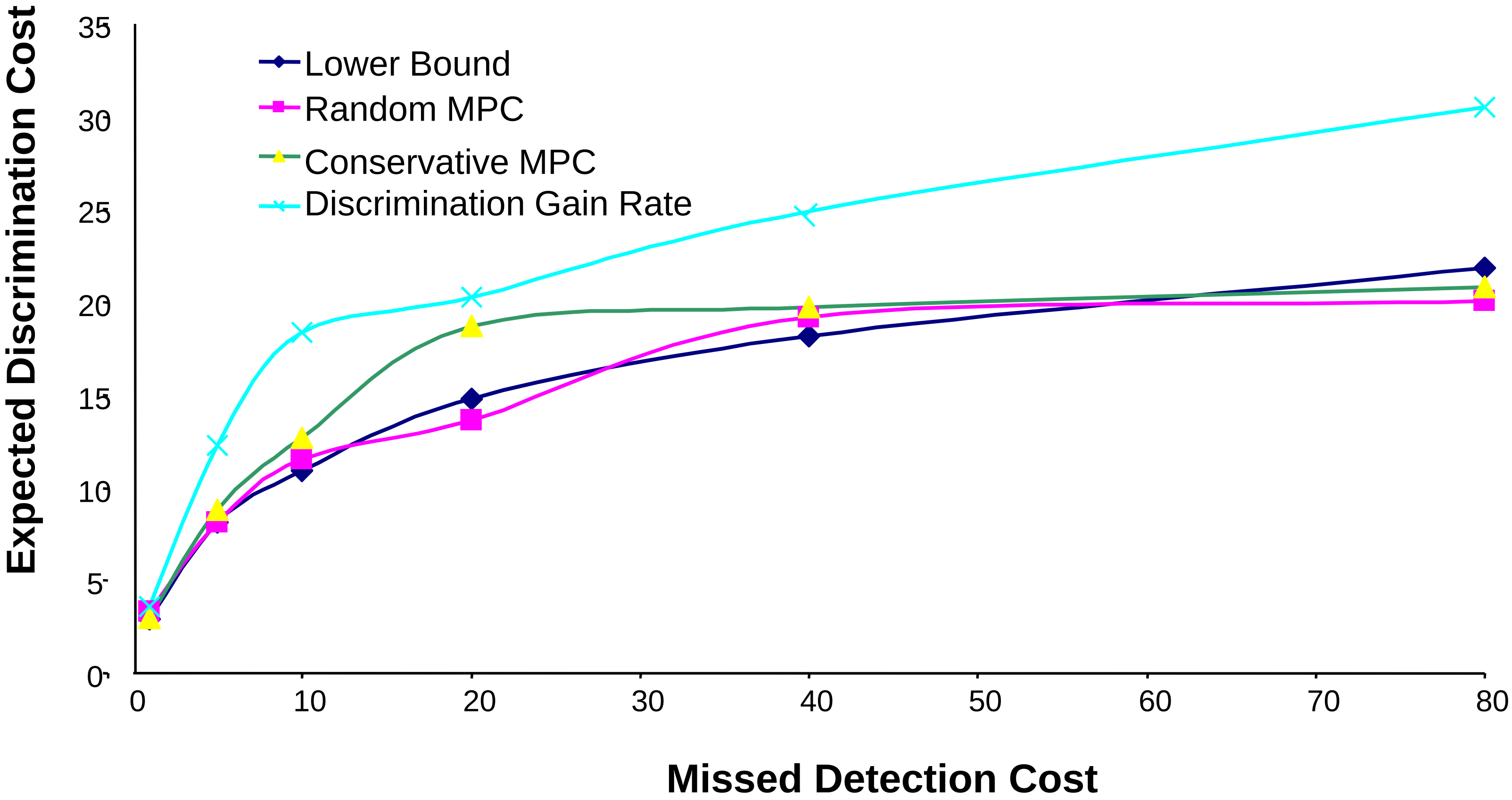
1) Control by objective
   - Baseline approach
   - Assumes all agents know information state, adapt accordingly
2) Control by geographical partitioning and local objectives
   - Define local objectives for each agent based on partitioning
   - Agents process own information, select actions
   - Human control reallocates responsibility
3) Control by functional partitioning
   - Agents constrained to use specified modes
   - Human control changes mode assignment
4) Control by action: No autonomy…

Example: Control by Objective

- Problem Description
  - Objects: 100 sites with 3 types of objects: cars, military vehicles, trucks
  - Sensors
    - Two modes: low-resolution (1 sec) and high-resolution (5 sec)
    - Binary-valued measurements: military or not military
    - Low-Res separates cars from others, trucks; High-Res separates others, cars and trucks
  - Constraints: 300 – 700 seconds of sensing time
  - Objective: MD for error of declaring military vehicle as car or truck, 1 for declaring car or truck as military vehicle, all after terminal time
  - Prior distribution: 10 % military vehicles, 20 % trucks, 70 % cars
- Algorithms for multi-mode sensors
  - Dynamic model predictive control algorithm using lower bound with 4 sensing actions per object lookahead horizon
  - Randomized model predictive control variation
  - Greedy
  - Lower bound for performance

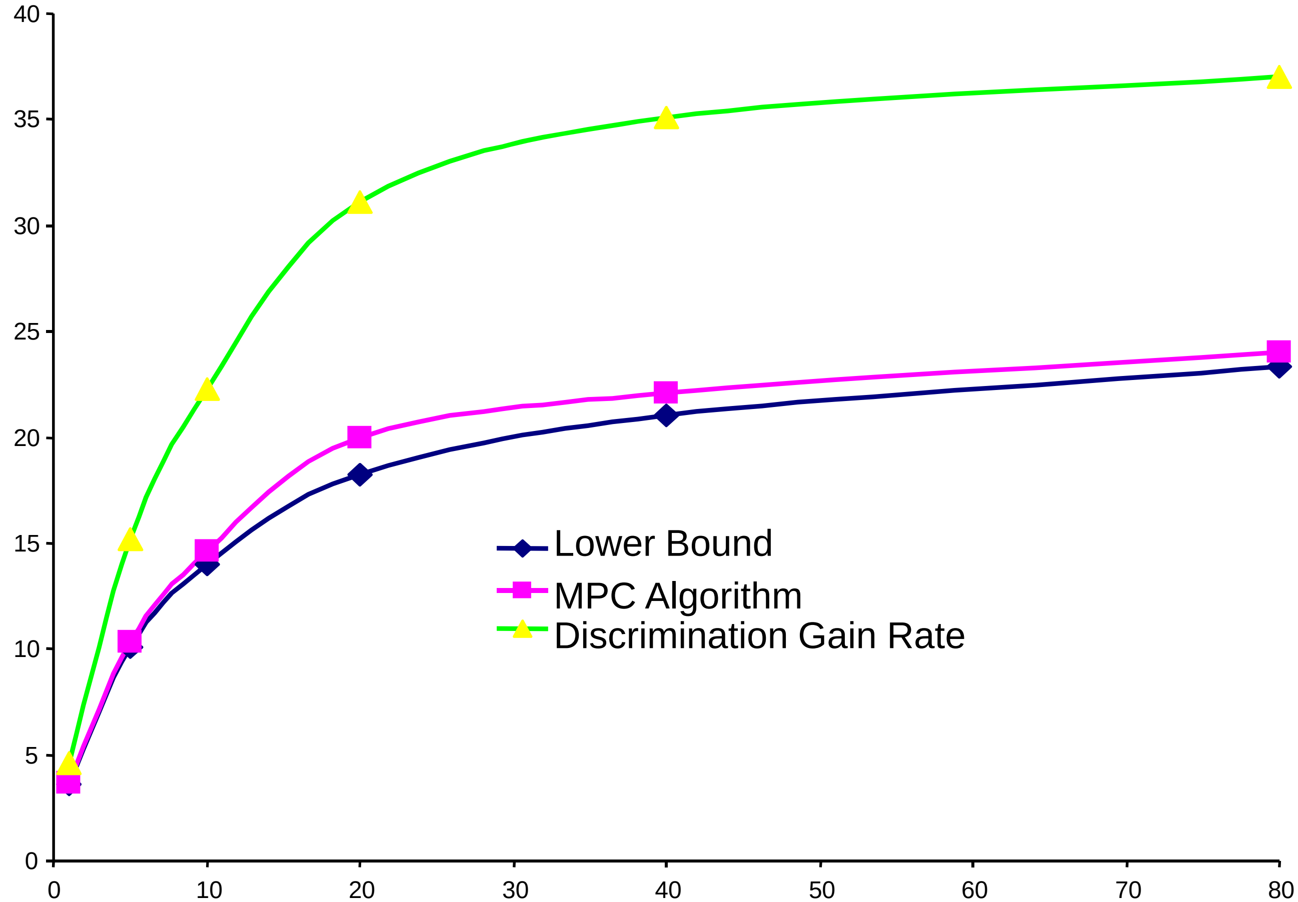
Multi-mode Single Agent Results

- 500 seconds of observations
- `Algorithms “outperform” bound!
  - Monte Carlo simulation has 3 % less high value targets than model



Two Agents, each with one mode

- 250 seconds of observations per agent
- Loss of performance over optimal partitioning of time among modes

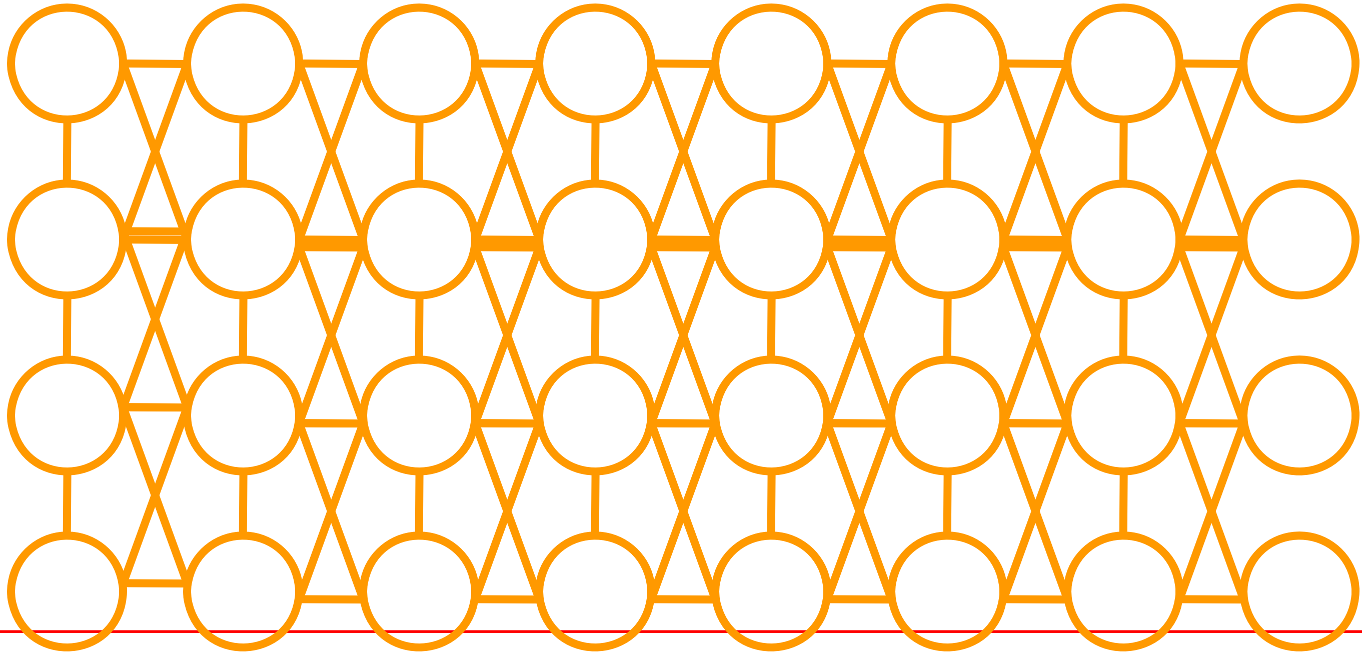


Paradigm Extension: Mobile Agents

- Viewable sites depend on agent positions
  - Slower time scale control
  - Focus on trajectory selection and mode
  - Sequencing of sites critical to set up future sites
- Mobile agents: trajectory and focus of attention control
  - Models where electronic steering is not feasible
  - Sequence-dependent setup cost for activities
- Simplify uncertainty: focus on risk of travel
  - Visiting a site accomplishes task that gains task value
  - Traversing among sites can result in vehicle failure and loss

Illustration of Problem

- Nodes represent sites, arcs represent feasible transitions
- Agents can travel among nodes using arcs
- Transitions on arcs are risky
- Visiting sites can collect value at sites; however, multiple visits do not add value



Mathematical Representation

- N sites (nodes in a graph) each containing a valued task
- Task may require specific agent type to visit site
- Underlying state at each site: x_i task is done or not
- Feasible transitions: arcs (i,j) with transition times t_{ij} and probability of successful transition p_{ij}
- Multiple agents K, each agent of a certain type
- Decisions: paths for each agent k among nodes
- Finite total travel time resource per agent k: T_k
- Agent states q_j: current node or 0 to indicate agent dead
- Discrete event dynamics: stochastic agent transitions, site transitions when agents visit
- Task values only obtained when task is not done yet

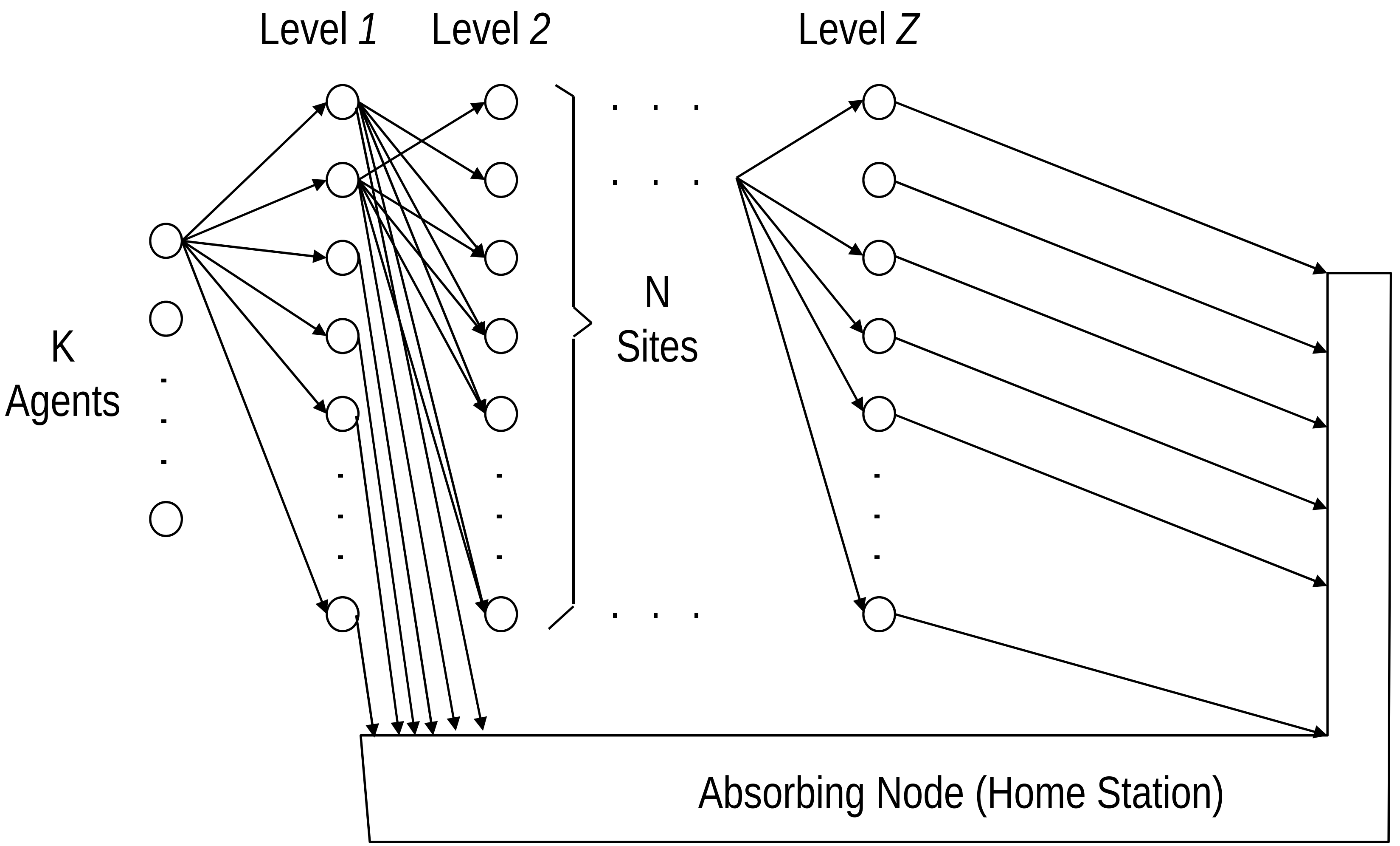
Variations on Human Control

1) Control by objective
   - Provide objectives in terms of values of site tasks and cost of losing agents
   - Agent control algorithms seek to maximize expected net value completed
2) Control by geographical partitioning
   - Partition site responsibility among agents, adapting site allocation in response to progress and workload
3) Control by action: Select activities of agents adaptively based on observations

Autonomous Control Algorithms: Theory Overview

- Without considering risk, problem becomes instance of multi-vehicle routing problem
  - NP-Hard!
- Can formally write as integer multi-commodity flow problem
  - Useful for development of approximate algorithms that can compute routes in real time
- Approximation approaches
  - Start with layered network representation
  - Lagrangian Relaxation
  - Rollout techniques

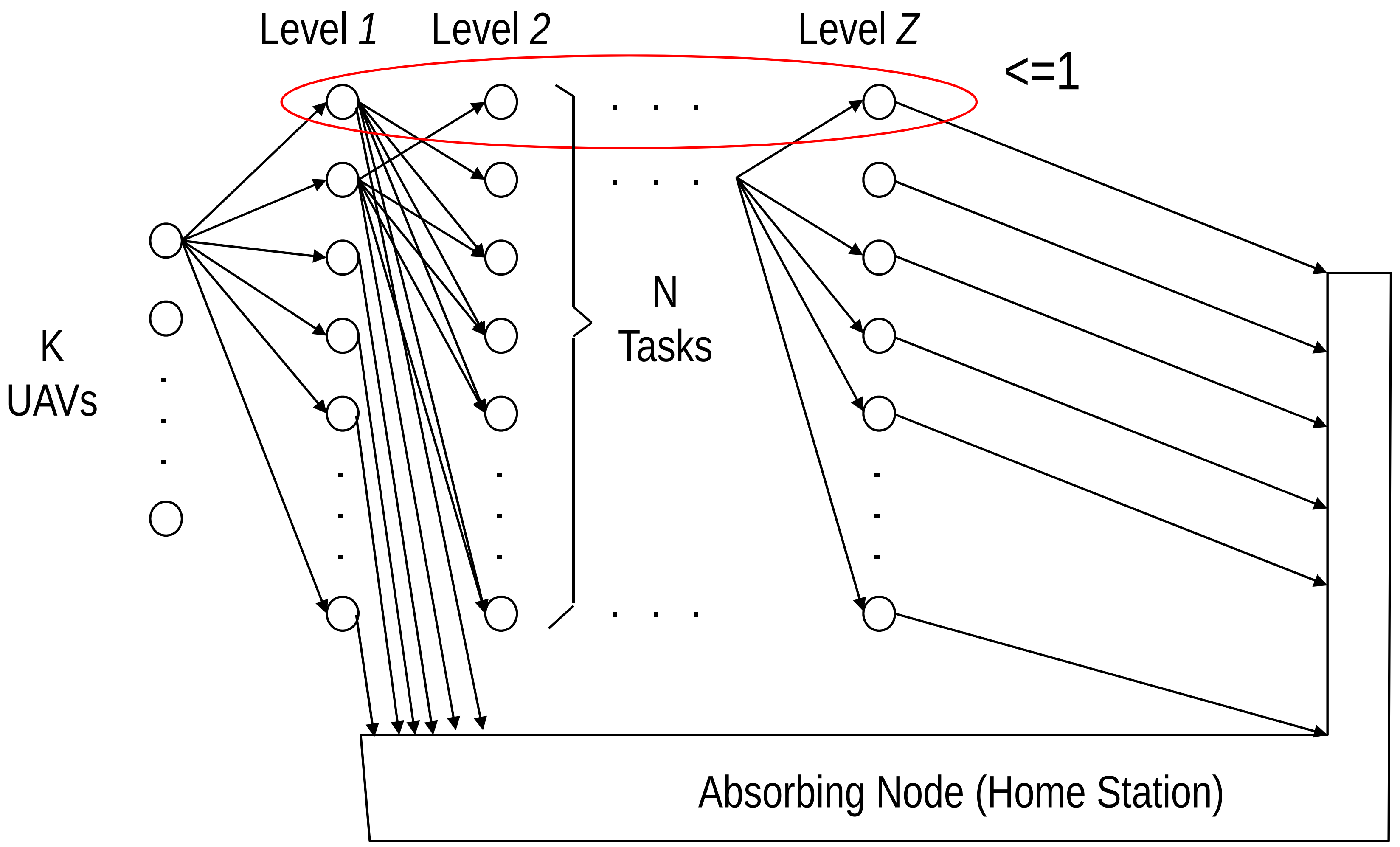
Discrete Event Task Network



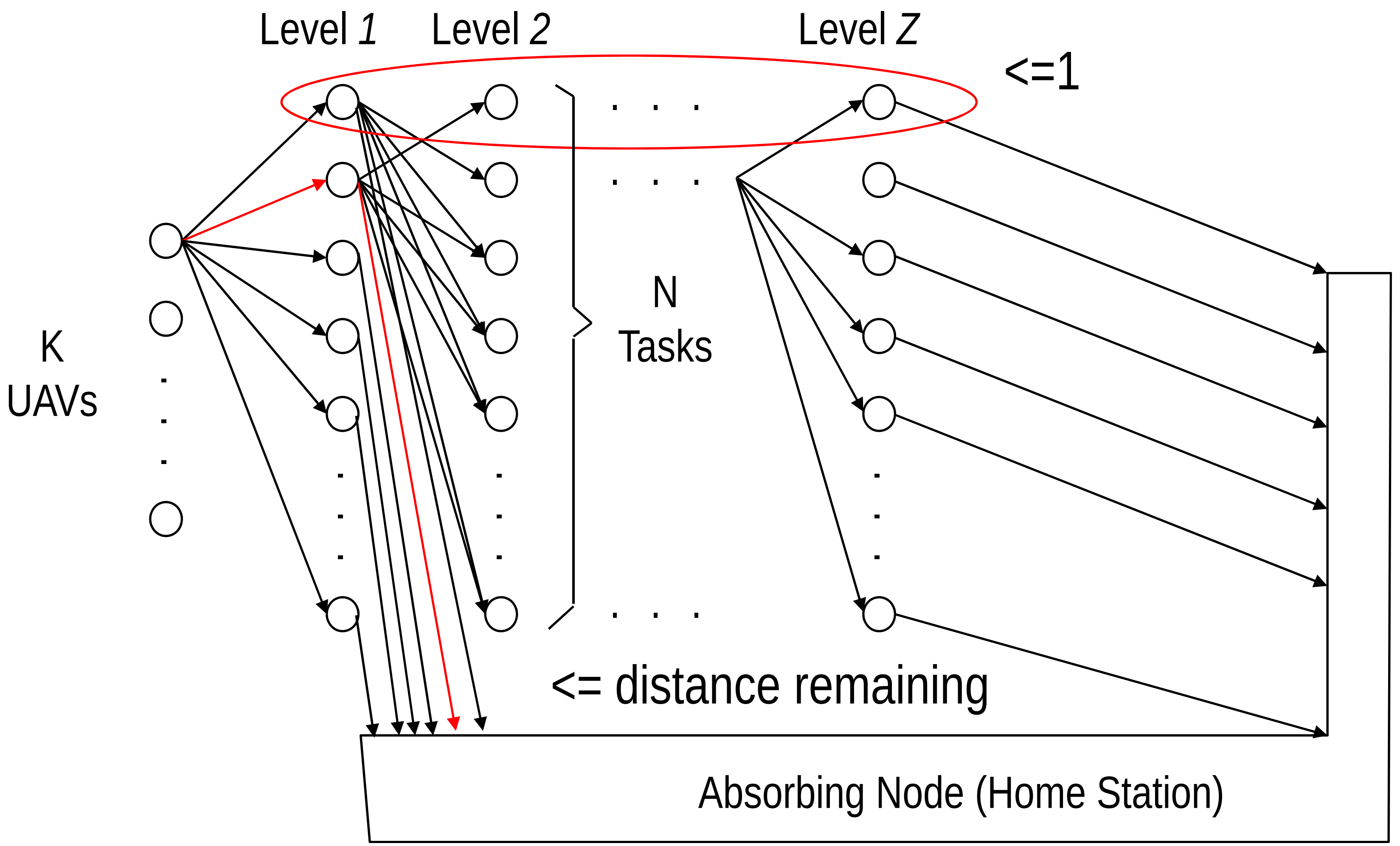
Multicommodity Integer Flow
Levels: task order assigned to Agents

Arcs: travel times
Nodes: Valued sites

Discrete Event Task Network



Discrete Event Task Network



Experiments

Problem	Depth	Single Rollout Value	Single Rollout Time	Multi Rollout Value	Multi Rollout Time	Lagr. Relax. Value	Lagr. Relax. Time
1	7	7000	0.16	9100	47	9000	109
1	14	11200	0.27	9900	68	10900	249
1	20	12400	0.38	10400	75	11100	360
2	7	6800	0.31	8400	193	9000	288
2	14	9400	0.61	12600	339	13200	471
2	20	10400	0.70	14400	372	14400	684
3	7	15000	0.22	18000	101	18000	146
3	14	24000	0.30	23500	144	25000	330
3	20	24000	0.33	23500	147	25000	452

- Rollout algorithms compete well with technique with much faster solution times

Extension: Risk on Arcs

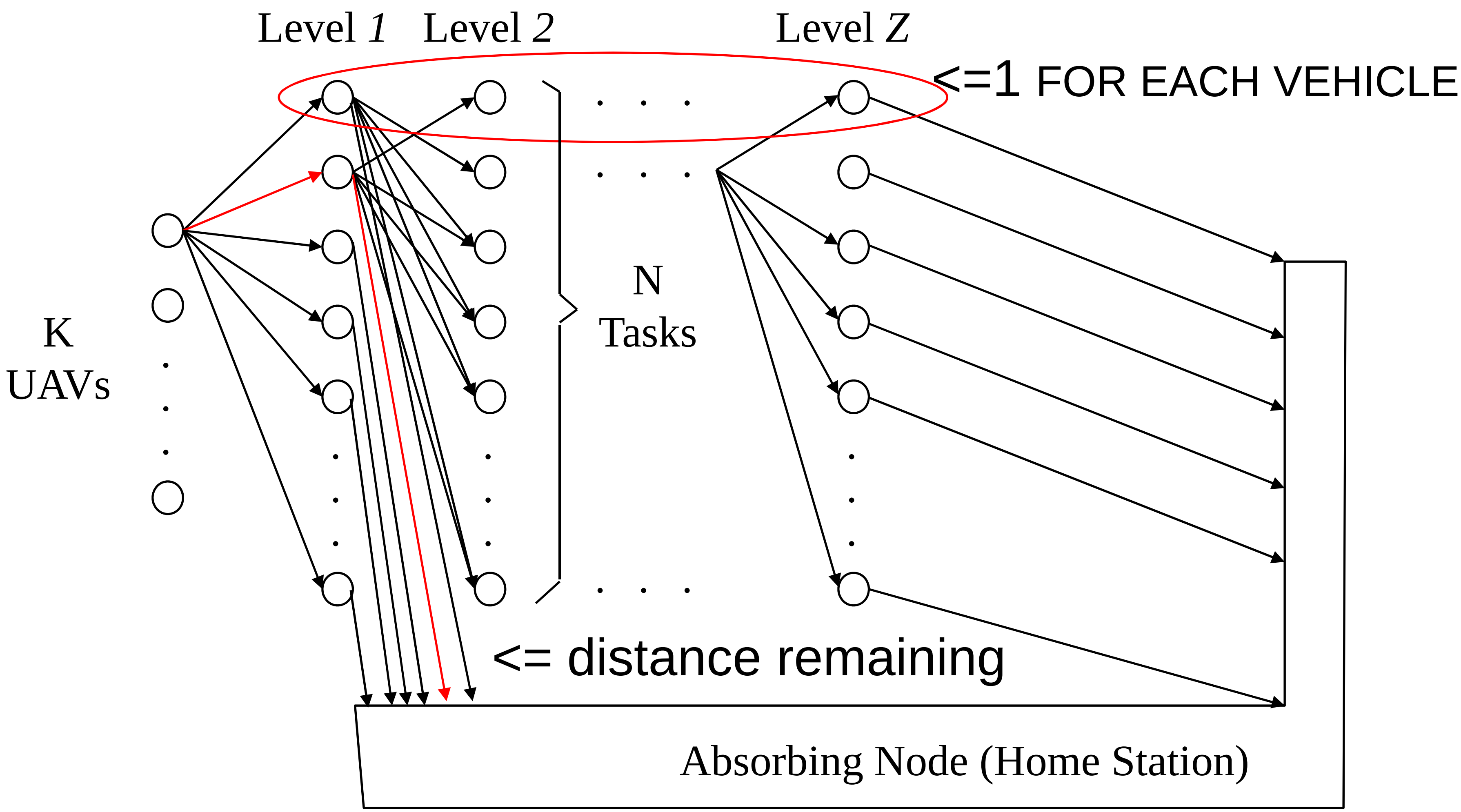
- Risky modification of integer multicommodity flow
  - Risk depends on task sequence
  - New objective to account for the possibility that scheduled tasks may not be completed
  - New constraints: multiple vehicles allowed to schedule same task (but each vehicle can only schedule a task once)

$$\max \sum_{j=1}^{n} V_j \left[1 - \prod_{k=1}^{K} \left[1 - \sum_{z1=1}^{Z} \left(\prod_{z=1}^{z1-1} \left(\sum_{a \in \mathcal{L}} \sum_{b \in \mathcal{O}} x_{a,b}^{z,k} p_{a,b} \right) \times \sum_{i \in \mathcal{L}} x_{i,j}^{z1,k} p_{i,j} \right) \right] \right] \tag{1}$$

subject to

$$\mathbf{D}\mathbf{x} \leq \mathbf{d},$$
$$\mathbf{E_2}\mathbf{x} \leq \mathbf{e_2},$$
$$\mathcal{N}\mathbf{x} = \mathbf{n}$$
$$\mathbf{x} \in \mathcal{B}$$

Risky Discrete Event Network



- More than one UAV can visit each task

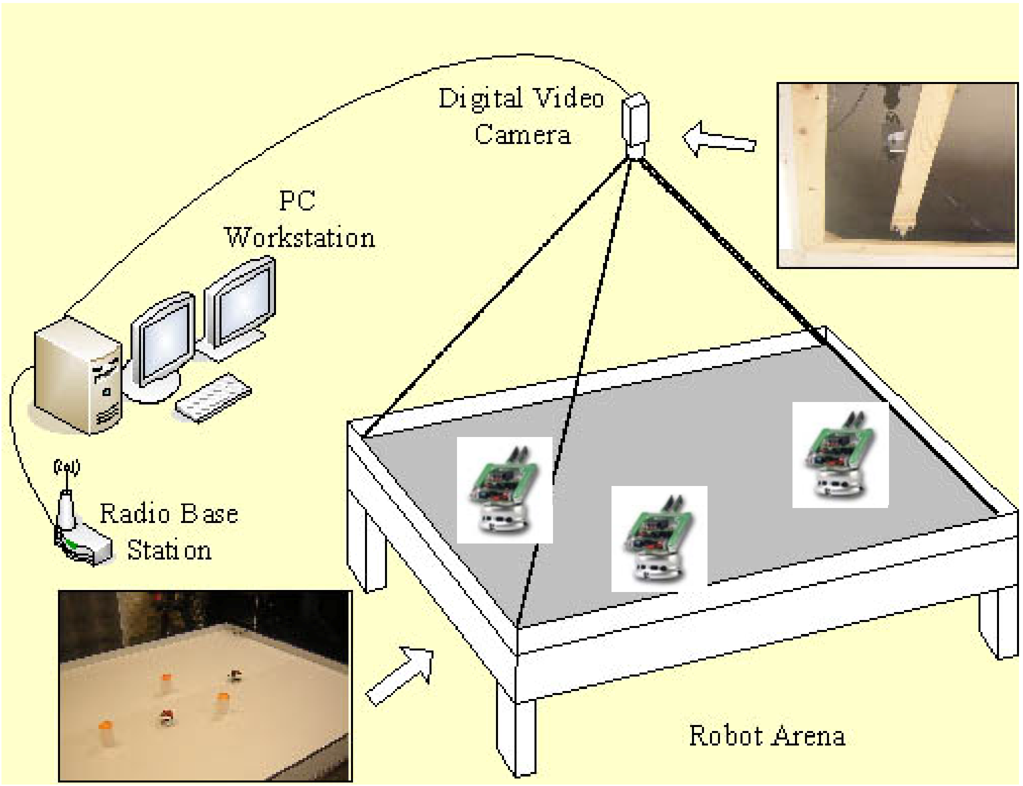
Experiments

Prob	Depth	Risk	Single Rollout Value	Risky Rollout Value	Coord Ascent Value	Hybrid Value	Risky Rollout Time (s)	Coord Ascent Time (s)
1	7	L	6648	8343	8831	8589	0.28	1.88
1	14	L	10158	10147	10538	10147	0.56	4.41
1	20	L	10356	10441	11241	10698	0.69	5.56
1	7	H	5435	6821	8249	7617	0.30	2.11
1	14	H	6963	8682	10536	9763	0.53	4.71
1	20	H	7427	10157	11090	10218	0.67	6.20
2	7	L	6328	8815	8933	8876	0.33	3.81
2	14	L	8697	12919	12532	12998	0.64	10.20
2	20	L	7375	13499	14351	14283	0.89	16.17
2	7	H	4366	7758	8530	8514	0.28	3.58
2	14	H	4896	11310	12322	12730	0.56	9.99
2	20	H	4634	12374	14114	13921	0.86	17.94

- Risk modeling important
- Tradeoff algorithm performance for computation speed

Experimental Platform for Research

- Multiple robots search for and perform tasks at BU's Mechatronics Lab
  - Can provide varying levels of operator control: human-automata teams
  - Control information displayed, risk to each operator using video



Future Activities

- Implement research experiments involving tasks with performance uncertainty in test facility
  - Vary tempo, size, uncertainty, information
- Implement autonomous team control algorithms to interact with operators in alternative roles
  - Supervisory control
  - Team partners
- Extend existing algorithms to different classes of tasks
  - Area search, task discovery, risk to platforms
- Develop approaches to assist operators in predicting behavior of automata teams in uncertain environments
- Collaborate with MURI team to design and analyze experiments involving alternative structures for human-automata teams