Distributed learning over dynamic networks with adversarial agents

Pooja Vyavahare
Assistant Professor
Department of Electrical Engineering
Indian Institute of Technology, Tirupati

June 28, 2019

Accepted in Fusion 2019

Joint work with

Dr. Lili Su
Postdoc
CSAIL, MIT

Dr. Nitin H. Vaidya
Department of Computer Science
Georgetown University

Outline of the talk

1. Motivation

Outline of the talk

1. Motivation
2. Problem statement

Outline of the talk

1. Motivation
2. Problem statement
3. Our result

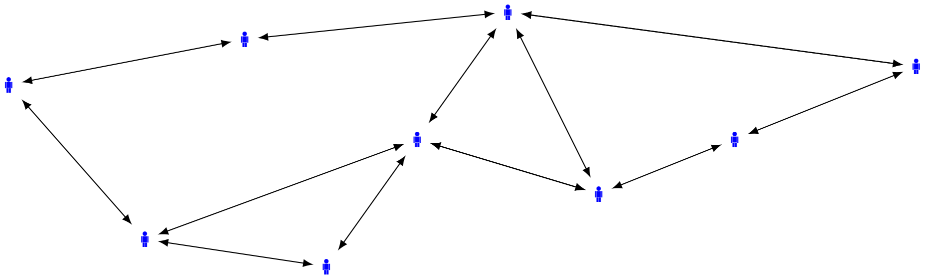
Outline of the talk

1. Motivation
2. Problem statement
3. Our result
4. Extension to time-varying networks

Outline of the talk

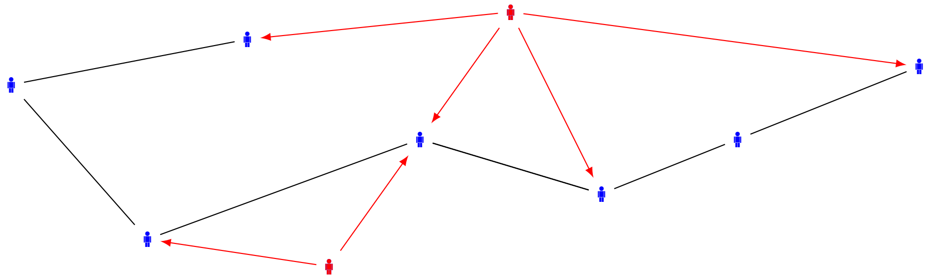
1. Motivation
2. Problem statement
3. Our result
4. Extension to time-varying networks
5. Discussion

Distributed (hypothesis testing) learning



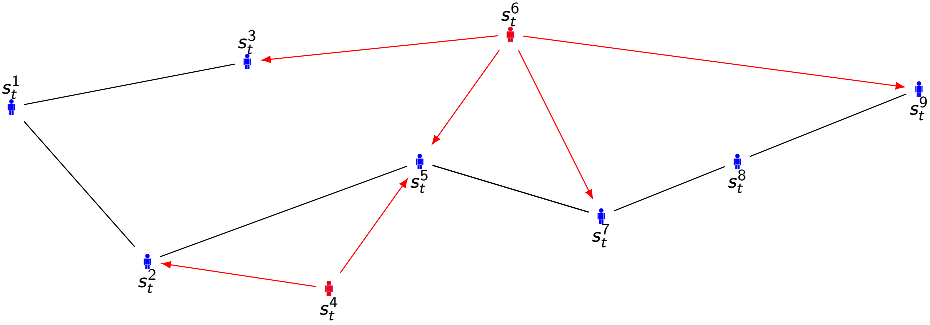
- Example: Social networks, sensor networks.
- Each agent wants to know the state of the environment.
- Example: everyone wants to know "best college to attend"

Distributed (hypothesis testing) learning



- There are some adversarial agents in the network.
- We consider Byzantine faults scenario.
- Adversarial agents may send arbitrary information and may not follow the specified algorithm.

Distributed (hypothesis testing) learning



- At every time slot t, each agent collects *partial* information about the state of the environment.
- There are m possible states of the environment but there is only one true state.
- Can agents learn the true state?

Problem formulation

- n agents are connected via a directed graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$

Problem formulation

- n agents are connected via a directed graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$
- Set of adversarial agents $\mathcal{F} \subset \mathcal{V}$; $|\mathcal{F}| = \phi \leq f$

Problem formulation

- n agents are connected via a directed graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$
- Set of adversarial agents $\mathcal{F} \subset \mathcal{V}$; $|\mathcal{F}| = \phi \leq f$
- $\mathcal{F}$ is not known to the good agents but f is known

Problem formulation

- n agents are connected via a directed graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$
- Set of adversarial agents $\mathcal{F} \subset \mathcal{V}$; $|\mathcal{F}| = \phi \leq f$
- $\mathcal{F}$ is not known to the good agents but f is known
- A finite set $\{\theta_1, \ldots, \theta_m\}$ of possible states; but only one true state θ^*

Problem formulation

- n agents are connected via a directed graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$
- Set of adversarial agents $\mathcal{F} \subset \mathcal{V}$; $|\mathcal{F}| = \phi \leq f$
- $\mathcal{F}$ is not known to the good agents but f is known
- A finite set $\{\theta_1, \ldots, \theta_m\}$ of possible states; but only one true state θ^*
- s_i^t is the signal observed by agent i at time t which comes from marginal distribution $l_i(s_i^t|\theta_j)$

Problem formulation

- n agents are connected via a directed graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$
- Set of adversarial agents $\mathcal{F} \subset \mathcal{V}$; $|\mathcal{F}| = \phi \leq f$
- $\mathcal{F}$ is not known to the good agents but f is known
- A finite set $\{\theta_1, \ldots, \theta_m\}$ of possible states; but only one true state θ^*
- s_i^t is the signal observed by agent i at time t which comes from marginal distribution $l_i(s_i^t|\theta_j)$
- Local communication: One hop neighbors

Problem formulation

- n agents are connected via a directed graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$
- Set of adversarial agents $\mathcal{F} \subset \mathcal{V}$; $|\mathcal{F}| = \phi \leq f$
- $\mathcal{F}$ is not known to the good agents but f is known
- A finite set $\{\theta_1, \ldots, \theta_m\}$ of possible states; but only one true state θ^*
- s_i^t is the signal observed by agent i at time t which comes from marginal distribution $l_i(s_i^t|\theta_j)$
- Local communication: One hop neighbors
- Each agent maintains a stochastic vector $\mu_i^t \in \mathbb{R}^m$ over all states

Problem formulation

- n agents are connected via a directed graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$
- Set of adversarial agents $\mathcal{F} \subset \mathcal{V}$; $|\mathcal{F}| = \phi \leq f$
- $\mathcal{F}$ is not known to the good agents but f is known
- A finite set $\{\theta_1, \ldots, \theta_m\}$ of possible states; but only one true state θ^*
- s_i^t is the signal observed by agent i at time t which comes from marginal distribution $l_i(s_i^t | \theta_j)$
- Local communication: One hop neighbors
- Each agent maintains a stochastic vector $\mu_i^t \in \mathbb{R}^m$ over all states
- Can all the good agents learn the true state?

Existing literature

- without adversarial agents

Existing literature

- without adversarial agents
  - Non-Bayesian learning by using distributed consensus algorithm first proposed by [Jadbabaie et al., 2012]

Existing literature

- without adversarial agents
  - Non-Bayesian learning by using distributed consensus algorithm first proposed by [Jadbabaie et al., 2012]
  - Many variants of system model; survey of results in [Nedić et al., 2016]

Existing literature

- without adversarial agents
  - Non-Bayesian learning by using distributed consensus algorithm first proposed by [Jadbabaie et al., 2012]
  - Many variants of system model; survey of results in [Nedić et al., 2016]
- with adversarial agents

Existing literature

- without adversarial agents
  - Non-Bayesian learning by using distributed consensus algorithm first proposed by [Jadbabaie et al., 2012]
  - Many variants of system model; survey of results in [Nedić et al., 2016]
- with adversarial agents
  - Using distributed consensus algorithm with faulty agents by [Su and Vaidya, 2016]

Existing literature

- without adversarial agents
  - Non-Bayesian learning by using distributed consensus algorithm first proposed by [Jadbabaie et al., 2012]
  - Many variants of system model; survey of results in [Nedić et al., 2016]
- with adversarial agents
  - Using distributed consensus algorithm with faulty agents by [Su and Vaidya, 2016]
- Existing literature network condition:

Existing literature

- without adversarial agents
  - Non-Bayesian learning by using distributed consensus algorithm first proposed by [Jadbabaie et al., 2012]
  - Many variants of system model; survey of results in [Nedić et al., 2016]
- with adversarial agents
  - Using distributed consensus algorithm with faulty agents by [Su and Vaidya, 2016]
- Existing literature network condition:

Condition (Network condition NC)

All the good agents in the network should be able to achieve vector consensus via iteratively sharing information with one hop neighbors.

Existing literature

- without adversarial agents
  - Non-Bayesian learning by using distributed consensus algorithm first proposed by [Jadbabaie et al., 2012]
  - Many variants of system model; survey of results in [Nedić et al., 2016]
- with adversarial agents
  - Using distributed consensus algorithm with faulty agents by [Su and Vaidya, 2016]
- Existing literature network condition:

Condition (Network condition NC)

All the good agents in the network should be able to achieve vector consensus via iteratively sharing information with one hop neighbors.

Condition (Identifiability condition IC)

A source component (strongly connected with no incoming edge) of good agents can estimate the true state.

Algorithm for fault-tolerant learning [Su and Vaidya, 2016]

- Transmit the vector $\log \mu_{t-1}^{i}$ on all outgoing links

Algorithm for fault-tolerant learning [Su and Vaidya, 2016]

- Transmit the vector $\log \mu_{t-1}^i$ on all outgoing links
- Find a point η_t^i in the convex hull of all good agents' vectors in neighborhood of i:

Algorithm for fault-tolerant learning [Su and Vaidya, 2016]

- Transmit the vector $\log \mu_{t-1}^{i}$ on all outgoing links
- Find a point η_t^i in the convex hull of all good agents' vectors in neighborhood of i:
  - Finding the Tverberg point of all subsets of incoming messages of size $(m+1)f+1$ Go to definition

Algorithm for fault-tolerant learning [Su and Vaidya, 2016]

- Transmit the vector $\log \mu_{t-1}^{i}$ on all outgoing links
- Find a point η_t^i in the convex hull of all good agents' vectors in neighborhood of i:
  - Finding the Tverberg point of all subsets of incoming messages of size $(m+1)f+1$ Go to definition
- Observe the signal s_t^i and update the marginal distribution

Algorithm for fault-tolerant learning [Su and Vaidya, 2016]

- Transmit the vector $\log \mu_{t-1}^{i}$ on all outgoing links
- Find a point η_t^i in the convex hull of all good agents' vectors in neighborhood of i:
  - Finding the Tverberg point of all subsets of incoming messages of size $(m+1)f+1$ ▸ Go to definition
- Observe the signal s_t^i and update the marginal distribution
- Update the vector μ_t^i based on the distribution of observed signal and η_t^i

Algorithm analysis

$$\eta_t^i(\theta) = \sum_{j=1}^{n-\phi} \mathbf{A}_{ij}[t] \log \mu_{t-1}^j(\theta) \quad \forall \theta$$

where $\mathbf{A}[t]$ is a random row stochastic matrix which depends on the cumulative observed signals till time t and the behavior of faulty agents.

Algorithm analysis

$$\eta_t^i(\theta) = \sum_{j=1}^{n-\phi} \mathbf{A}_{ij}[t] \log \mu_{t-1}^j(\theta) \quad \forall \theta$$

where $\mathbf{A}[t]$ is a random row stochastic matrix which depends on the cumulative observed signals till time t and the behavior of faulty agents.

$$\phi_t^i(\theta, \theta^*) := \log \frac{\mu_t^i(\theta)}{\mu_t^i(\theta^*)} = \underbrace{\sum_{j=1}^{n-\phi} \mathbf{A}_{ij}[t] \phi_{t-1}^j(\theta, \theta^*)}_{\text{Other agents' influence}} + \underbrace{\sum_{k=1}^{t} \mathcal{L}_k^i(\theta, \theta^*)}_{\text{Agent's private signal influence}}$$

where $\mathcal{L}_k^i(\theta, \theta^*) := \log \frac{l_i(s_k^i|\theta)}{l_i(s_k^i|\theta^*)}$

Analysis (cont...)

$$\phi_t^i(\theta, \theta^*) = \sum_{r=1}^{t} \sum_{j=1}^{n-\phi} \mathbf{\Phi}_{ij}(t, r+1) \sum_{k=1}^{r} \mathcal{L}_k^j(\theta, \theta^*)$$

Analysis (cont...)

$$\phi_t^i(\theta, \theta^*) = \sum_{r=1}^{t} \sum_{j=1}^{n-\phi} \mathbf{\Phi}_{ij}(t, r+1) \sum_{k=1}^{r} \mathcal{L}_k^j(\theta, \theta^*)$$

where $\mathbf{\Phi}(t, r) = \mathbf{A}[t] \ldots \mathbf{A}[r]$ for $r \in [1, t+1]$

Result [Su and Vaidya, 2016]

- Every good agent i will concentrate its vector on the true state θ^* almost surely, i.e., $\mu_t^i(\theta) \xrightarrow{\text{a.s.}} 0 \; \forall \theta \neq \theta^*$.

Result [Su and Vaidya, 2016]

- Every good agent i will concentrate its vector on the true state θ^* almost surely, i.e., $\mu_t^i(\theta) \xrightarrow{\text{a.s.}} 0 \ \forall \theta \neq \theta^*$.
- It is sufficient to show that $\phi_t^i(\theta, \theta^*) \xrightarrow{\text{a.s.}} -\infty$.

Result [Su and Vaidya, 2016]

- Every good agent i will concentrate its vector on the true state θ^* almost surely, i.e., $\mu_t^i(\theta) \xrightarrow{\text{a.s.}} 0 \ \forall \theta \neq \theta^*$.
- It is sufficient to show that $\phi_t^i(\theta, \theta^*) \xrightarrow{\text{a.s.}} -\infty$.
- When network of good agents is capable of achieving consensus (**Condition NC**) then,

$$\lim_{t \geq r, t \to \infty} \mathbf{\Phi}(t, r) = \mathbf{1}\pi(r).$$

Result [Su and Vaidya, 2016]

- Every good agent i will concentrate its vector on the true state θ^* almost surely, i.e., $\mu_t^i(\theta) \xrightarrow{\text{a.s.}} 0 \ \forall \theta \neq \theta^*$.
- It is sufficient to show that $\phi_t^i(\theta, \theta^*) \xrightarrow{\text{a.s.}} -\infty$.
- When network of good agents is capable of achieving consensus (**Condition NC**) then,

$$\lim_{t \geq r, t \to \infty} \mathbf{\Phi}(t, r) = \mathbf{1}\pi(r).$$

- Identifiability condition (**Condition IC**):

Result [Su and Vaidya, 2016]

- Every good agent i will concentrate its vector on the true state θ^* almost surely, i.e., $\mu_t^i(\theta) \xrightarrow{\text{a.s.}} 0 \ \forall \theta \neq \theta^*$.
- It is sufficient to show that $\phi_t^i(\theta, \theta^*) \xrightarrow{\text{a.s.}} -\infty$.
- When network of good agents is capable of achieving consensus (**Condition NC**) then,

$$\lim_{t \geq r, t \to \infty} \mathbf{\Phi}(t, r) = \mathbf{1}\pi(r).$$

- Identifiability condition (**Condition IC**):
  - In the graph defined by $\mathbf{A}[t]$ at time t, there is only one source component S.

Result [Su and Vaidya, 2016]

- Every good agent i will concentrate its vector on the true state θ^* almost surely, i.e., $\mu_t^i(\theta) \xrightarrow{\text{a.s.}} 0 \ \forall \theta \neq \theta^*$.
- It is sufficient to show that $\phi_t^i(\theta, \theta^*) \xrightarrow{\text{a.s.}} -\infty$.
- When network of good agents is capable of achieving consensus (**Condition NC**) then,

$$\lim_{t \geq r, t \to \infty} \mathbf{\Phi}(t, r) = \mathbf{1}\pi(r).$$

- Identifiability condition (**Condition IC**):
  - In the graph defined by $\mathbf{A}[t]$ at time t, there is only one source component S.
  - S has a path to every good agent in the network.

Result [Su and Vaidya, 2016]

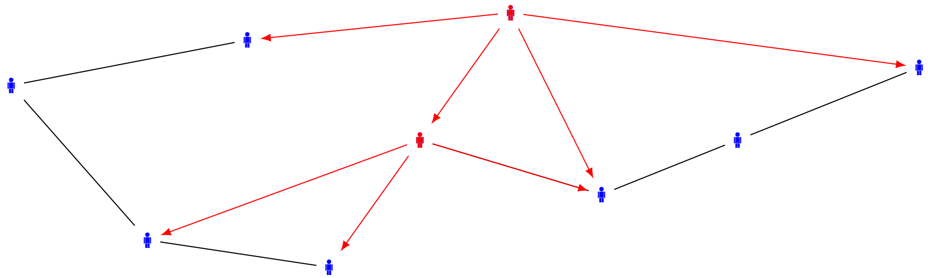
- Every good agent i will concentrate its vector on the true state θ^* almost surely, i.e., $\mu_t^i(\theta) \xrightarrow{\text{a.s.}} 0 \ \forall \theta \neq \theta^*$.
- It is sufficient to show that $\phi_t^i(\theta, \theta^*) \xrightarrow{\text{a.s.}} -\infty$.
- When network of good agents is capable of achieving consensus (**Condition NC**) then,

$$\lim_{t \geq r, t \to \infty} \mathbf{\Phi}(t, r) = \mathbf{1}\pi(r).$$

- Identifiability condition (**Condition IC**):
  - In the graph defined by $\mathbf{A}[t]$ at time t, there is only one source component S.
  - S has a path to every good agent in the network.
  - Agents in S can collectively estimate the true state.

Relaxing the network condition

- The network of good agents may not be able to achieve distributed consensus.



Relaxed conditions

- After removing the faulty agents, the network may have more than one connected component.

Relaxed conditions

- After removing the faulty agents, the network may have more than one connected component.
- After removing the faulty agents, the network may be weakly connected and each agent might be receiving information from disjoint components.

Relaxed conditions

- After removing the faulty agents, the network may have more than one connected component.
- After removing the faulty agents, the network may be weakly connected and each agent might be receiving information from disjoint components.
- $$\lim_{t \geq r, t \to \infty} \mathbf{\Phi}(t, r) \neq \mathbf{1}\pi(r).$$

Relaxed conditions

- After removing the faulty agents, the network may have more than one connected component.
- After removing the faulty agents, the network may be weakly connected and each agent might be receiving information from disjoint components.
-
$$\lim_{t \geq r, t \to \infty} \mathbf{\Phi}(t, r) \neq \mathbf{1}\pi(r).$$
- Relaxed IC:

Relaxed conditions

- After removing the faulty agents, the network may have more than one connected component.
- After removing the faulty agents, the network may be weakly connected and each agent might be receiving information from disjoint components.
-
$$\lim_{t\geq r, t\to\infty} \mathbf{\Phi}(t,r) \neq \mathbf{1}\pi(r).$$
- Relaxed IC:
  - Every graph corresponding to $\mathbf{A}[t]$ may have one or more source components.

Relaxed conditions

- After removing the faulty agents, the network may have more than one connected component.
- After removing the faulty agents, the network may be weakly connected and each agent might be receiving information from disjoint components.
- $$\lim_{t \geq r, t \to \infty} \mathbf{\Phi}(t, r) \neq \mathbf{1}\pi(r).$$
- Relaxed IC:
  - Every graph corresponding to $\mathbf{A}[t]$ may have one or more source components.
  - Every good agent has a path from at least one such component.

Relaxed conditions

- After removing the faulty agents, the network may have more than one connected component.
- After removing the faulty agents, the network may be weakly connected and each agent might be receiving information from disjoint components.
- $$\lim_{t \geq r, t \to \infty} \mathbf{\Phi}(t, r) \neq \mathbf{1}\pi(r).$$
- Relaxed IC:
  - Every graph corresponding to $\mathbf{A}[t]$ may have one or more source components.
  - Every good agent has a path from at least one such component.
  - Agents of each component S can collectively estimate the true state.

Proof outline

1. Under relaxed network condition, each good agent in the graph corresponding to $\mathbf{A}[t]$ has at least $(m+1)f+1$ incoming edges.

Proof outline

1. Under relaxed network condition, each good agent in the graph corresponding to $\mathbf{A}[t]$ has at least $(m+1)f+1$ incoming edges.
2. For every good agent i, there exists a source component S_i such that

$$\mathbf{\Phi}_{ij}(t, r+1) \geq \beta > 0 \ \ \forall j \in S_i, t-r \geq \nu$$

Proof outline

1. Under relaxed network condition, each good agent in the graph corresponding to $\mathbf{A}[t]$ has at least $(m+1)f+1$ incoming edges.
2. For every good agent i, there exists a source component S_i such that

$$\mathbf{\Phi}_{ij}(t, r+1) \geq \beta > 0 \quad \forall j \in S_i, t - r \geq \nu$$

3. $$\sum_{r=1}^{t} r \sum_{j=1}^{n-\phi} \mathbf{\Phi}_{ij}(t, r+1) H_j(\theta, \theta^*) \leq -Ct^2$$

   where C is a constant and $H_j(\theta, \theta^*) = -D(l_j(.|\theta^*)||l_j(.|\theta))$ is the negative of the KL divergence between states' marginal distribution.

Proof outline

1. Under relaxed network condition, each good agent in the graph corresponding to $\mathbf{A}[t]$ has at least $(m+1)f+1$ incoming edges.
2. For every good agent i, there exists a source component S_i such that

$$\mathbf{\Phi}_{ij}(t, r+1) \geq \beta > 0 \quad \forall j \in S_i, t-r \geq \nu$$

3. $$\sum_{r=1}^{t} r \sum_{j=1}^{n-\phi} \mathbf{\Phi}_{ij}(t, r+1) H_j(\theta, \theta^*) \leq -Ct^2$$

   where C is a constant and $H_j(\theta, \theta^*) = -D(l_j(.|\theta^*)||l_j(.|\theta))$ is the negative of the KL divergence between states' marginal distribution.

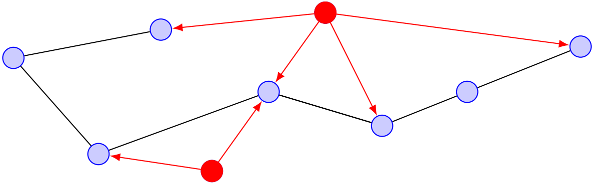
4. $$\phi_t^i(\theta, \theta^*) = \sum_{r=1}^{t} \sum_{j=1}^{n-\phi} \mathbf{\Phi}_{ij}(t, r+1) \sum_{k=1}^{r} \mathcal{L}_k^j(\theta, \theta^*) \xrightarrow{\text{a.s.}} -\infty$$

Our result

Theorem

Under **relaxed identifiability and connectivity conditions for graph of $\mathbf{A}[t]$**, *for the distributed algorithm every agent i will concentrate its vector on the true state θ^* almost surely, i.e.,* $\mu_t^i(\theta) \xrightarrow{\text{a.s.}} 0 \ \forall \theta \neq \theta^*$.

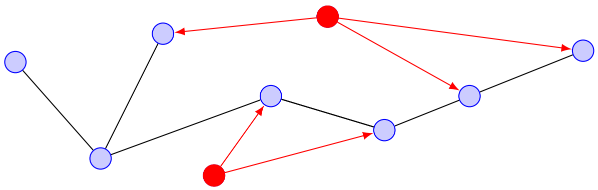
Learning with time-varying networks



Network at time t_1

- A sequence of graphs $\{\mathcal{G}_t | t = 1, 2, \ldots\}$

Learning with time-varying networks



At time t_2

- A sequence of graphs $\{\mathcal{G}_t | t = 1, 2, \ldots\}$
- Set of faulty agents $\mathcal{F}$ is fixed across time.

Relaxed conditions: NC and IC

- Source component and connectivity conditions defined on union of $B < \infty$ consecutive graphs.

Relaxed conditions: NC and IC

- Source component and connectivity conditions defined on union of $B < \infty$ consecutive graphs.
- Every source component in any *union of* $B < \infty$ consecutive graphs of $\mathbf{A}[t]$ can estimate the true state.

Relaxed conditions: NC and IC

- Source component and connectivity conditions defined on union of $B < \infty$ consecutive graphs.
- Every source component in any *union of* $B < \infty$ consecutive graphs of $\mathbf{A}[t]$ can estimate the true state.

Relaxed conditions: NC and IC

- Source component and connectivity conditions defined on union of $B < \infty$ consecutive graphs.
- Every source component in any *union of* $B < \infty$ consecutive graphs of $\mathbf{A}[t]$ can estimate the true state.

Theorem

Under **identifiability of true state by union of a finite sequence of graphs corresponding to $\mathbf{A}[t]$**, *for the distributed algorithm every agent i will concentrate its vector on the true state θ^* almost surely, i.e.,* $\mu_t^i(\theta) \xrightarrow{\text{a.s.}} 0 \ \forall \theta \neq \theta^*$.

Open questions

- Necessary condition for network topology to learn true state.

Open questions

- Necessary condition for network topology to learn true state.
- Convergence time of the algorithm.

Open questions

- Necessary condition for network topology to learn true state.
- Convergence time of the algorithm.
- Asynchronous systems with faulty agents.

Open questions

- Necessary condition for network topology to learn true state.
- Convergence time of the algorithm.
- Asynchronous systems with faulty agents.
- Distributed learning with communication errors.

Bibliography I

Jadbabaie, A., Molavi, P., Sandroni, A., and Tahbaz-Salehi, A. (2012).
Non-baysian social learning.
Games and Economic Behavior, 76:210–225.

Nedić, A., Olshevsky, A., and Uribe, C. A. (2016).
A tutorial on distributed (non-bayesian) learning: Problem, algorithms and results.
In *55th IEEE Conf. on Decision and Control*, pages 6795–6801.

Su, L. and Vaidya, N. H. (2016).
Non-bayesian learning in the presence of byzantine agents.
In *International Symposium on Distributed Computing*, pages 414–427. Springer.

Thank You

Teverberg point

Theorem (Tverberg Theorem)

For any integer $f \geq 1$, for every multiset Y containing at least $(m+1)f+1$ vectors in $\mathbb{R}^m$, there exists a partition $Y_1, \ldots, Y_{f+1}$ of Y into $f+1$ nonempty multisets such that $\cap_{i=1}^{f+1} \mathcal{H}(Y_i) \neq \emptyset$.

Teverberg point

Theorem (Tverberg Theorem)

For any integer $f \geq 1$, for every multiset Y containing at least $(m+1)f+1$ vectors in $\mathbb{R}^m$, there exists a partition $Y_1, \ldots, Y_{f+1}$ of Y into $f+1$ nonempty multisets such that $\cap_{i=1}^{f+1} \mathcal{H}(Y_i) \neq \emptyset$.

- Points in $\cap_{i=1}^{f+1} \mathcal{H}(Y_i)$ are called Tverberg point.

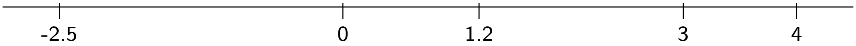
Go to algorithm

Teverberg point

Theorem (Tverberg Theorem)

For any integer $f \geq 1$, for every multiset Y containing at least $(m+1)f+1$ vectors in $\mathbb{R}^m$, there exists a partition $Y_1, \ldots, Y_{f+1}$ of Y into $f+1$ nonempty multisets such that $\cap_{i=1}^{f+1} \mathcal{H}(Y_i) \neq \emptyset$.

- Points in $\cap_{i=1}^{f+1} \mathcal{H}(Y_i)$ are called Tverberg point.
- Example for one dimensional case $m = 1$ and single faulty agent $f = 1$:



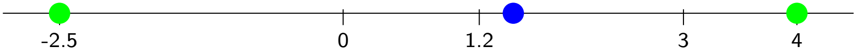
▸ Go to algorithm

Teverberg point

Theorem (Tverberg Theorem)

For any integer $f \geq 1$, for every multiset Y containing at least $(m+1)f+1$ vectors in $\mathbb{R}^m$, there exists a partition $Y_1, \ldots, Y_{f+1}$ of Y into $f+1$ nonempty multisets such that $\cap_{i=1}^{f+1}\mathcal{H}(Y_i) \neq \emptyset$.

- Points in $\cap_{i=1}^{f+1}\mathcal{H}(Y_i)$ are called Tverberg point.
- Example for one dimensional case $m = 1$ and single faulty agent $f = 1$:



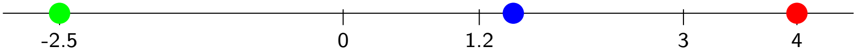
Go to algorithm

Teverberg point

Theorem (Tverberg Theorem)

For any integer $f \geq 1$, for every multiset Y containing at least $(m+1)f+1$ vectors in $\mathbb{R}^m$, there exists a partition $Y_1, \ldots, Y_{f+1}$ of Y into $f+1$ nonempty multisets such that $\cap_{i=1}^{f+1} \mathcal{H}(Y_i) \neq \emptyset$.

- Points in $\cap_{i=1}^{f+1} \mathcal{H}(Y_i)$ are called Tverberg point.
- Example for one dimensional case $m = 1$ and single faulty agent $f = 1$:



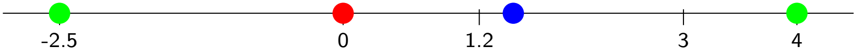
Go to algorithm

Teverberg point

Theorem (Tverberg Theorem)

For any integer $f \geq 1$, for every multiset Y containing at least $(m+1)f+1$ vectors in $\mathbb{R}^m$, there exists a partition $Y_1, \ldots, Y_{f+1}$ of Y into $f+1$ nonempty multisets such that $\cap_{i=1}^{f+1} \mathcal{H}(Y_i) \neq \emptyset$.

- Points in $\cap_{i=1}^{f+1} \mathcal{H}(Y_i)$ are called Tverberg point.
- Example for one dimensional case $m = 1$ and single faulty agent $f = 1$:



▸ Go to algorithm