

Wave algorithms



A **decide event** is a special internal event.
agreed upon all processes

(e.g. a lockstep for building a synchronous network)

In a **wave algorithm**, each computation (also called wave) satisfies the following properties:

- ▶ **termination**: it is finite;
- ▶ **decision**: it contains one or more decide events; and
- ▶ **dependence**: for each decide event e and process p ,
 $f \prec e$ for an event f at p .
(so a wave can only terminate if all processes have a say)

Wave algorithms - Example

In the *ring algorithm*, the **initiator** sends a **token**, which is passed on by all other processes.

The initiator decides after the token has returned.

Question: For each process, which event is causally before the decide event?

The ring algorithm is an example of a **traversal** algorithm.

Tarry's algorithm (from 1895)

Consider an **undirected** network.

- R1 A process never forwards the token through the same channel twice.
- R2 A process only forwards the token to its parent when there is no other option.

The token travels through each channel both ways, and finally ends up at the initiator.

Message complexity: $2E$ messages

Time complexity: $\leq 2E$ time units



Gaston Tarry



Tarry's algorithm (from 1895)

Consider an **undirected** network.

- R1 A process never forwards the token through the same channel twice.
- R2 A process only forwards the token to its parent when there is no other option.

The token travels through each channel both ways, and finally ends up at the initiator.

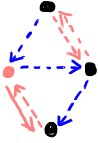
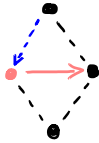
Message complexity: $2E$ messages

Time complexity: $\leq 2E$ time units



Gaston Tarry

initiator



decide

Tree algorithm

The tree algorithm is a **decentralized wave** algorithm for **undirected, acyclic** networks.



The local algorithm at a process p :

- ▶ p waits until it received messages from all neighbors except one, which becomes its *parent*.

Then it sends a message to its parent.

- ▶ If p receives a *message* from its parent, it **decides**.

It sends the decision to all neighbors except its parent.

- ▶ If p receives a *decision* from its parent, it passes it on to all other neighbors.

Always *two* (neighboring) processes decide.

Tree algorithm

The tree algorithm is a **decentralized wave** algorithm for **undirected, acyclic** networks.



The local algorithm at a process p :

- ▶ p waits until it received messages from all neighbors except one, which becomes its *parent*.

Then it sends a message to its parent.

- ▶ If p receives a *message* from its parent, it **decides**.
It sends the decision to all neighbors except its parent.
- ▶ If p receives a *decision* from its parent, it passes it on to all other neighbors.

Always *two* (neighboring) processes decide.

Tree algorithm

The tree algorithm is a **decentralized wave** algorithm for **undirected, acyclic** networks.



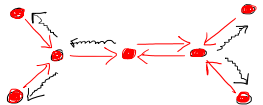
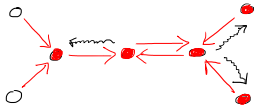
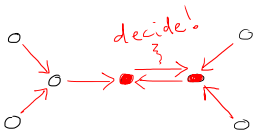
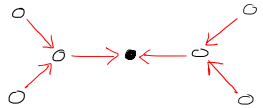
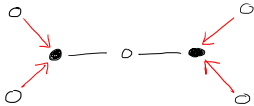
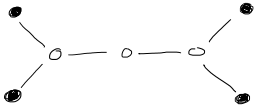
The local algorithm at a process p :

- ▶ p waits until it received messages from all neighbors except one, which becomes its *parent*.

Then it sends a message to its parent.

- ▶ If p receives a *message* from its parent, it **decides**.
It sends the decision to all neighbors except its parent.
- ▶ If p receives a *decision* from its parent, it passes it on to all other neighbors.

Always *two* (neighboring) processes decide.



What happens if the tree algorithm is applied to a network containing a cycle?

Apply the tree algorithm to compute the size of an undirected, acyclic network.

What happens if the tree algorithm is applied to a network containing a cycle?

Apply the tree algorithm to compute the size of an undirected, acyclic network.

Echo algorithm

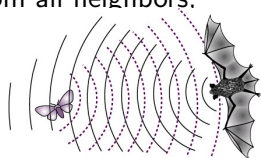
The echo algorithm is a **centralized wave** algorithm for **undirected** networks.

- ▶ The **initiator** sends a message to all neighbors.
- ▶ When a **noninitiator** receives a message for the first time, it makes the sender its *parent*.

Then it sends a message to all neighbors except its parent.

- ▶ When a **noninitiator** has received a message from all neighbors, it sends a message to its parent.
- ▶ When the **initiator** has received a message from all neighbors, it *decides*.

Message complexity: $2E$ messages



Echo algorithm

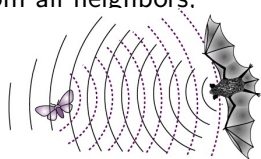
The echo algorithm is a **centralized wave** algorithm for **undirected** networks.

- ▶ The **initiator** sends a message to all neighbors.
- ▶ When a **noninitiator** receives a message for the first time, it makes the sender its *parent*.

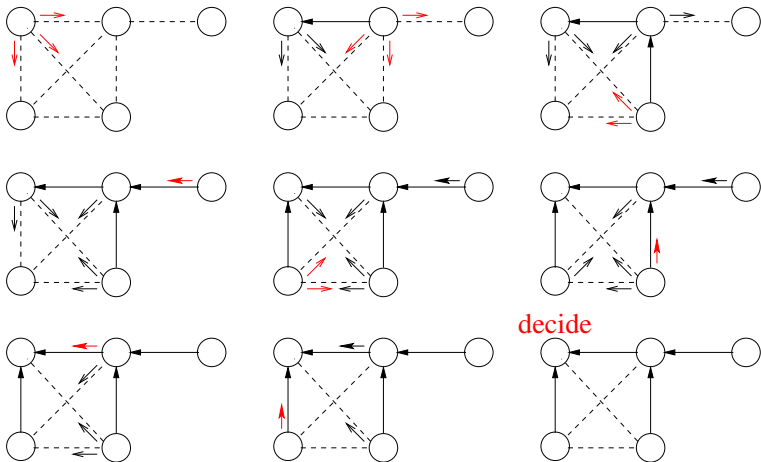
Then it sends a message to all neighbors except its parent.

- ▶ When a **noninitiator** has received a message from all neighbors, it sends a message to its parent.
- ▶ When the **initiator** has received a message from all neighbors, it *decides*.

Message complexity: $2E$ messages



Echo algorithm - Example



Echo algorithm - Correctness

Claim: The echo algorithm is a wave algorithm.

Proof: A spanning tree is constructed.

Clearly, through each frond edge, one message travels either way.

A leaf p in the tree eventually receives a message from all neighbors, and sends a message to its parent.

When this message arrives, consider the network without p :
By induction, noninitiators eventually send a message to their parent.

Concluding, through each channel one message travels either way.

So eventually the initiator decides.

All messages are causally before this decision.

Echo algorithm - Correctness

Claim: The echo algorithm is a wave algorithm.

Proof: A spanning tree is constructed.

Clearly, through each frond edge, one message travels either way.

A leaf p in the tree eventually receives a message from all neighbors, and sends a message to its parent.

When this message arrives, consider the network without p :
By induction, noninitiators eventually send a message to their parent.

Concluding, through each channel one message travels either way.

So eventually the initiator decides.

All messages are causally before this decision.

Echo algorithm - Correctness

Claim: The echo algorithm is a wave algorithm.

Proof: A spanning tree is constructed.

Clearly, through each frond edge, one message travels either way.

A leaf p in the tree eventually receives a message from all neighbors, and sends a message to its parent.

When this message arrives, consider the network without p :
By induction, noninitiators eventually send a message to their parent.

Concluding, through each channel one message travels either way.

So eventually the initiator decides.

All messages are causally before this decision.

Echo algorithm - Correctness

Claim: The echo algorithm is a wave algorithm.

Proof: A spanning tree is constructed.

Clearly, through each frond edge, one message travels either way.

A leaf p in the tree eventually receives a message from all neighbors, and sends a message to its parent.

When this message arrives, consider the network without p :
By induction, noninitiators eventually send a message to their parent.

Concluding, through each channel one message travels either way.

So eventually the initiator decides.

All messages are causally before this decision.

Use the echo algorithm to determine the largest process id.

Let each process initiate a run of the echo algorithm, tagged by its id.

Processes only participate in the “largest” wave they have seen so far.

Which of these concurrent waves complete?

Use the echo algorithm to determine the largest process id.

Let each process initiate a run of the echo algorithm, tagged by its id.

Processes only participate in the “largest” wave they have seen so far.

Which of these concurrent waves complete?