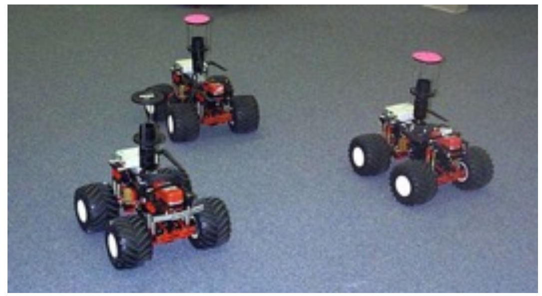
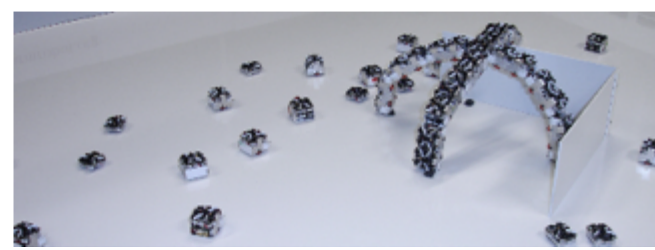
CMU 15-781

Lecture 21: Multi-Robot Systems

Teacher:
Gianni A. Di Caro

Multi-Robot Systems?





Multi-Robot Systems?

So far: 

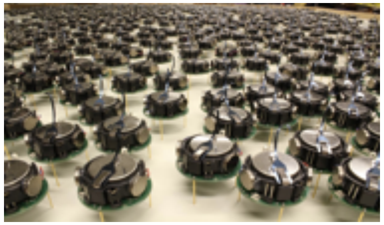
- How to *represent* world and knowledge
- How to make *rational decisions*
- How to *learn* to make rational decisions
- How to take decisions as a *collective*

Our rational (AI) agent was quite abstract → *Physical* AI agents

- Systems of **multiple physical agents** embedded in environments subject to the *laws of physics*
- Subject to **physical constraints and limitations** for motion/action, perception, communication, computation
- **Partial knowledge** and **uncertainty** are inherent
- **Autonomy** in acting and decision-making

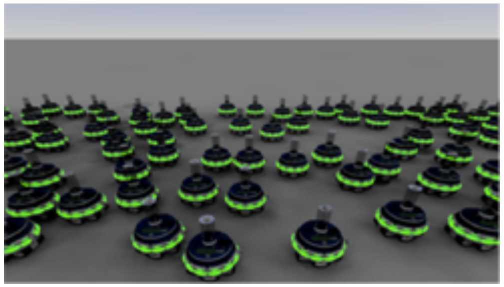
Why "Multi"-Robot Systems?

- 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 …



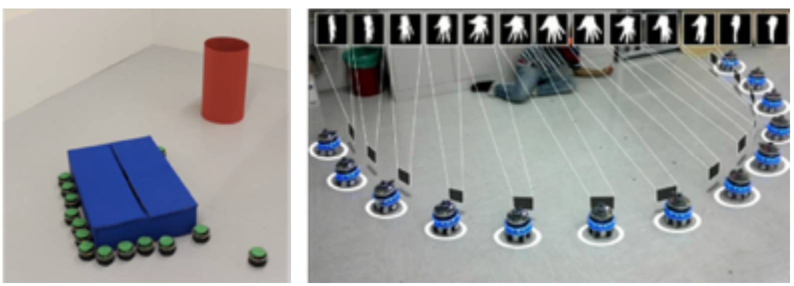
- Environment inherently *dynamic*
- Complex *g-local* interactions
- Access *shared* resources
- Need for (some) *coordination*
- Increased (state) *uncertainty*
- *Communication* issues
- Costs / Benefits ratio
- Practical problems $\times N$

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

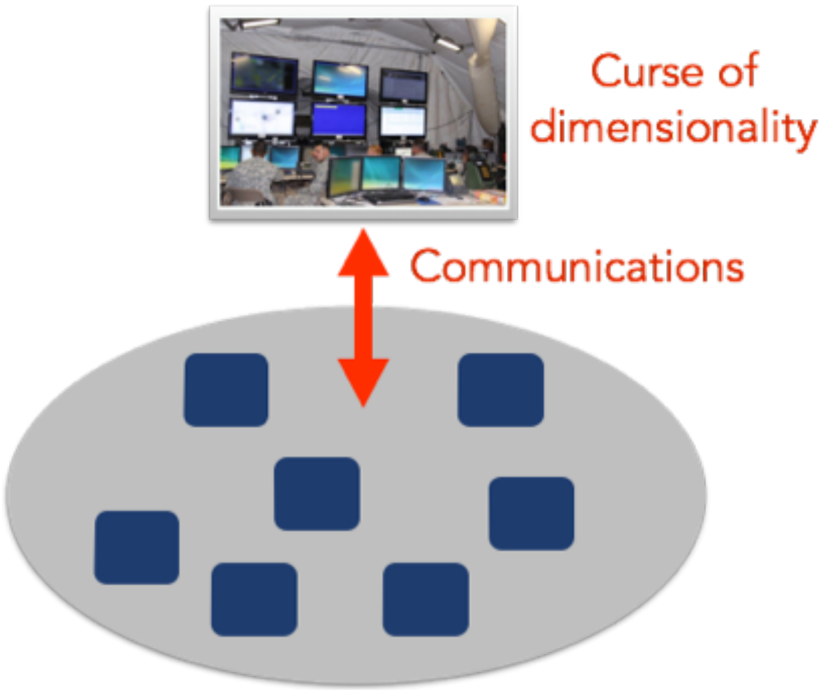
Tightly coupled:
They need each other to successfully complete the team task
Cooperation, Coordination

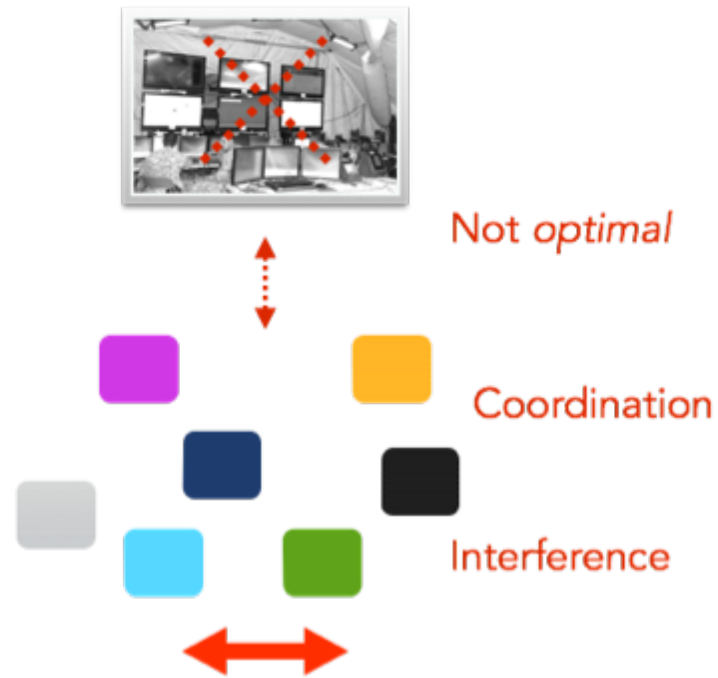
Basic Taxonomy

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

Competitive:
Robots competing for resources, are in *adversarial* scenario

Basic Taxonomy





Centralized control

Decentralized/Distributed control

Central Problem: Multi-Robot Task Allocation (MRTA)

Team **Mission**

↓

Decomposition in **sub-tasks**

↓

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

← Team **resources** and **status**

← **Dependencies** (tasks, agents)

MRTA: A Formal Definition (Opt)

Given:

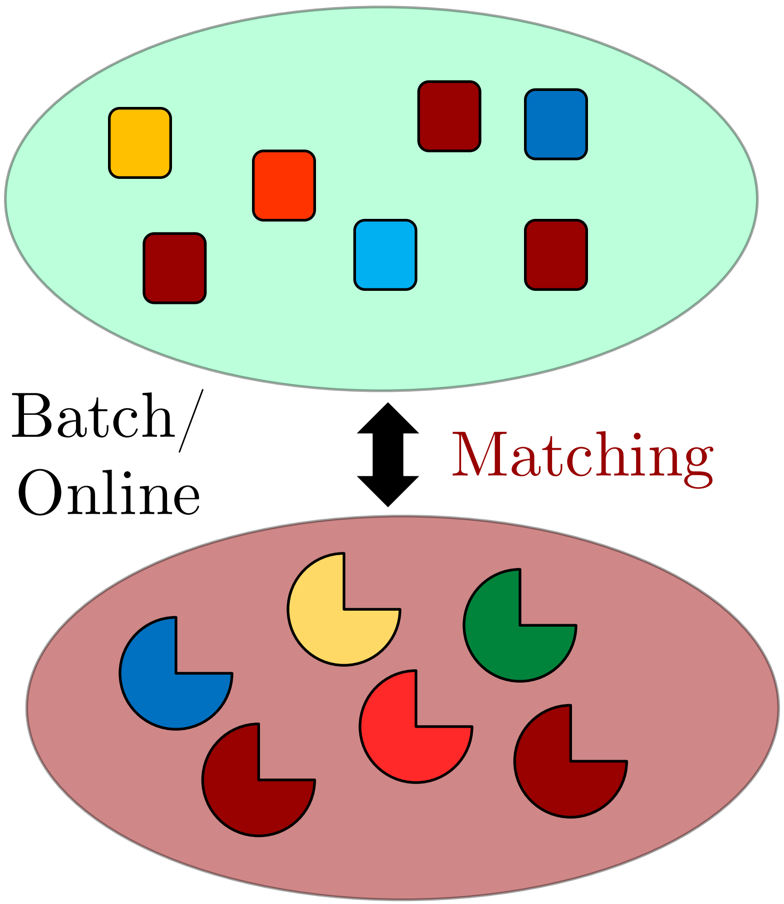
- ✓ A set of tasks, T
- ✓ A set of robots, R
- ✓ $\mathfrak{R} = 2^R$ is the set of all possible robot sub-teams (e.g., $(r_1 = 0, r_2 = 0, r_3 = 1, r_4 = 0, r_5 = 1)$)
- ✓ A robot sub-team utility (or cost) function: $\mathcal{U}_r\colon 2^T \rightarrow \mathbb{R} \cup \{\infty\}$ (the utility/cost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

- ✓ An allocation is a function $A\colon T \rightarrow \mathfrak{R}$ 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 $\mathcal{U}\colon \mathfrak{R}^T \rightarrow \mathbb{R} \cup \{\infty\}$

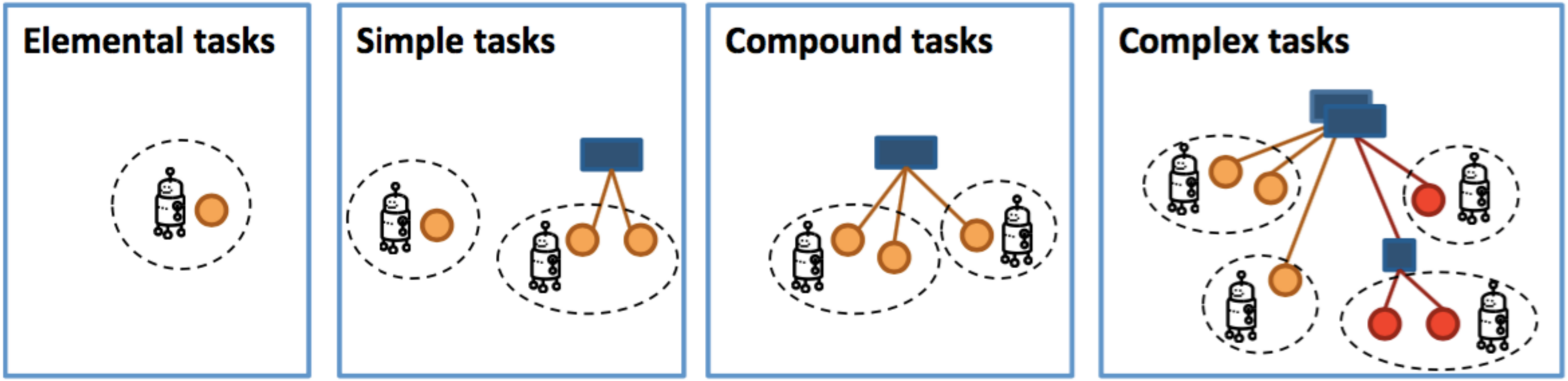
Intentional / Emergent



- **Explicit/intentional TA:** robots explicitly cooperate and tasks are explicitly assigned to the robot

- **Emergent TA:** tasks are assigned *as the result* of local interactions among the robots and with the environment

Tasks



(Zlot, 2006)

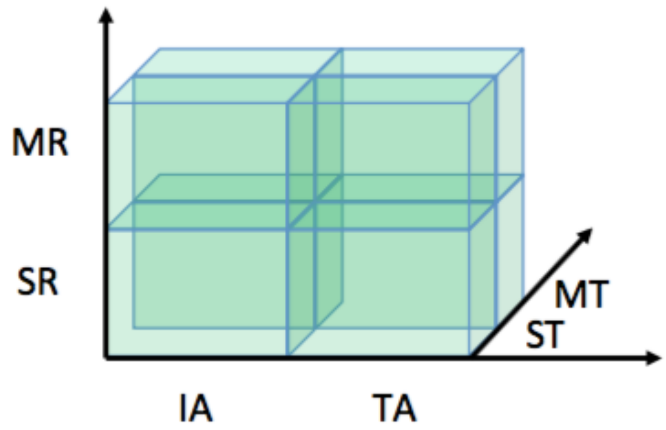
Utility function

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

- Q and C are somehow **estimates** that account for all uncertainties, missing, information, ...
- Optimal allocation: Optimal based on all the available information $\rightarrow$ *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, 2006)

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** $\rightarrow$ 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 exist!

$$
\begin{aligned}
\max \quad & \sum_{r=1}^{|R|} \sum_{t=1}^{|T|} U_{rt} x_{rt} \\
s.t. \quad & \sum_{r=1}^{|R|} x_{rt} = 1 \qquad t = 1, \ldots |T| \\
& \sum_{t=1}^{|T|} x_{rt} = 1 \qquad r = 1, \ldots |R| \\
& x_{rt} \in \{0, 1\}
\end{aligned}
$$

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
- 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 is available since the beginning of the operations
- How to deal with new / revised evidence (utility) in an **iterative** scheme?
- Recompute from scratch 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.

ST-SR-IA: Online Assignment

- Tasks are revealed one at-a-time
- If robots can be *reassigned*, then solving each time the linear assignment provides the optimal solution

MURDOCH (2002)

When a new task is introduced, assign it to the most fit robot that is currently available.

- *Farthest Neighbor* algorithm
- Performance bound of FNA is the best possible for any on-line assignment algorithm (Kalyana-sundaram, Pruhs 1993).

ST-SR-TA: Generalized Assignment

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

Robots gets a schedule of tasks

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

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

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

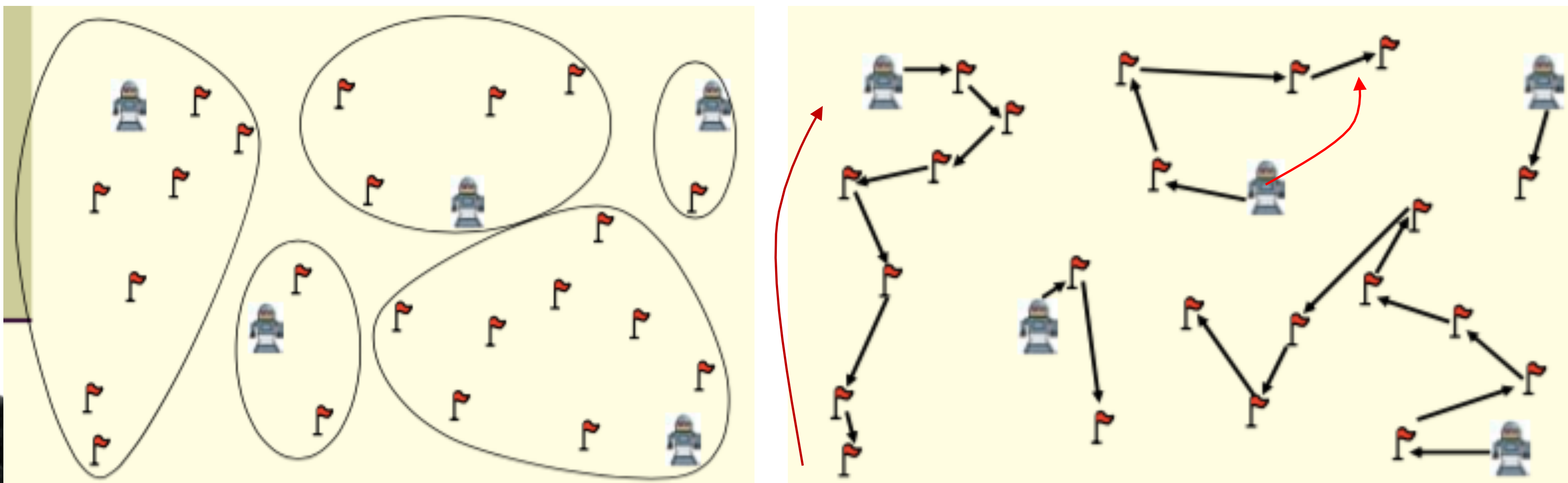
The "budget" constraints restricts the max number T_r of tasks (or the total time/energy to execute them based on some cost parameter c) that can be assigned to robot r

NP-hard!

ST-SR-TA: Generalized Assignment

If dependencies / constraints are included, “more” NP-Hard
→ If the utility is related to *traveling distances* the problem falls in the class of mTSP, VRP problems

Multi-robot routing



MT-SR-IA: Generalized Assignment

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

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

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

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

Robots can work in || on multiple tasks

The "capacity" constraint explicitly restricts the max number T_r of tasks that robot r can take, this time simultaneously

Not common in the instances from MRTA

NP-hard!

MT-SR-TA: VRP

Robots can work in || on multiple tasks and
have a time-extended schedule of tasks:
quite uncommon in current MR literature

Vehicle routing problems with capacity constraints and pick-up and delivery fall in this category:

- Multiple vehicles transporting multiple items (goods, people) and picking up items along the way
- Between a pick-up and delivery location the vehicle is dealing with MT
- Visiting multiple locations is equivalent to TA

NP-hard!

ST-MR-IA: Set Partitioning Coalition Formation

- Model of the problem of dividing (partitioning) the set of robots into *non-overlapping sub-teams* (**coalitions**) to perform the given tasks instantaneously assigned
- This problem is mathematically equivalent to *set partitioning problem* in combinatorial optimization.

Cover (Partition) the elements in R (Robots) using the elements in CT (feasible coalition-task pairs) without duplicates (overlapping) and at the min cost / max utility

CT

R						
1	x	x			x	
2	x		x			
3		x		x		
4			x			x
5		x	x			x

NP-hard!

MT-MR-IA: Set Covering Coalition Formation

- Model of the problem of dividing (partitioning) the set of robots into *sub-teams* (**coalitions**) to perform the given tasks instantaneously assigned. Overlap is admitted to model MT
- This problem is mathematically equivalent to *set covering problem* in combinatorial optimization.

Cover (Partition) the elements in R (Robots) using the elements in CT (feasible coalition-task pairs) admitting duplicates (overlapping) and at the min cost / max utility

CT

R						
1	x	x			x	
2	x		x			
3		x		x		
4			x			x
5		x	x			x

NP-hard!

Other cases

- ST-MR-TA: Involves both coalition formation and scheduling, and it's mathematically equivalent to MT-SR-TA
- MT-MR-TA: Scheduling problem with multiprocessor tasks and multipurpose machines
- *Modeling of dependencies?* → *G. Ayorkor Korsah, Anthony Stentz, and M. Bernardine* Dias. 2013. A comprehensive taxonomy for multi-robot task allocation. *Int. J. Rob. Res.* 32, 12 (October 2013), 1495-1512.

Solution approaches

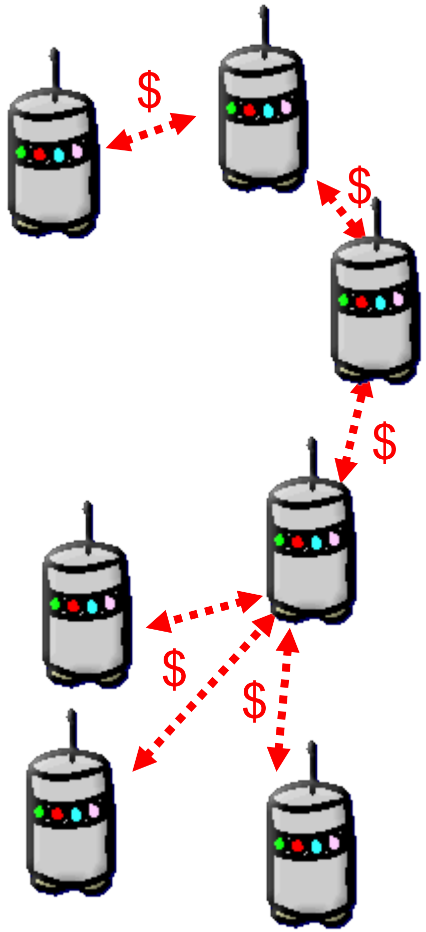
- Use the reference optimization models in a centralized scheme, solving the problems to optimality (e.g., Hungarian algorithm, IP solvers using branch-and-bound, optimization heuristics)
- Use the reference optimization models adopting a top-down decentralized scheme (e.g., all robots employ the same optimization model, and rely on local information exchange to build the model)
- Adopt different solution models avoiding to explicitly formulate optimization problems.
- Market-based approaches are an effective and popular option
- Emergent/Swarm approaches: effective / simpler alternative

Market-based: Basic Ideas

- Based on the economic model of a free market
- Each robot seeks to maximize individual "profit"
- Individual profit helps the common good
- An auctioneer (i.e. a robot spotting a new task) offers tasks (or roles, or resources) in an *announcement* phase
- Robots can negotiate and bid for tasks based on their (estimated) utility function
- Once all bids are received or the deadline has passed, the auction is cleared in the *winner determination phase*: the auctioneer decides which items to award and to whom.
- Decisions are made locally but effects approach optimality
  - Preserve advantages of distributed approach

Market-based: Basic Ideas

- Robots model an economy:
  - Accomplish task → Receive revenue
  - Consume resources → Incur cost
  - Robot goal: maximize own profit
  - Trade tasks and resources over the market (auctions)
- By maximizing individual profits, team finds better solution
- Time permitting → more centralized
- Limited computational resources → more distributed



Market-based: Basic Ideas

- Utility = Revenue – Cost
- Team revenue is sum of individual revenues
- Team cost is sum of individual costs
- Costs and revenues set up per application
  - Maximizing individual profits must move team towards globally optimal solution
- Robots that produce well at low cost receive a larger share of the overall profit

Market-based: Implementations

- **MURDOCH** (Gerkey and Matarić, IEEE Trans. On Robotics and Automation, 2002 / IJRR 2004)
- **M+** (Botelho and Alami, ICRA 1999)
- **TraderBots** (Dias et al., multiple publications 1999-2006)

Summary

- Characteristics and basic taxonomy of multi-robot systems
- Taxonomy of multi-robot task allocation (MRTA) problems
- Optimization models for the different classes of MRTA problems
- Computational complexity of the different classes
- Basic solution approaches exploiting the optimization models
- Basic ideas about market-based methods