16-311-Q Introduction to Robotics Fall'17

Lecture 28: Multi-Robot Systems 1

Instructor:
Gianni A. Di Caro

جامعة كارنيجي ميلون في قطر
Carnegie Mellon University Qatar

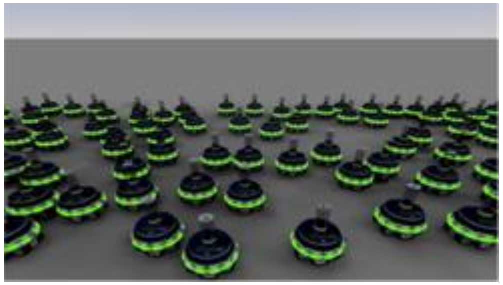
MULTI-ROBOT SYSTEMS

Pros and Cons

- Some tasks *needs* 2 or more robots
- Linear / superlinear *speedups*
- *Parallel and spatially distributed* system
- *Redundancy* of resources ➔ *Robustness*
- A robot *ecology* is being developed …

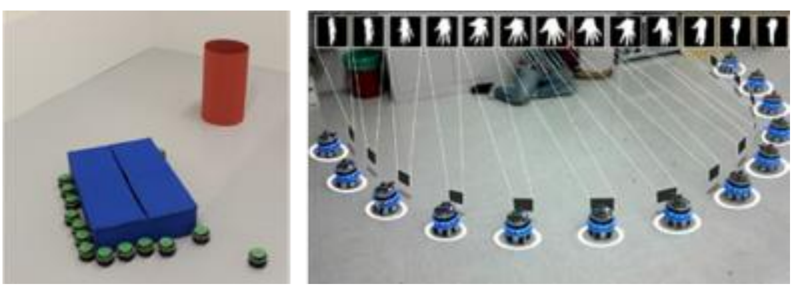
- Environment inherently *dynamic*
- Complex *g-local* interactions
- Access *shared* resources
- Need for (some) *coordination*
- Increased (state) *uncertainty*
- *Communication* issues
- Costs / Benefits ratio
- Practical problems ×*N*

BASIC TAXONOMY



Homogeneous system:
members are interchangeable

Heterogeneous system:
different members have different skills



Loosely coupled:
Being together is an advantage
but not a strict necessity
Speedup

Tightly coupled:
They need each other to successfully
complete the team task
Cooperation, Coordination

NON-COOPERATIVE VS. COOPERATIVE

- Non cooperative
- Maximization of individual utilities
- Equilibrium concepts
- Social welfare?

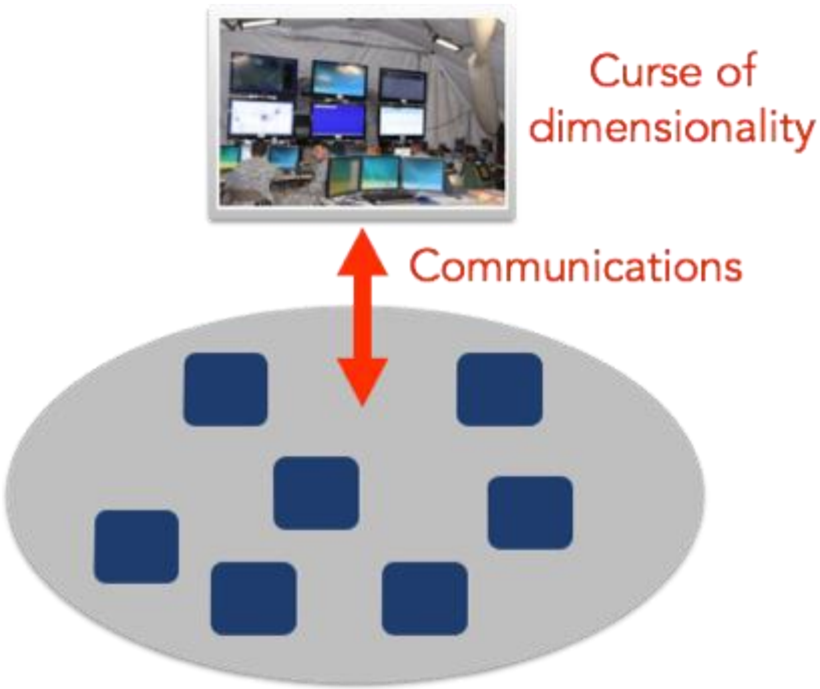
- Cooperative
- Optimization of individual utilities aiming to maximize a global utility
- Optimization concepts
- Social welfare!

BASIC TAXONOMY

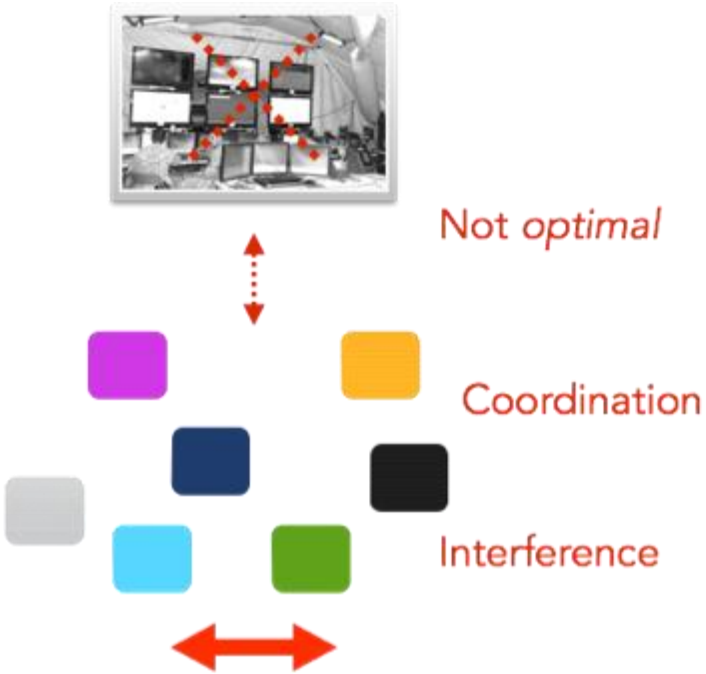
Cooperative (Benevolent) :
Agents are working together, forming a team

Competitive:
Competing for resources and utilities, adversarial scenario

BASIC TAXONOMY



Centralized control



Decentralized/Distributed control

Core Issues: Coordination and Planning

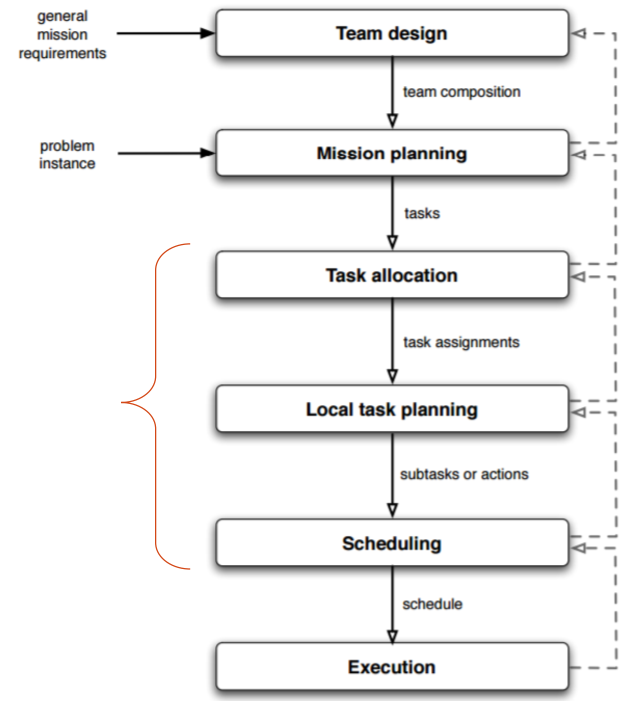
Decision / Action making

Explicit

Implicit

Motion control

(One) Central Problem: Multi-Agent/Robot Task Allocation (MRTA)



Team Mission

↓

Decomposition in sub-tasks

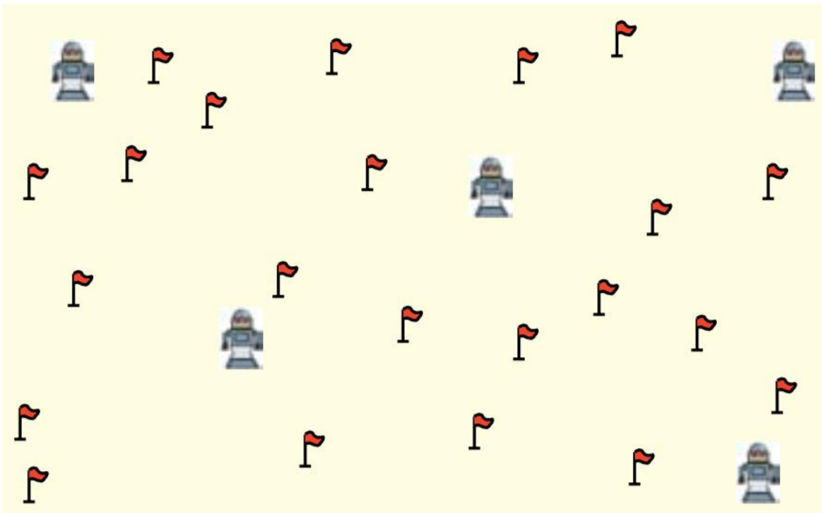
↓

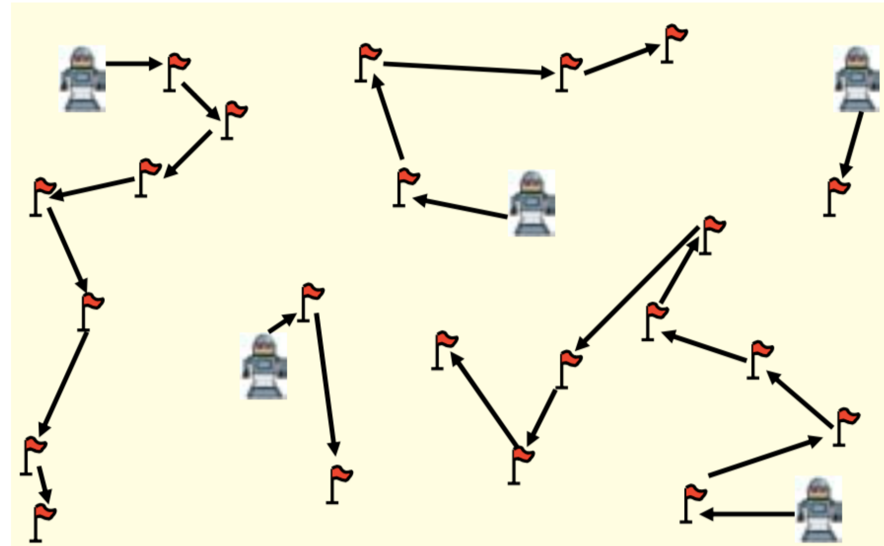
Team resources and status

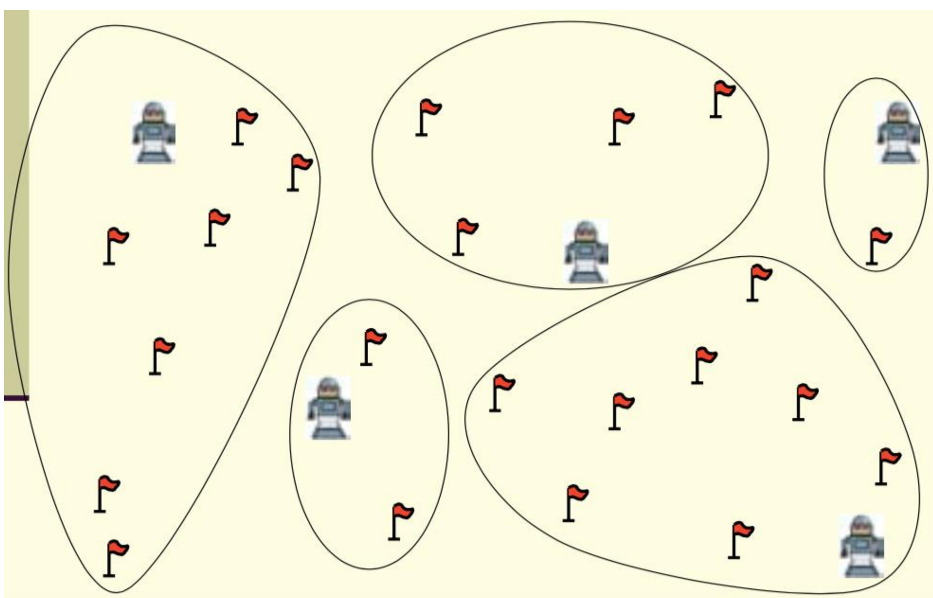
Dependencies (tasks, agents)

Who does what?
(and when, how)
Optimizing team performance

EXAMPLE: CUSTOMER SERVICE







Routing
(performance metric + constraints)

Customer Assignment
(performance metric + constraints)

MRTA: A Formal Definition (Opt)

Given:

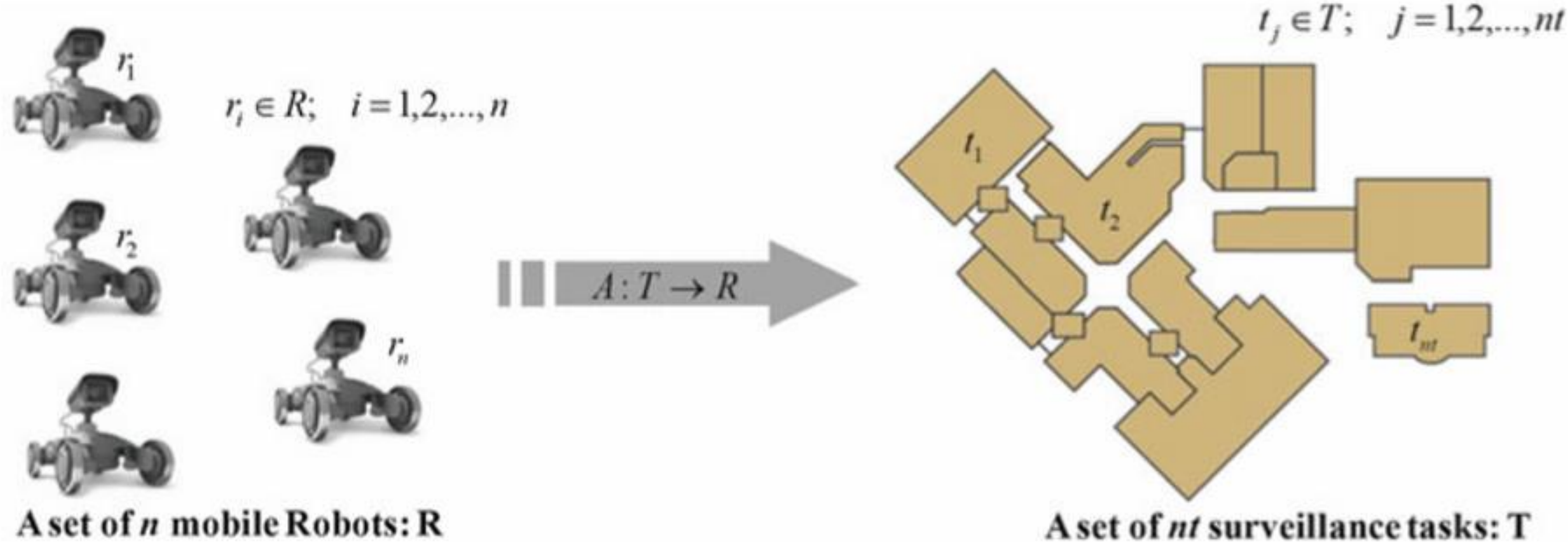
- ✓ A set of tasks, T
- ✓ A set of robots, R
- ✓ $\mathfrak{R} = 2^R$ is the set of all possible robot sub-teams E.g., $(r_1 = 0, r_2 = 0, r_3 = 1, r_4 = 0, r_5 = 1)$
- ✓ A robot sub-team utility (or cost) function: $\mathcal{U}_r$: $2^T \times \mathfrak{R} \rightarrow \mathbb{R} \cup \{\infty\}$ (the utility/cost sub-team r incurs by handling a subset of tasks)

- ✓ An allocation is a function $A: T \rightarrow \mathfrak{R}$ mapping each task to a subset of robots. $\mathfrak{R}^T$ is the set of all possible allocations

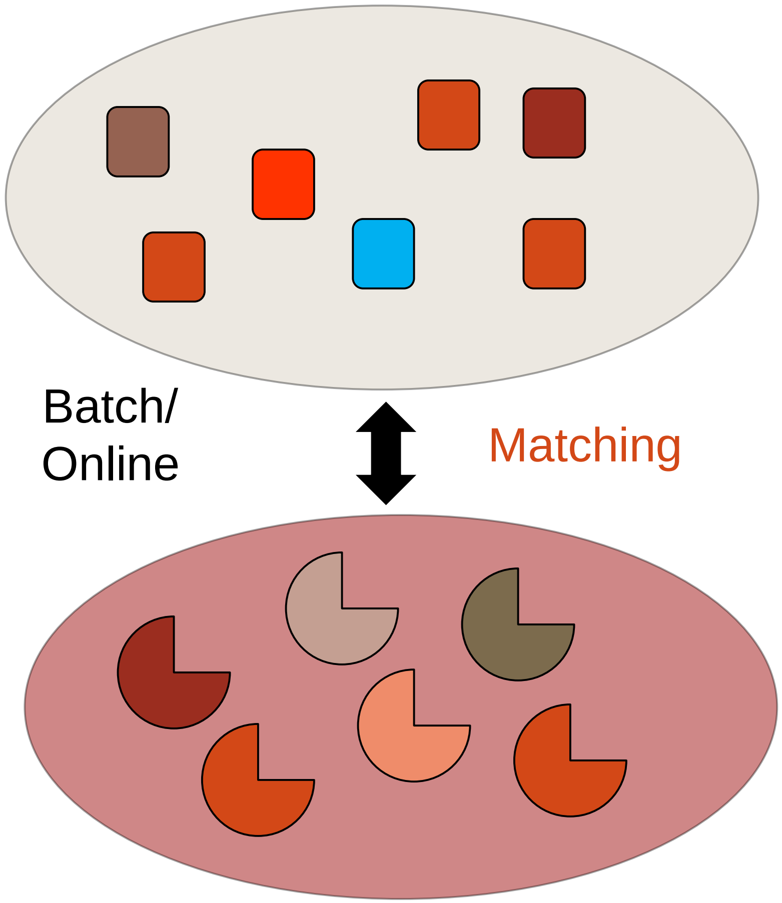
Find:

- ➢ The allocation $A^* \in \mathfrak{R}^T$ that maximizes (minimizes) a global, team-level utility (objective) function $\mathcal{U}: \mathfrak{R}^T \rightarrow \mathbb{R} \cup \{\infty\}$

Example: Surveillance



INTENTIONAL VS. EMERGENT



- **Explicit/Intentional TA**: robots explicitly cooperate and tasks are explicitly assigned to the robot
- **Implicit / Emergent TA**: tasks are assigned as the result of local interactions among the robots and with the environment

Utility function

- Utility function for a pair (*robot, task*)

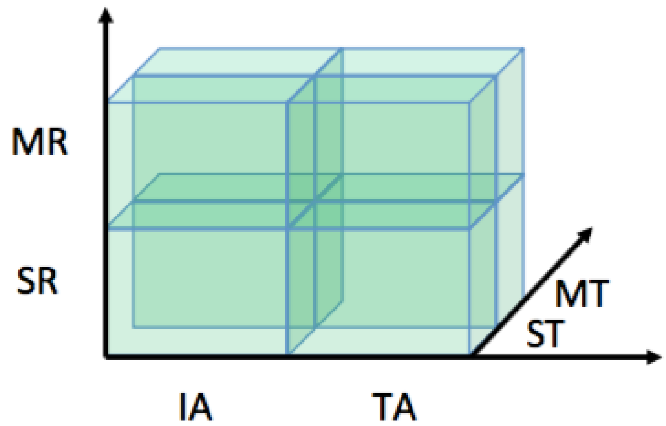
$$U_{rt} = \begin{cases} Q_{rt} - C_{rt} & \text{if } r \text{ is capable of executing } t \\ -\infty & \text{otherwise} \end{cases}$$

- *Q* and *C* are somehow *estimates* of Quality and Cost that account for all uncertainties, missing information, …
- Optimal allocation: Optimal based on all the available information → Rational decision-making

- For some problems, an agent's (sub-team's) utility for performing a task is independent of its utility for performing any other task.
- In general, this is not always true
- Our definition fails capturing *dependencies*

BASIC TAXONOMY

Task Type
Single robot (SR) versus multi robot (MR) tasks



Robot Type
Single-task (ST) versus multi –task (MT) robots

Allocation Type
Instantaneous assignment (IA) versus time-extended assignment (TA)

(Gerkey and Mataric, 2004)

Assumption: Individual tasks can be assigned independently of each other and have independent robot utilities

WHY A TAXONOMY?

- A lot of “different MR scenarios”
- A lot of “different” MRTA methods
- Analysis and comparisons are difficult!

- Taxonomy → Single out core features of a MRTA scenario
- Allow to understand the complexity of different scenarios
- Allow to compare and evaluate different approaches
- A scenario is identified by a 3-vector (e.g., ST-MR-TA)

ST-SR-IA: Linear Assignment

If |R|=|T| the problem becomes a **linear assignment** and a polynomial-time solution does exist!

$$\max \quad \sum_{r=1}^{|R|} \sum_{t=1}^{|T|} U_{rt} x_{rt}$$

$$s.t. \quad \sum_{r=1}^{|R|} x_{rt} = 1 \qquad t = 1, \ldots |T|$$

$$\sum_{t=1}^{|T|} x_{rt} = 1 \qquad r = 1, \ldots |R|$$

$$x_{rt} \in \{0, 1\}$$

The Hungarian algorithm has complexity $O(|T|^3)$

In a centralized architecture, with each robot sending its |T| utilities to the controller, $O(|T|^2)$ messages are needed

Assignment with hundreds of robots in < 1s

ST-SR-IA: Linear Assignment

- What if $|R| \neq |T|$?
- To preserve polynomial time solution, “dummy” robots or tasks can be included in a two-step process
- If $|R| < |T|$: $(|T|-|R|)$ dummy robots are added and given very low utility values with respect to all tasks, such that that their assignment will not affect the optimal assignment of $|R|$ tasks to the “real” robots
- The remaining $|T|-|R|$ tasks (i.e., assigned to the dummy robots) can be optimally assigned in a second round, which will likely feature # of robots greater than the # of tasks
- If $|T| < |R|$: Dummy tasks with very low, flat, utilities are introduced such that their assignment will not affect the assignment of real tasks

ST-SR-IA: Iterated Assignment

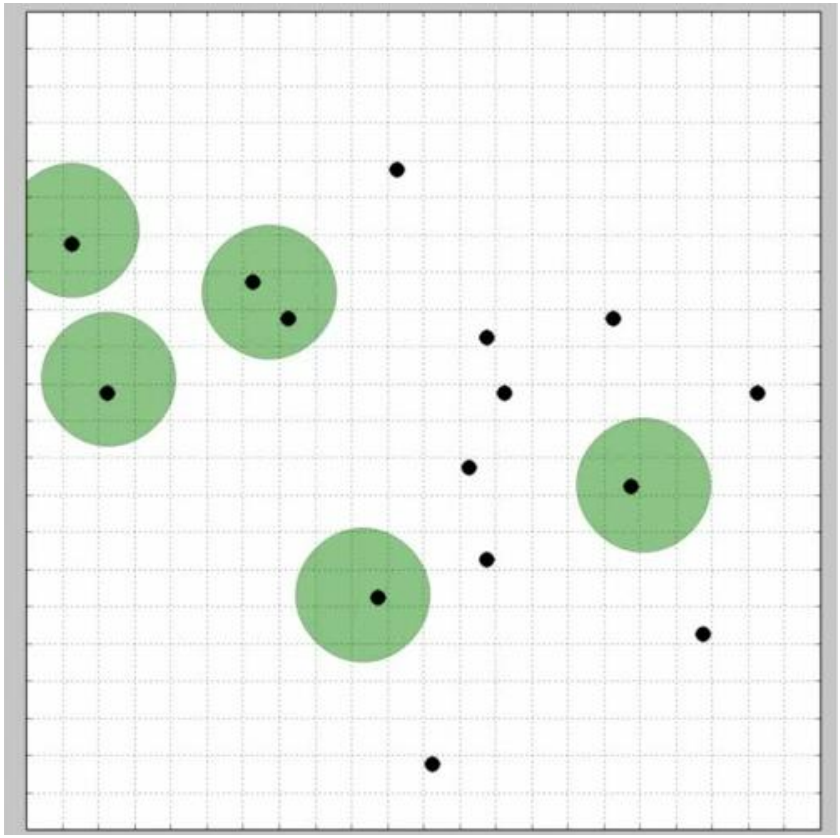
- Not always full/final task information and utility is available since the beginning of the operations
- How to deal with new / revised evidence (utility) in an *iterative* scheme?
- Recompute from scratch to solve the assignment, or, **adapt greedily**:

Broadcast of Local Eligibility (BLE, 2001), worst-case 50% opt

1. If any robot remains unassigned, find the robot–task pair (i, j) with the highest utility. Otherwise, quit.
2. Assign robot i to task j and remove them from consideration.
3. Go to step 1.

- 2-competitive: $U(\text{BLE}) \geq c \cdot U(\text{OptOffline})$ - $a, \qquad c = 2$
- L-ALLIANCE (1998) can *learn* the best assignments over time

EXAMPLES: CMOMMT, SOCCER



Cooperative multi-robot observation of multiple moving targets (MT)

- Robots are interchangeable → it is often advantageous to allow any player to take on any role within the team based on scenarios
- Iterated assignment problem in which the robots' roles are periodically reevaluated, usually at a frequency of about 10 Hz.