Reputation-based Resilient Distributed Algorithms

Daniel Silvestre

Seminário, Universidade Coimbra
Coimbra

October 18th 2019

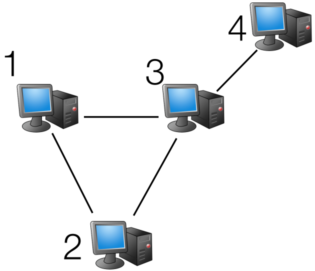
Outline

Motivation

- Compute the average of samples - a set of sensors produces measurements that require some gathering approach.
- No centralized supervision - there does not exist an entity establishing the quality of the measurements.
- Lightweight removal of attackers - proposed solutions need to require a low computational cost.

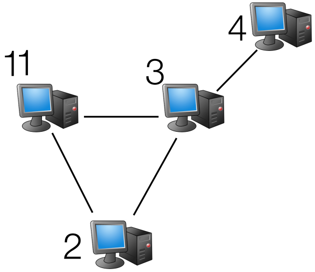
Example

- Take the network given by the following graph and state at time instant k, $x(k) = [1, 2, 3, 4]^\intercal$
- The consensus algorithm will asymptotically converge to all the nodes having $x_{\text{av}} = 2.5$
- If an attacker controlling one node in one time instant resets the state to $\tilde{x}_1(k) = x_1(k) + m(c - x_{\text{av}})$
- In this example $c = 5$, but the attacker can force the network to converge to any constant c!



Example

- Take the network given by the following graph and state at time instant k, $x(k) = [1, 2, 3, 4]^\intercal$
- The consensus algorithm will asymptotically converge to all the nodes having $x_{\text{av}} = 2.5$
- If an attacker controlling one node in one time instant resets the state to $\tilde{x}_1(k) = x_1(k) + m(c - x_{\text{av}})$
- In this example $c = 5$, but the attacker can force the network to converge to any constant c!



Traditional Solutions

1. Centralized approach
2. Use of an additional layer that computes worst-case values for the measurements [1][2].
3. Discard measurements from the computation that are at the extremes [3].

Traditional Solutions

1. *High computational costs and need for a central processing unit.*
2. Use of an additional layer that computes worst-case values for the measurements [1][2].
3. Discard measurements from the computation that are at the extremes [3].

Traditional Solutions

1. *High computational costs and need for a central processing unit.*
2. Use of an additional layer that computes worst-case values for the measurements [1][2].
3. Discard measurements from the computation that are at the extremes [3].

Traditional Solutions

1. *High computational costs and need for a central processing unit.*
2. *High computational costs, requires a dynamical model.*
3. Discard measurements from the computation that are at the extremes [3].

Traditional Solutions

1. *High computational costs and need for a central processing unit.*
2. *High computational costs, requires a dynamical model.*
3. Discard measurements from the computation that are at the extremes [3].

Traditional Solutions

1. *High computational costs and need for a central processing unit.*
2. *High computational costs, requires a dynamical model.*
3. *Low computational costs but cannot distinguish relevant data.*

Intuition behind Reputation-based approach

- In our daily lives, we are drawn towards decisions based on reputation.
- Reputation is a metric that assigns a large value if data is *similar*.
- Is suitable for distributed computation as processors can use the reputation to weight the data.

Intuition behind Reputation-based approach

- In our daily lives, we are drawn towards decisions based on reputation.
- Reputation is a metric that assigns a large value if data is *similar*.
- Is suitable for distributed computation as processors can use the reputation to weight the data.

Intuition behind Reputation-based approach

- In our daily lives, we are drawn towards decisions based on reputation.
- Reputation is a metric that assigns a large value if data is *similar*.
- Is suitable for distributed computation as processors can use the reputation to weight the data.

Problem Statement

- Consider an iterative algorithm to compute some weighted average of the measurements:

$$x_i^{(k+1)} = \sum_{j \in \mathcal{N}_i} c_{ij}^{(k)} x_j^{(k)} \Big/ \sum_{j \in \mathcal{N}_i} c_{ij}^{(k)} \tag{1}$$

- Can we compute weights $c_{ij}^{(k)}$ such that:

$$\forall i \in \mathcal{N} \setminus \mathcal{A} : x_i^{(k+1)} \to \bar{x} \tag{2}$$

Resilient Consensus Problem

Can we design an algorithm that assigns weights based on reputation?

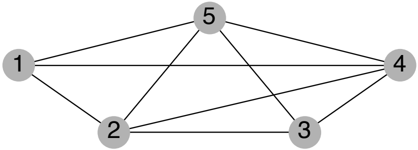
Reputation-based Consensus Algorithm

- We use work on reputation systems with some mathematical adjustments to compute $c_{ij}^{(k)}$;
- It has $\mathcal{O}\left(l^2 i\right)$ complexity for the maximum degree l and i iterations;
- No false positives;
- Has a better behavior than discarding extreme values in simulation.

Simulation - no attack

The figure illustrates the used topology.

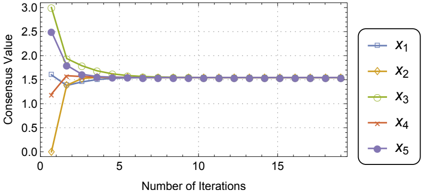
- Without attacks the algorithm works as the non-resilient version.
- In the absence of attackers, reputations eventually equal 1.
- The algorithm converges to a consensus state.



Simulation - no attack

The figure illustrates the used topology.

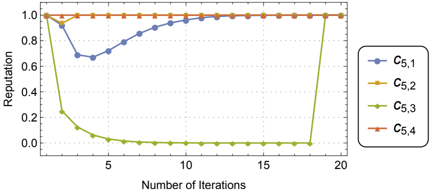
- Without attacks the algorithm works as the non-resilient version.
- In the absence of attackers, reputations eventually equal 1.
- The algorithm converges to a consensus state.



Simulation - no attack

The figure illustrates the used topology.

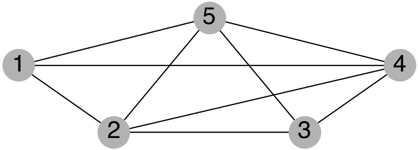
- Without attacks the algorithm works as the non-resilient version.
- In the absence of attackers, reputations eventually equal 1.
- The algorithm converges to a consensus state.



Simulation - single attacker with value close to consensus

Same topology.

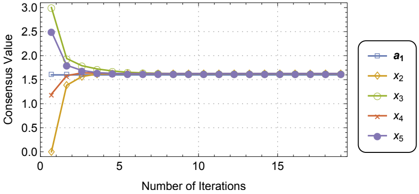
- At first sight it seems unsuccessful.
- By zooming in, we see that the nodes exclude the close value of the attacker.
- By looking at the reputation of node 5, the correct identification is clear.



Simulation - single attacker with value close to consensus

Same topology.

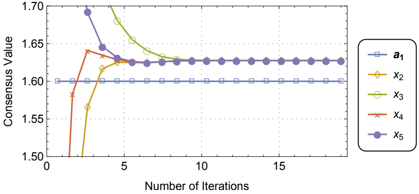
- At first sight it seems unsuccessful.
- By zooming in, we see that the nodes exclude the close value of the attacker.
- By looking at the reputation of node 5, the correct identification is clear.



Simulation - single attacker with value close to consensus

Same topology.

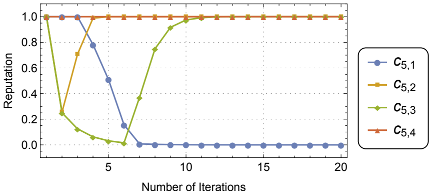
- At first sight it seems unsuccessful.
- By zooming in, we see that the nodes exclude the close value of the attacker.
- By looking at the reputation of node 5, the correct identification is clear.



Simulation - single attacker with value close to consensus

Same topology.

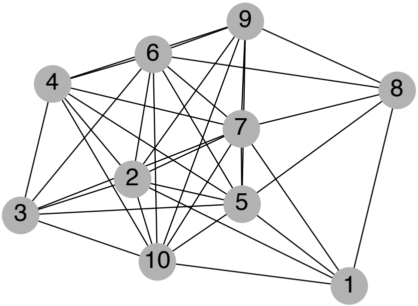
- At first sight it seems unsuccessful.
- By zooming in, we see that the nodes exclude the close value of the attacker.
- By looking at the reputation of node 5, the correct identification is clear.



Simulation - two attackers with different values

More complex topology.

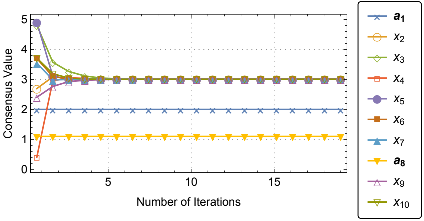
- It works when the two attackers share smaller values.
- But also when one is larger and the other smaller than the consensus value.



Simulation - two attackers with different values

More complex topology.

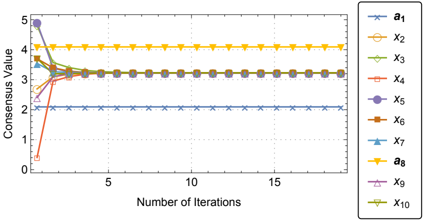
- It works when the two attackers share smaller values.
- But also when one is larger and the other smaller than the consensus value.



Simulation - two attackers with different values

More complex topology.

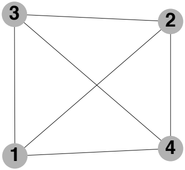
- It works when the two attackers share smaller values.
- But also when one is larger and the other smaller than the consensus value.



Simulation - Reputation-based vs State-of-the-art (1/2)

State-of-the-art consists in removing large and small values. We used the topology in the figure.

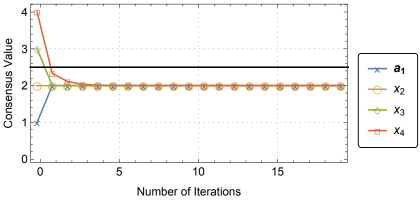
- Using MSR algorithm results in a shift regarding the final value.
- Using Reputation-based improves on minimizing the effect of the attacker.



Simulation - Reputation-based vs State-of-the-art (1/2)

State-of-the-art consists in removing large and small values. We used the topology in the figure.

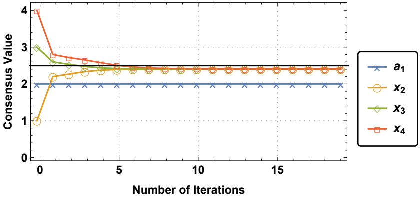
- Using MSR algorithm results in a shift regarding the final value.
- Using Reputation-based improves on minimizing the effect of the attacker.



Simulation - Reputation-based vs State-of-the-art (1/2)

State-of-the-art consists in removing large and small values. We used the topology in the figure.

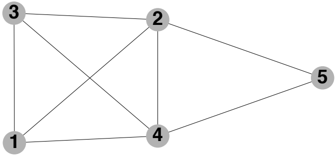
- Using MSR algorithm results in a shift regarding the final value.
- Using Reputation-based improves on minimizing the effect of the attacker.



Simulation - Reputation-based vs State-of-the-art (2/2)

Using a topology that is not fully connected.

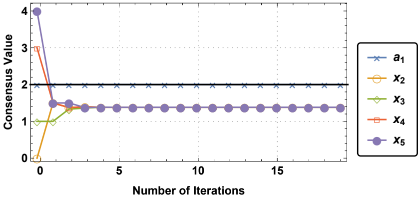
- Using MSR algorithm still results in a large shift in the steady-state.
- Using Reputation-based achieves a very small error.
- Even if the attacker uses a variable signal, using Reputation-based is still able to eliminate the influence.



Simulation - Reputation-based vs State-of-the-art (2/2)

Using a topology that is not fully connected.

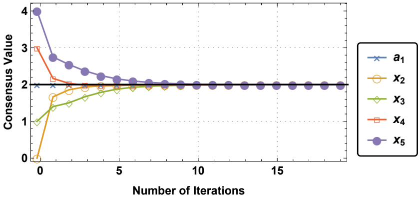
- Using MSR algorithm still results in a large shift in the steady-state.
- Using Reputation-based achieves a very small error.
- Even if the attacker uses a variable signal, using Reputation-based is still able to eliminate the influence.



Simulation - Reputation-based vs State-of-the-art (2/2)

Using a topology that is not fully connected.

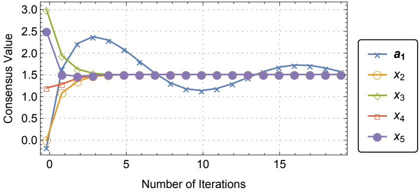
- Using MSR algorithm still results in a large shift in the steady-state.
- Using Reputation-based achieves a very small error.
- Even if the attacker uses a variable signal, using Reputation-based is still able to eliminate the influence.



Simulation - Reputation-based vs State-of-the-art (2/2)

Using a topology that is not fully connected.

- Using MSR algorithm still results in a large shift in the steady-state.
- Using Reputation-based achieves a very small error.
- Even if the attacker uses a variable signal, using Reputation-based is still able to eliminate the influence.



Concluding Remarks

Contributions:

- We present a Reputation-based algorithm that measures the quality of the values provided by each agent.
- The algorithm is lightweight with worst-case complexity dependent on the largest number of neighbors.

- Currently, this work is under review. In the journal version, we provide theoretical guarantees associated with the correctness and correct detection.
- We have also tested other networks (static and dynamics), type of attacker signal, number of attackers, among others.

References

D. Silvestre, J. P. Hespanha, and C. Silvestre, "Broadcast and gossip stochastic average consensus algorithms in directed topologies," *IEEE Transactions on Control of Network Systems*, vol. 6, no. 2, pp. 474–486, 2019. DOI: 10.1109/TCNS.2018.2839341.

D. Silvestre, P. Rosa, J. P. Hespanha, and C. Silvestre, "Stochastic and deterministic fault detection for randomized gossip algorithms," *Automatica*, vol. 78, pp. 46 –60, 2017, ISSN: 0005-1098. DOI: 10.1016/j.automatica.2016.12.011.

H. Y. Öksüz and M. Akar, "Distributed resilient consensus: A non-parametric approach," *Transactions of the Institute of Measurement and Control*, vol. 41, no. 8, pp. 2124–2134, 2019. DOI: 10.1177/0142331218785673.

The end

- Thank you for your time.

Reputation-based Resilient Distributed Algorithms

Daniel Silvestre

Seminário, Universidade Coimbra
Coimbra

October 18th 2019