DISTRIBUTED CONSTRAINT HANDLING AND OPTIMIZATION

[Some slides are taken from the material of the book by G. Weiss, *Multiagent Systems*, second edition, The MIT Press, 2013]

Video segment: scheduling of sense/sleep cycles of sensors

- Some fixed sensors are deployed in a city and tasked with detecting vehicles that travel along the roads (by A. Farinelli, A. Rogers, and N. Jennings)
- https://vimeo.com/48231842

- This is an instance of distributed constrained optimization problems which represent several real-world problems, where instead of amount of time agents should decide over time of a meeting, amount of energy, …

Constraint Networks

A constraint network $\mathcal{N}$ is formally defined as a tuple $\langle X, D, C \rangle$ where:

- $X = \{x_1, \ldots, x_n\}$ is a set of discrete variables;
- $D = \{D_1, \ldots, D_n\}$ is a set of variable domains, which enumerate all possible values of the corresponding variables; and
- $C = \{C_1, \ldots, C_m\}$ is a set of constraints; where a constraint C_i is defined on a subset of variables $S_i \subseteq X$ which comprise the scope of the constraint
  - $r = |S_i|$ is the arity of the constraint
  - Two types: hard or soft

Hard constraints

- A hard constraint C_i^h is a relation R_i that enumerates all the valid joint assignments of all variables in the scope of the constraint.

$$R_i \subseteq D_{i_1} \times \ldots \times D_{i_r}$$

R_i	x_j	x_k
	0	1
	1	0

Soft constraints

- A soft constraint C_i^s is a function F_i that maps every possible joint assignment of all variables in the scope to a real value.

$$F_i : D_{i_1} \times \ldots \times D_{i_r} \rightarrow \Re$$

F_i	x_j	x_k
2	0	0
0	0	1
0	1	0
1	1	1

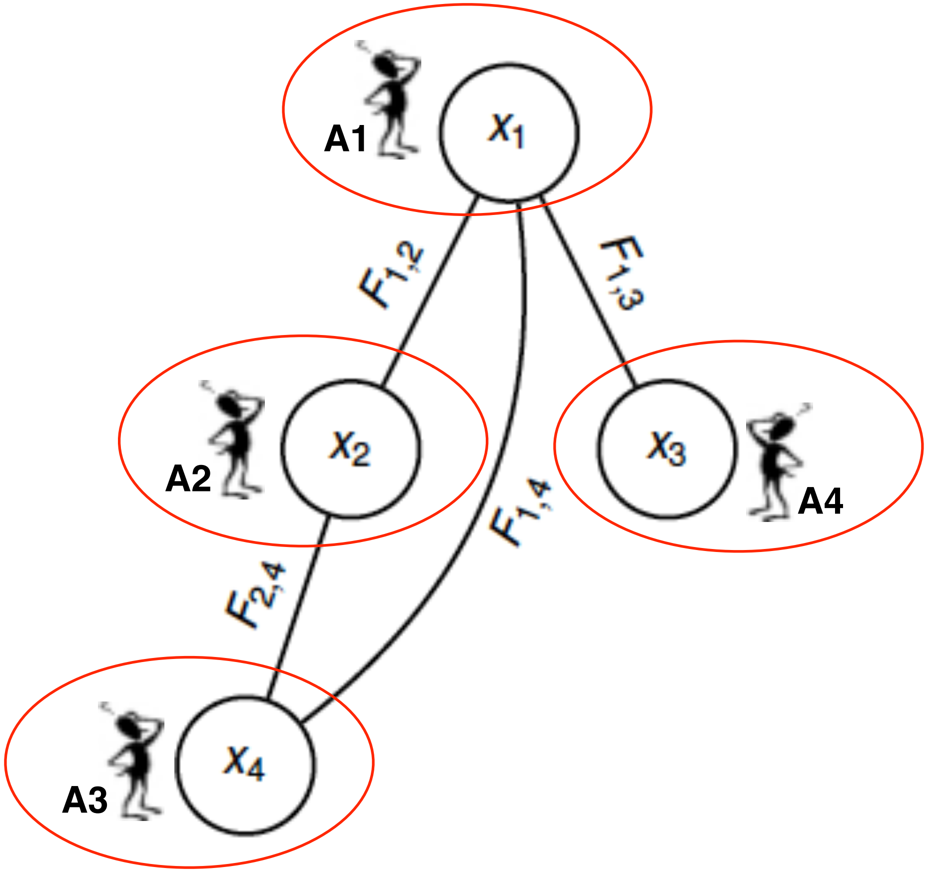
Different objectives, different problems

- **Constraint Satisfaction Problem (CSP)**
  - Objective: find an assignment for all the variables in the network that satisfies all constraints.
- **Constraint Optimization Problem (COP)**
  - Objective: find an assignment for all the variables in the network that satisfies all constraints and optimizes a global function.
  - Global function = aggregation (typically sum) of local functions. $F(x) = \sum_i F_i(x_i)$

Distributed Constraint Reasoning

When operating in a decentralized context:

- a set of agents control variables
- agents interact to find a solution to the constraint network



Distributed Constraint Reasoning

Two types of decentralized problems:

- **distributed CSP (DCSP)**
- **distributed COP (DCOP)**

Here, we focus on DCOPs.

Distributed Constraint Optimization Problem (DCOP)

A DCOP consists of a constraint network $\mathcal{N} = \langle X, D, C \rangle$ and a set of agents $A = \{A_1, \ldots, A_k\}$ where each agent:

- controls a subset of the variables $X_i \subseteq X$
- is only aware of constraints that involve variable it controls
- communicates only with its neighbours

Distributed Constraint Optimization Problem (DCOP)

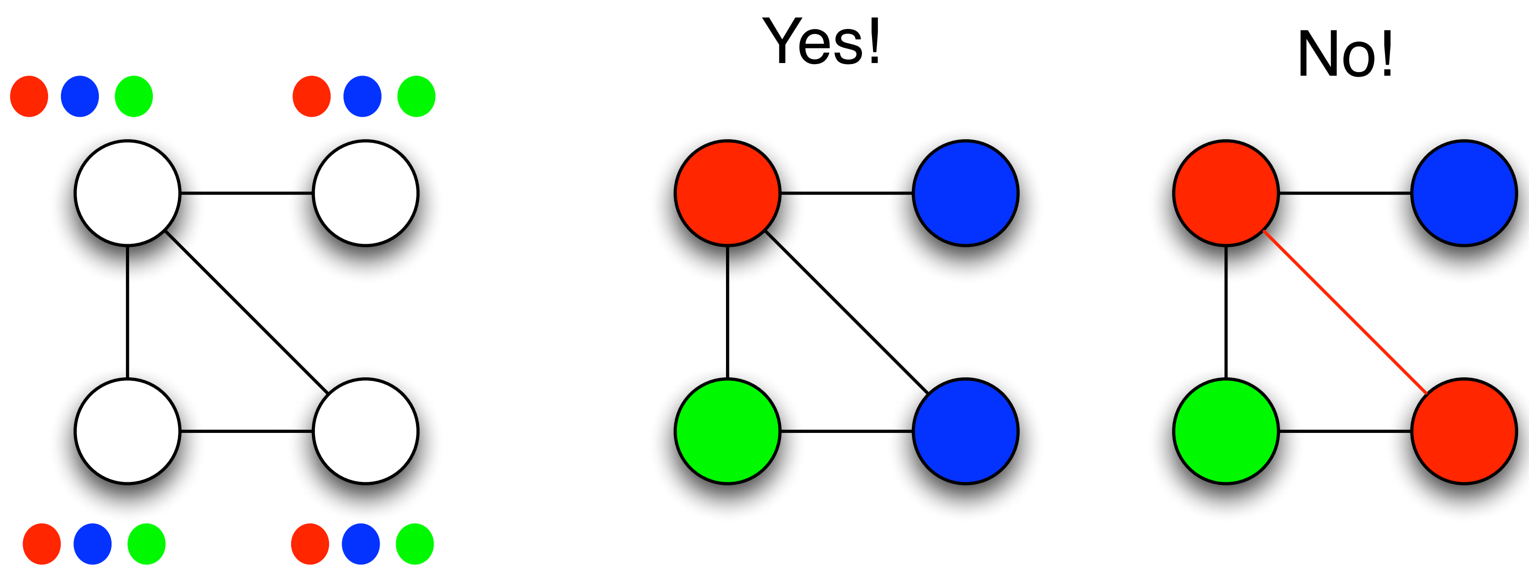
- Agents are assumed to be fully cooperative
  - Goal: find the assignment that optimizes the global function, not their local local utilities.
- Solving a COP is NP-Hard and DCOP is as hard as COP.

Graph coloring

- Popular benchmark
- Simple formulation
- Complexity controlled with few parameters:
  - Number of available colors
  - Number of nodes
  - Density ($\#nodes/\#constraints$)
- Many versions of the problem:
  - CSP, MaxCSP, COP

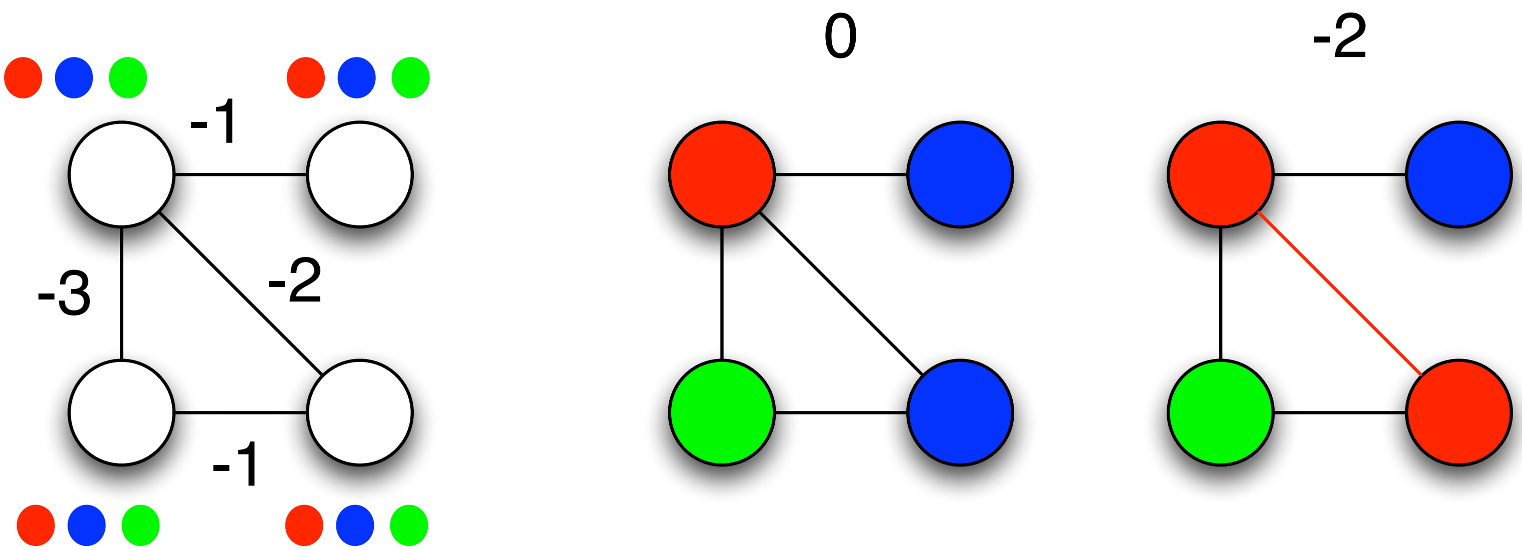
Graph coloring - CSP

- Nodes can take k colors
- Any two adjacent nodes should have different colors
  - If it happens this is a conflict



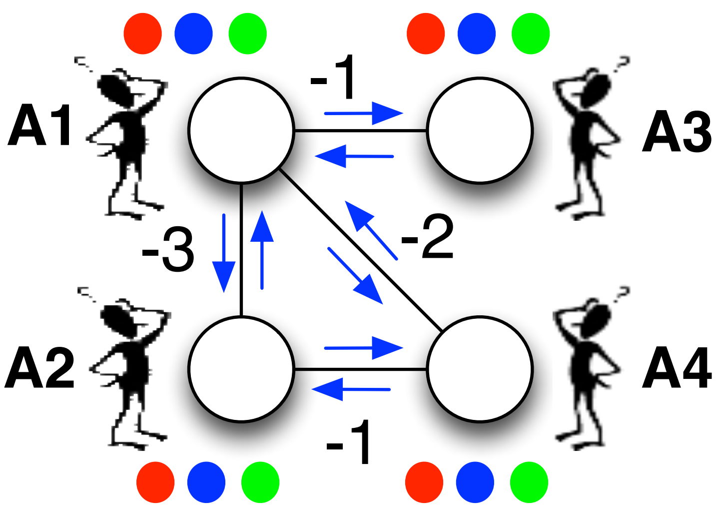
Graph coloring - COP

- Different weights to violated constraints
- Preferences for different colors



Graph coloring - DCOP

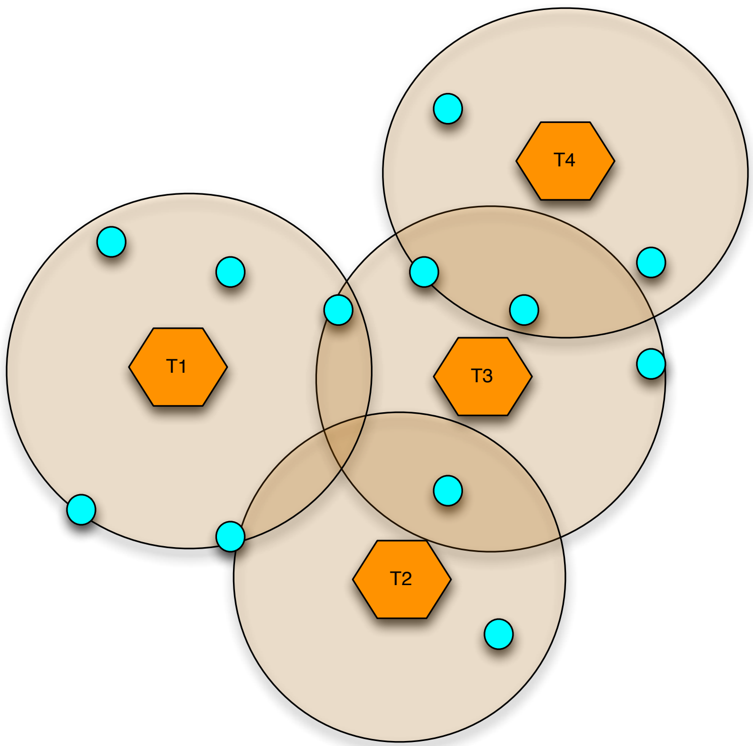
- Each node:
  - controlled by one agent
- Each agent:
  - Preferences for different colors
  - Communicates with its direct neighbours in the graph



- A1 and A2 exchange preferences and conflicts
- A3 and A4 do not communicate

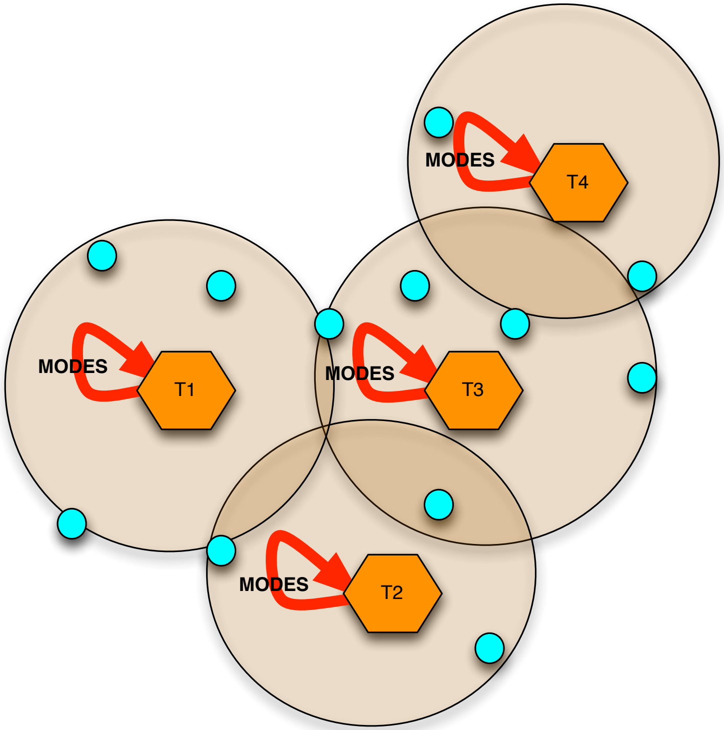
Target Tracking

A set of sensors tracking a set of targets in order to provide an accurate estimate of their positions.



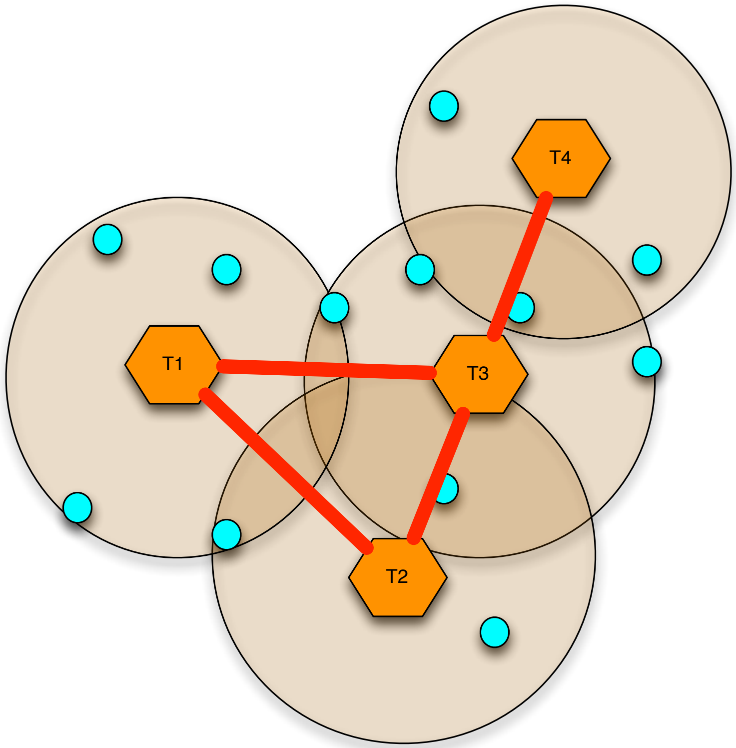
Target Tracking

Sensors can have different sensing modalities that impact on the accuracy of the estimation of the targets' positions.



Target Tracking

Collaboration among sensors is crucial to improve system performance



DCOP formalization for the target tracking problem

- Agents represent sensors
- Variables encode the different sensing modalities of each sensor
- Constraints
  - relate to a specific target
  - represent how sensor modalities impacts on the tracking performance
- Objective:
  - Maximize coverage of the environment
  - Provide accurate estimations of potentially dangerous targets

Complete Algorithms

- Always find an optimal solution
- Exhibit an exponentially increasing coordination overhead
- Very limited scalability on general problems.

Decentralised Complete Algorithms

Search-based

- Uses distributed search
- Exchange individual values
- Small messages but
  ... exponentially many

Representative: ADOPT [Modi et al., 2005]

Dynamic programming

- Uses distributed inference
- Exchange constraints
- Few messages but
  ... exponentially large

Representative: DPOP [Petcu and Faltings, 2005]

Algorithms

- ADOPT
  [presented by Federico Rosato]
- DPOP
- …

Why Approximate Algorithms

"Very often optimality in practical applications is not achievable"

Approximate algorithms

- Sacrify optimality in favor of computational and communication efficiency
- Well-suited for large scale distributed applications:
  - sensor networks
  - mobile robots

Centralized Local Greedy approaches

- Start from a random assignment for all the variables
- Do local moves if the new assignment improves the value (local gain)
- Local: changing the value of a small set of variables (in most case just one)
- The search stops when there is no local move that provides a positive gain, i.e., when the process reaches a local maximum.

Distributed Local Greedy approaches

When operating in a decentralized context:

- **Problem:** Out-of-date local knowledge
  - Assumption that other agents do not change their values
  - A greedy local move might be harmful/useless
- **Solution:**
  - Stochasticity on the decision to perform a move (DSA)
  - Coordination among neighbours on who is the agent that should move (MGM)

Decentralised greedy approaches

- Very little memory and computation
- Anytime behaviours
- Could result in very bad solutions
  - local maxima arbitrarily far from optimal

Algorithms

- MGM
  [presented by Matteo Bellusci]
- DSA
- *…*

Quality guarantees

So far, algorithms presented (DSA-1, MGM-1, Max-Sum) do not provide any guarantee on the quality of their solutions

- Quality highly dependent on many factors which cannot always be properly assessed before deploying the system.
- Particularly adverse behaviour on specific pathological instances.

Challenge:

- Quality assessment on approximate algorithms

Algorithms

- Bounded Max-Sum
- *…*