

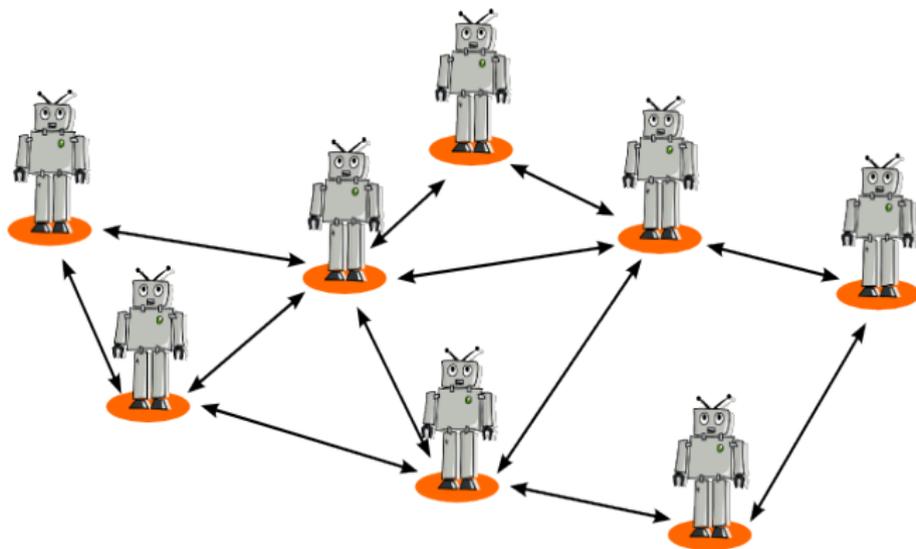
Fault-Tolerant Distributed Optimization

Lili Su, Arun Padakandla, Qiong Hu, Seyyed A. Fatemi,
Rehana Mahfuz, Vidyasagar Sadhu

CSOI-Data Science Workshop

May 27th, 2016

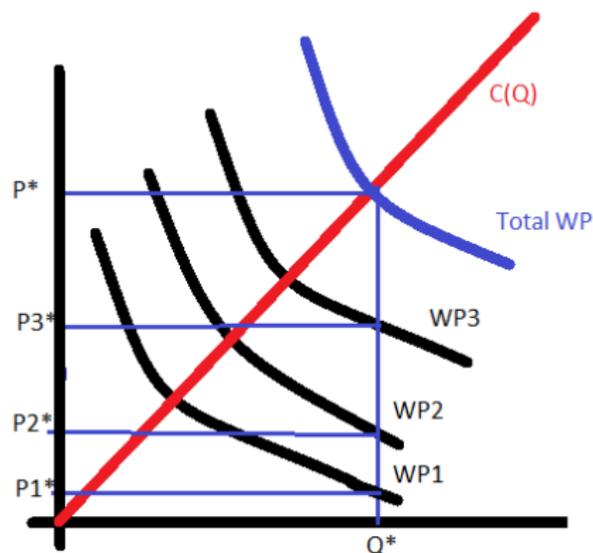
Multi-Agent Network



Distributed Optimization: design distributed algorithms to have the agents cooperatively achieve global tasks

Examples

- Power system: public good
- Sensor network
- Distributed machine learning
- ...



Existing work: all agents are cooperative

Our focus: the impact of non-cooperative agents

Problem Formulation

- n agents are connected by a network $G(\mathcal{V}, \mathcal{E})$, where $\mathcal{V} = \{1, \dots, n\}$;
- Local communication: single hop
- Each agent i has a local objective function $f_i(x) : \mathbb{R}^d \rightarrow \mathbb{R}$, which is known to agent i **only**;
- One agent is selfish: \tilde{j}
- Goal of selfish agent \tilde{j} : find the x to

$$\min f_{\tilde{j}}(x)$$

- Goal of the cooperative agents: find the x to

$$\min \frac{1}{|\mathcal{G}|} \sum_{i \in \mathcal{G}} f_i(x)$$

Specific Configuration:

- Completely connected graph.
- No memory across iterations.
- Single-hop communication.
- One simplified selfish agent.
- Exchange local estimates only without gradients

Question

Are we able to design distributed optimization algorithms/protocols such that the selfish agent can be detected and “removed”

Question: Are we able to design distributed optimization algorithms/protocols such that the selfish agent can be detected or “removed”?

- Yes.
 - Improve the algorithm
 - Consider more complex scenarios
 - arbitrary connected graph (instead of fully connected)
 - multihop communication
 - multiple selfish agents
 - “smart” selfish agents
- No.
 - Consider with memory
 - Consider hierarchical architecture

Applied directions: Study/improve algorithm for specific applications such as wireless sensor network, distributed machine learning, etc.