# PRACTICAL PROVABLY SECURE FLOODING FOR BLOCKCHAINS

Chen-Da Liu-Zhang, *NTT Research*

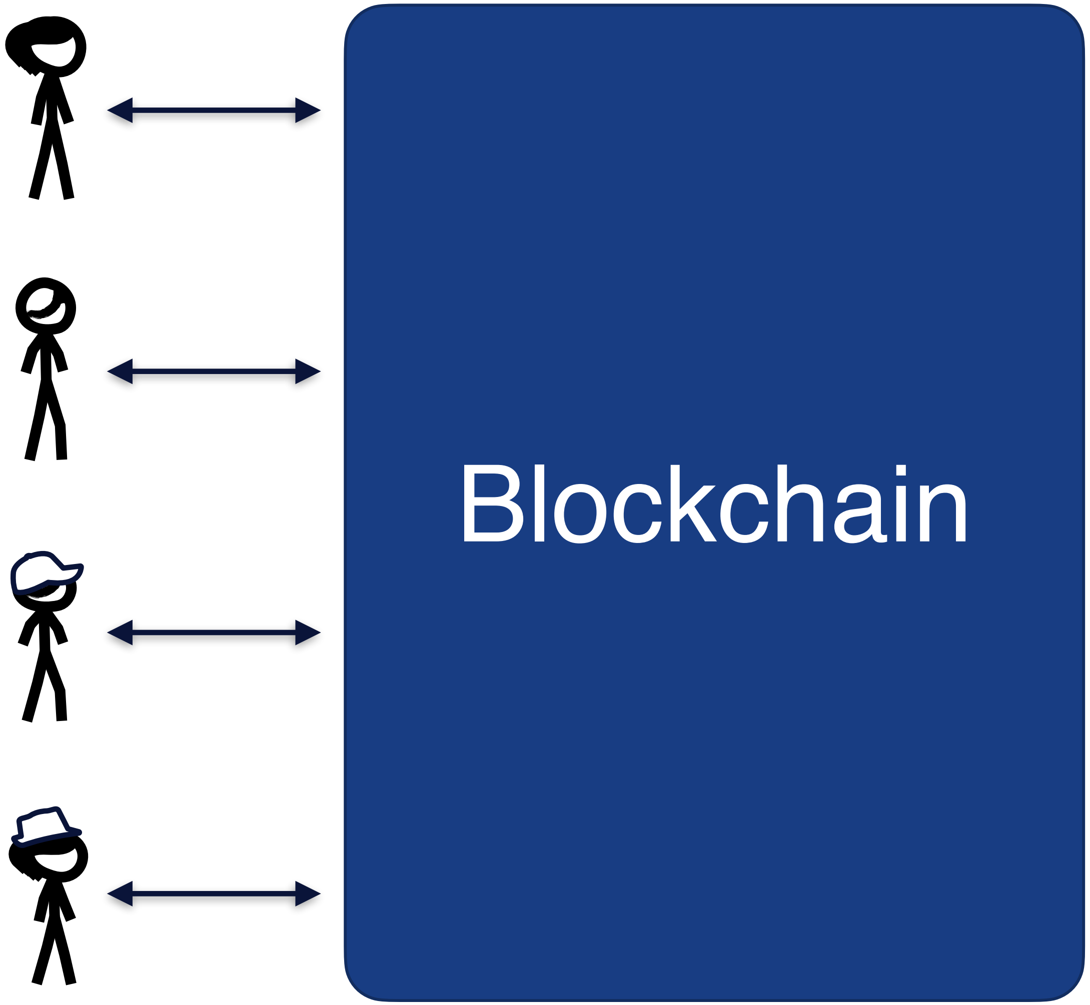
Christian Matt, *Concordium*

Ueli Maurer, *ETH Zurich*

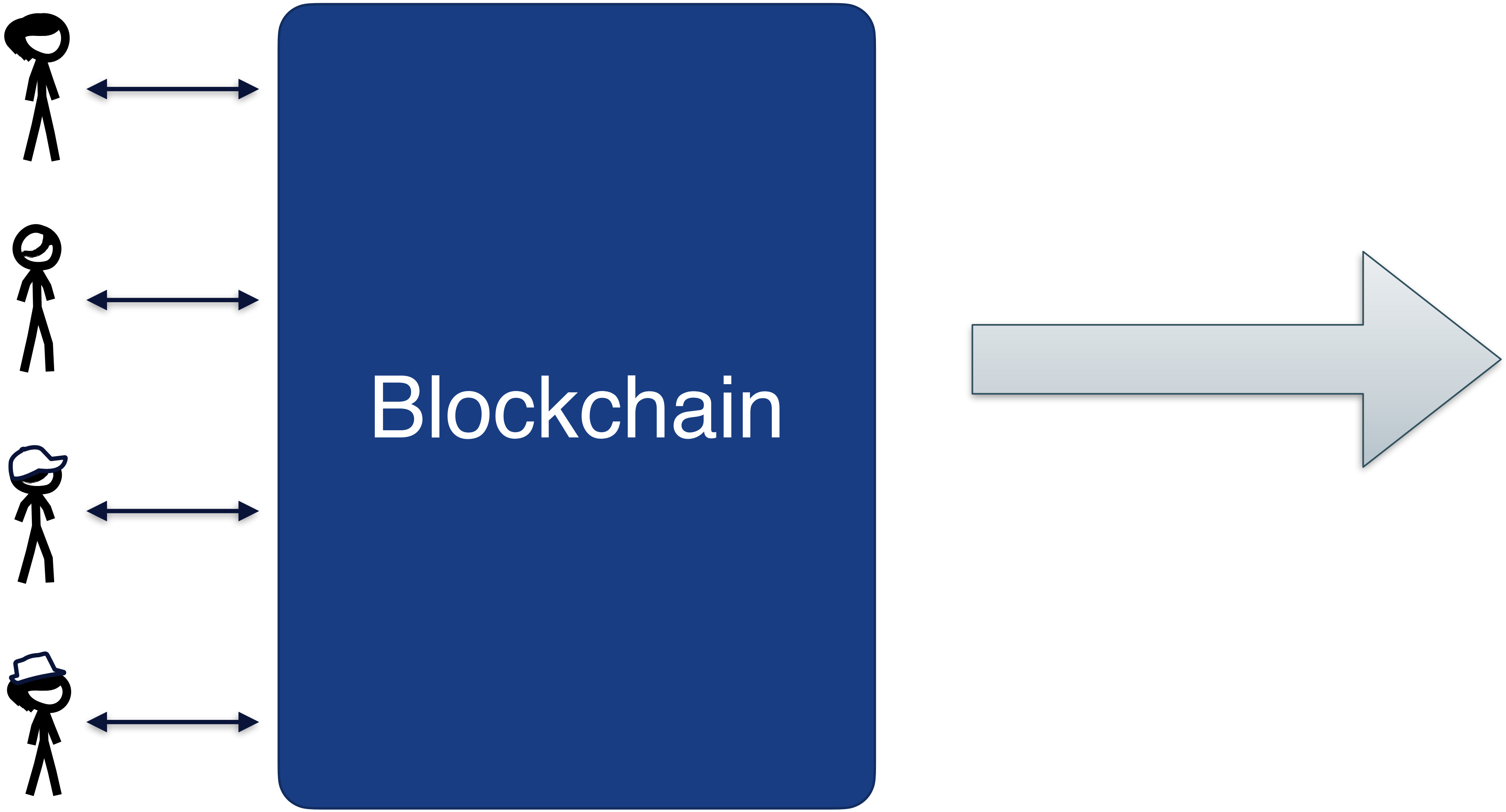
Guilherme Rito, *ETH Zurich*

**Søren Eller Thomsen,** *Aarhus University*

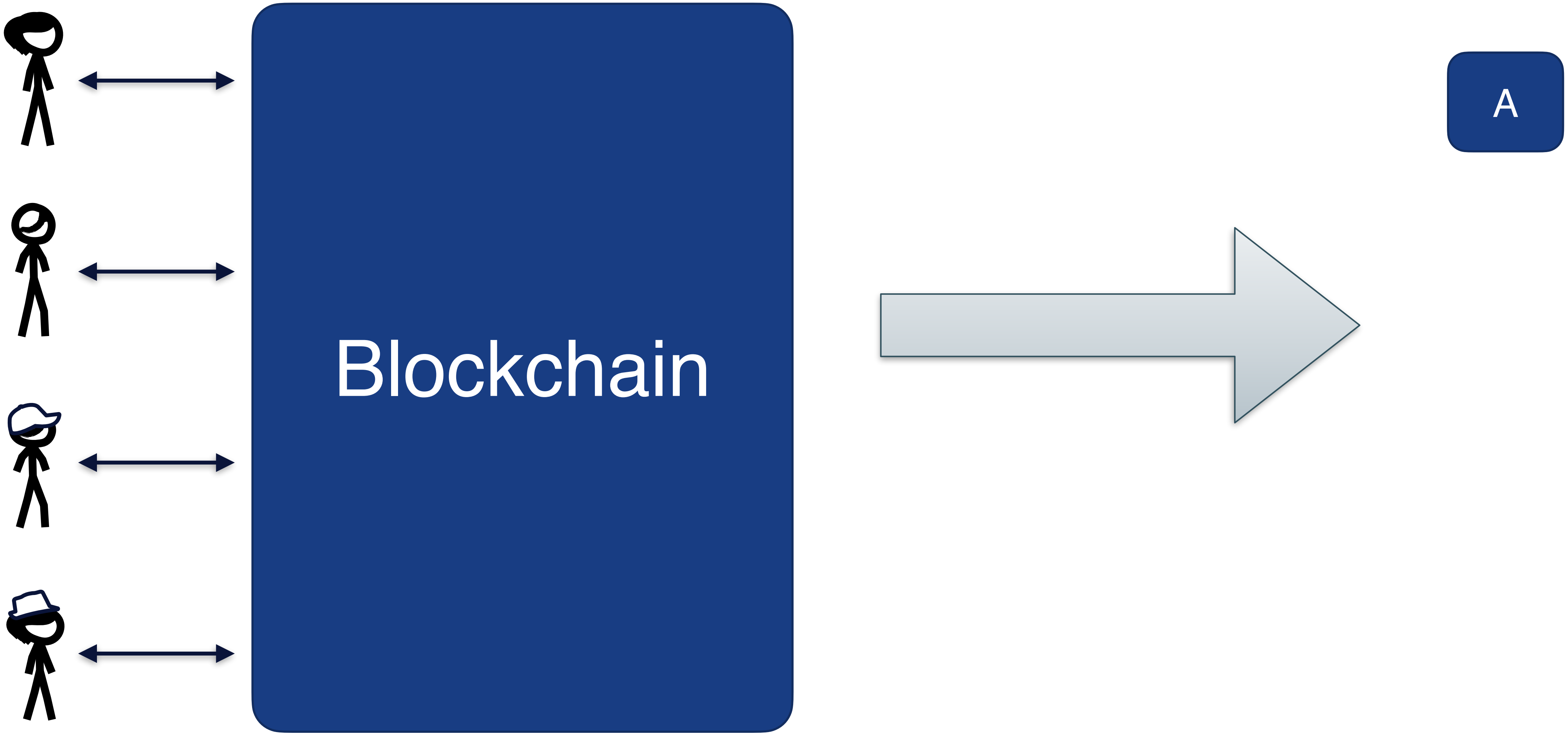
# BLOCKCHAINS



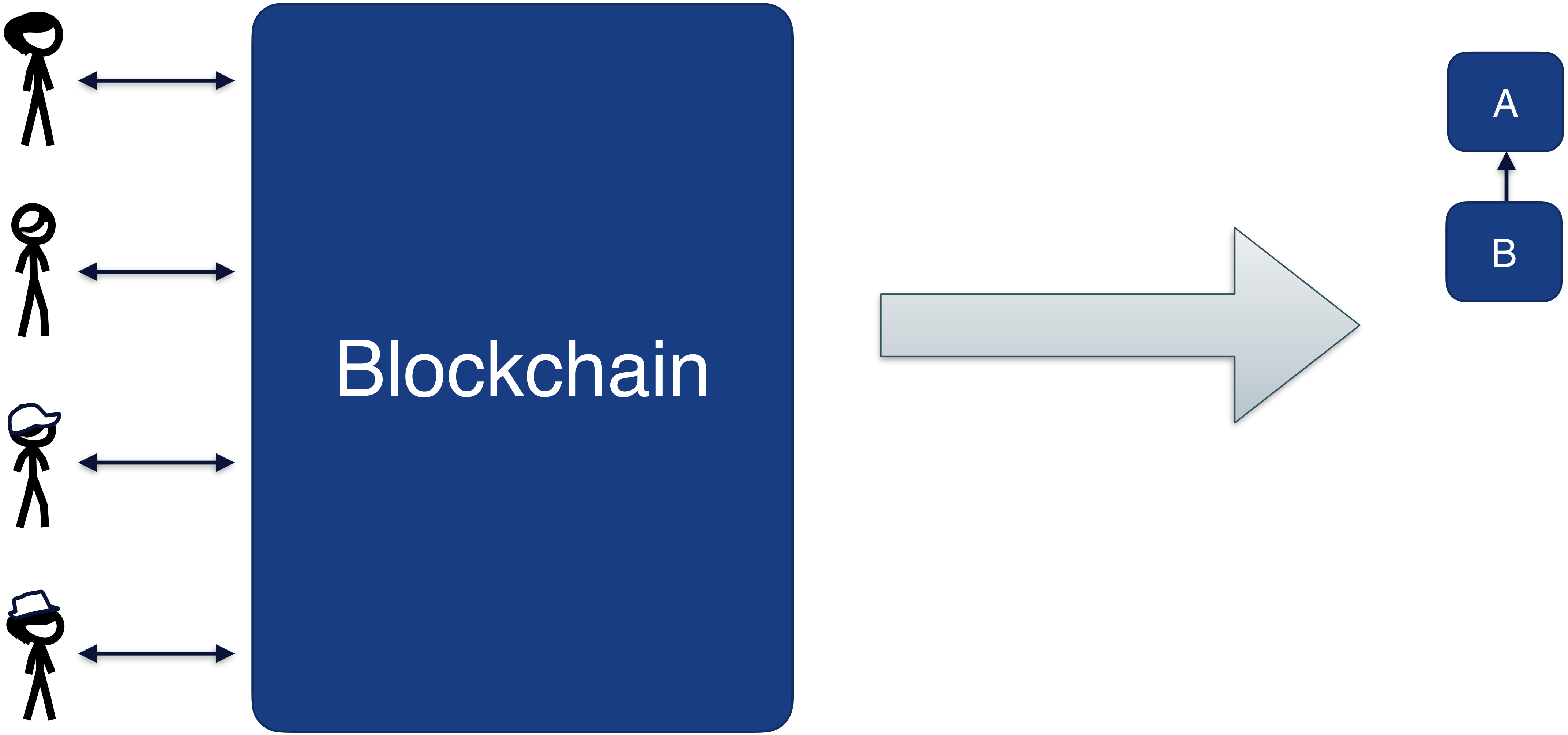
# BLOCKCHAINS



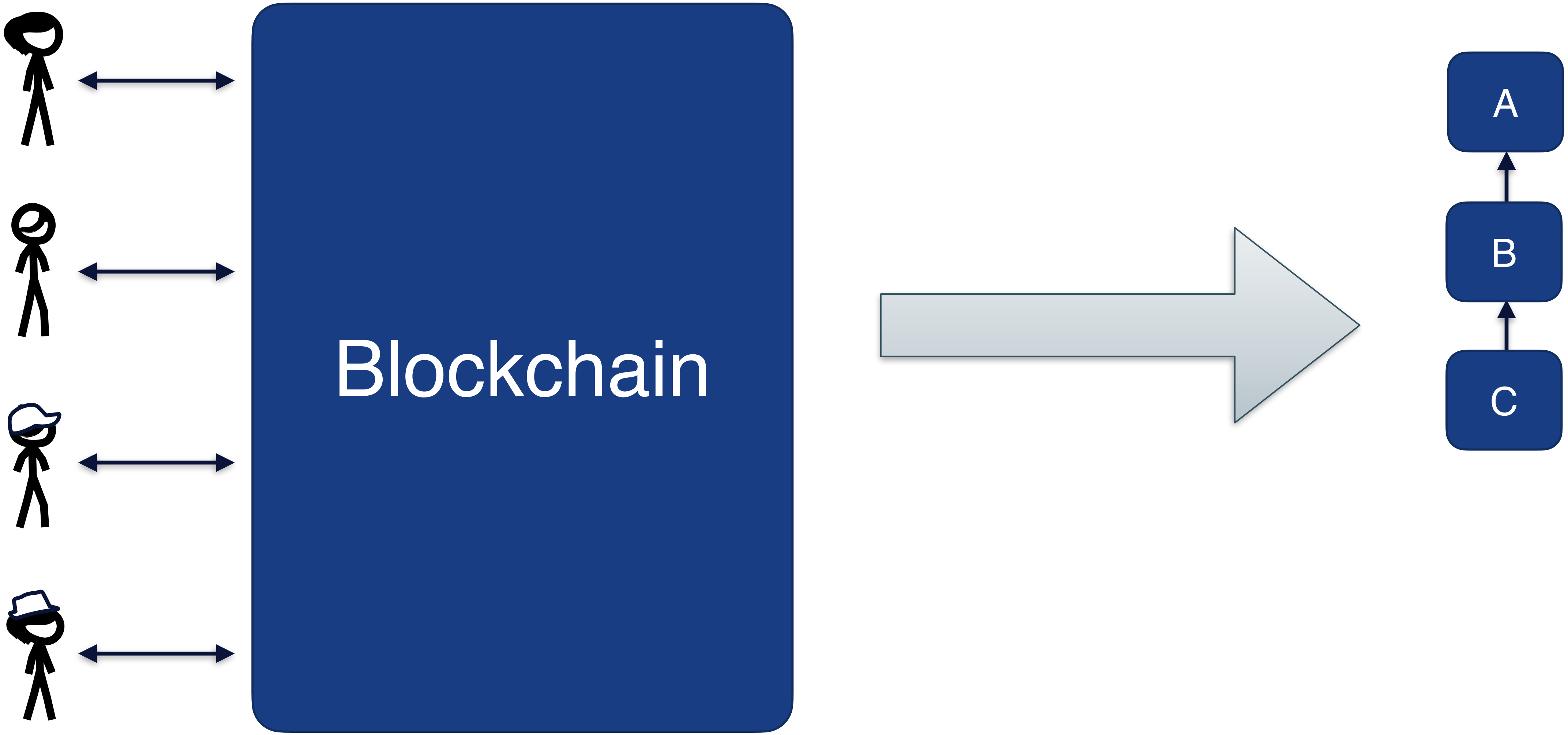
# BLOCKCHAINS



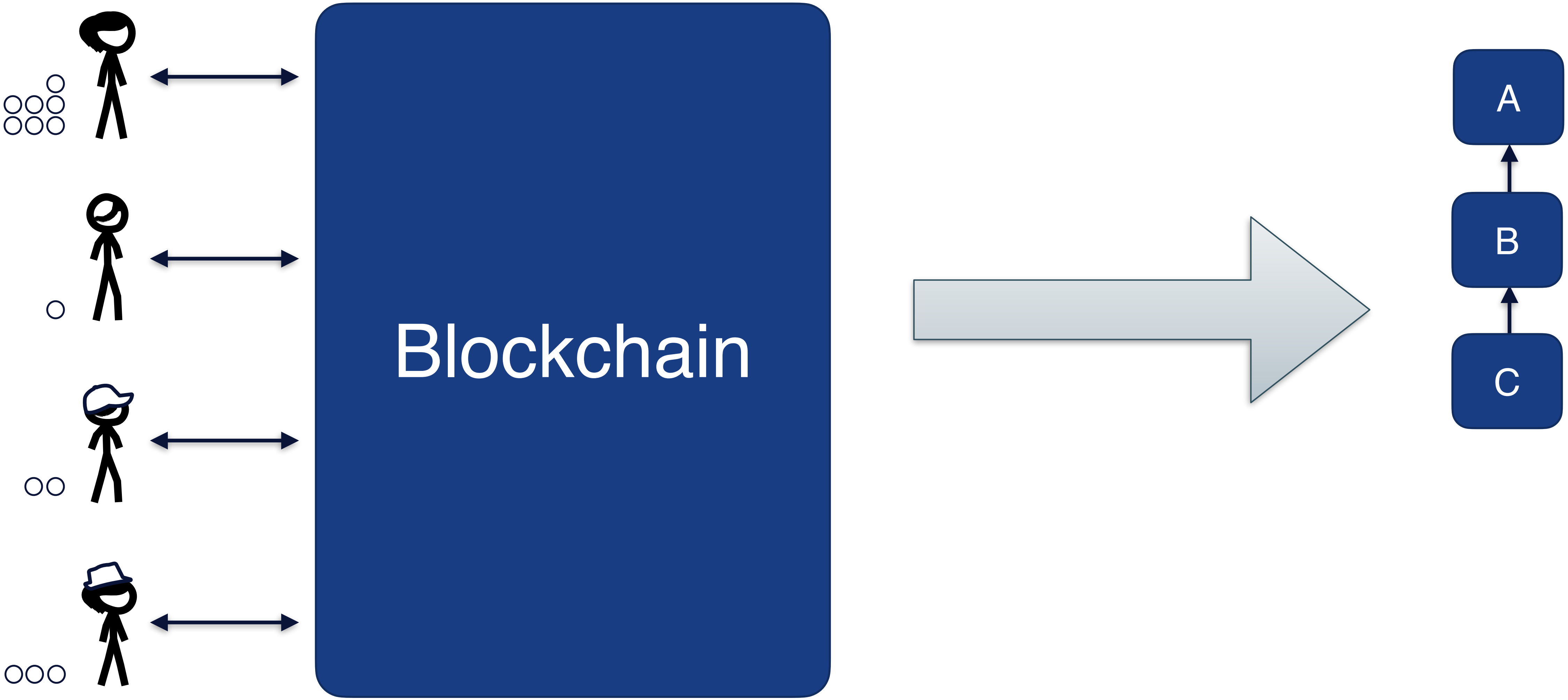
# BLOCKCHAINS



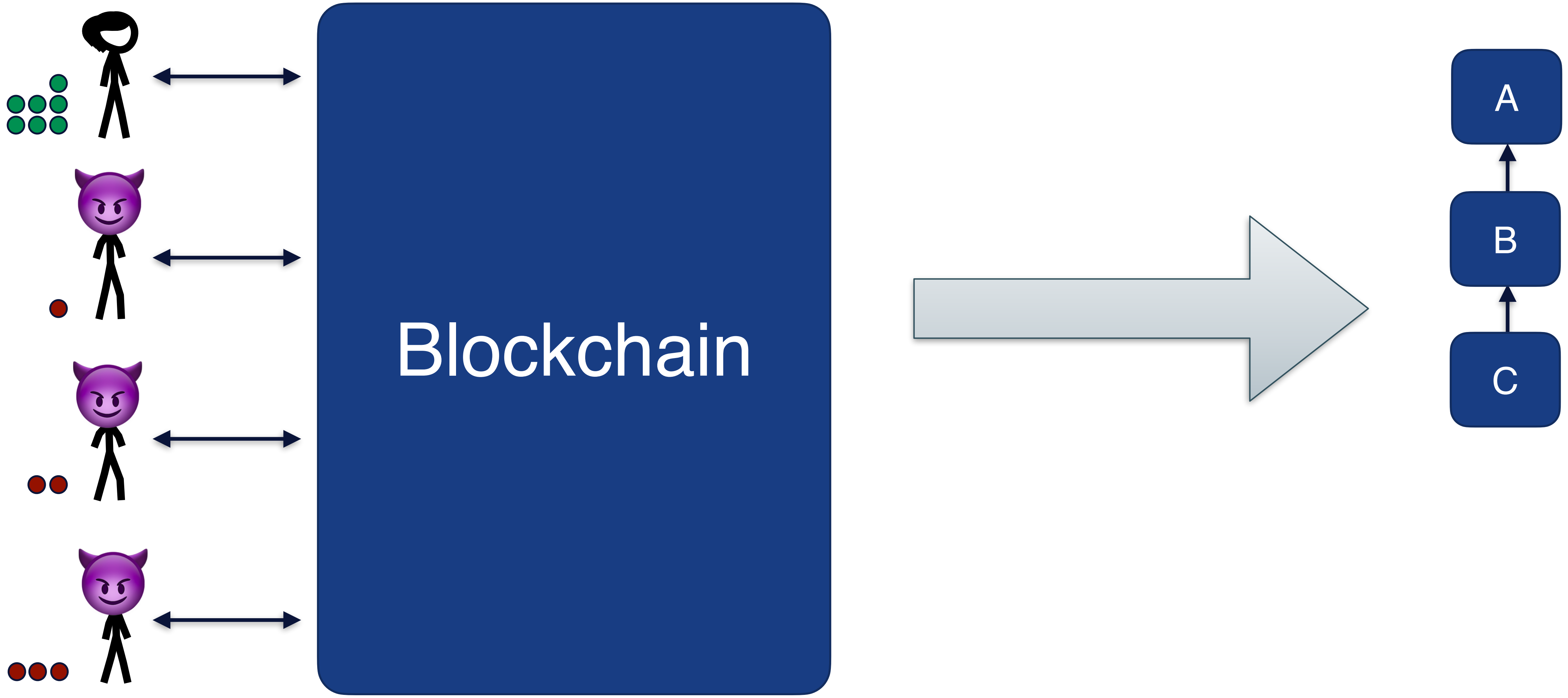
# BLOCKCHAINS



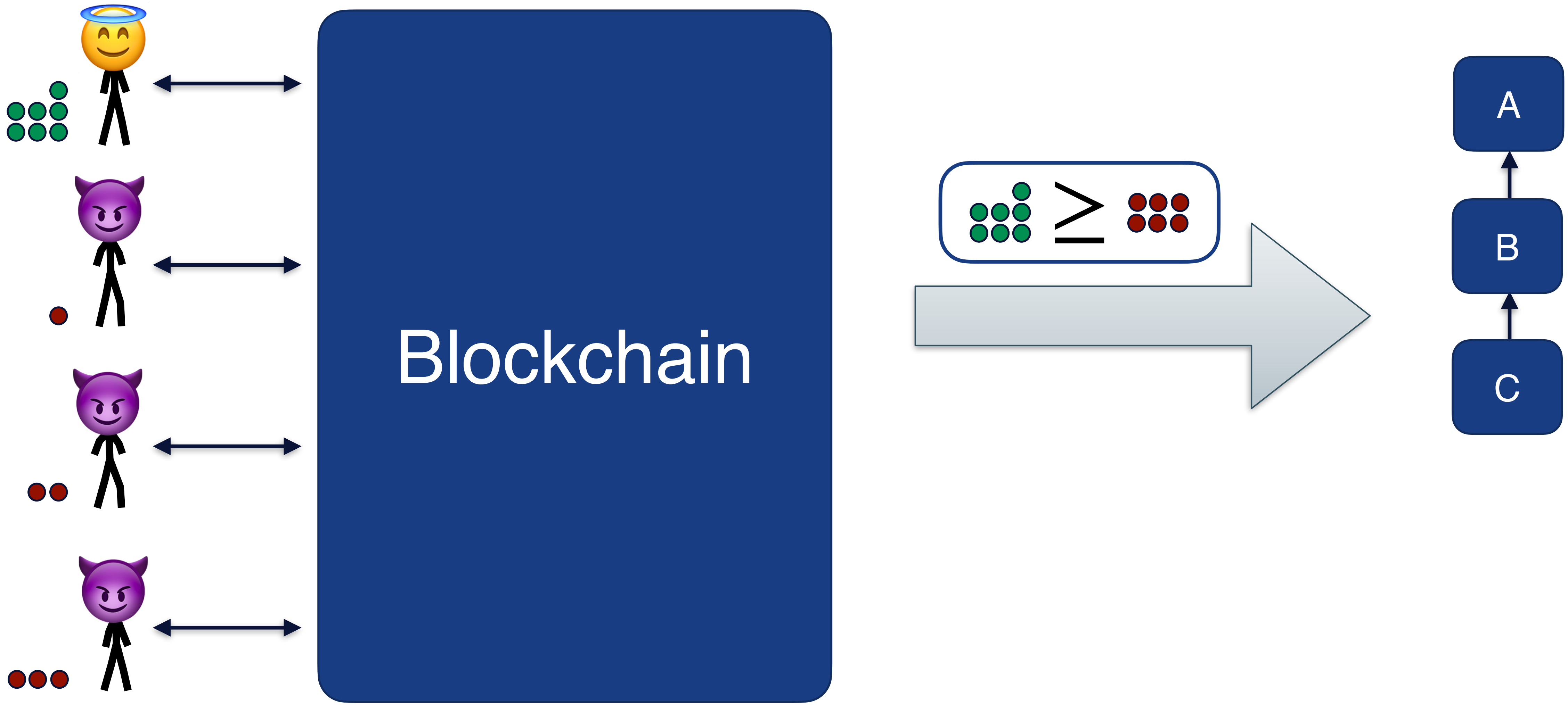
# BLOCKCHAINS



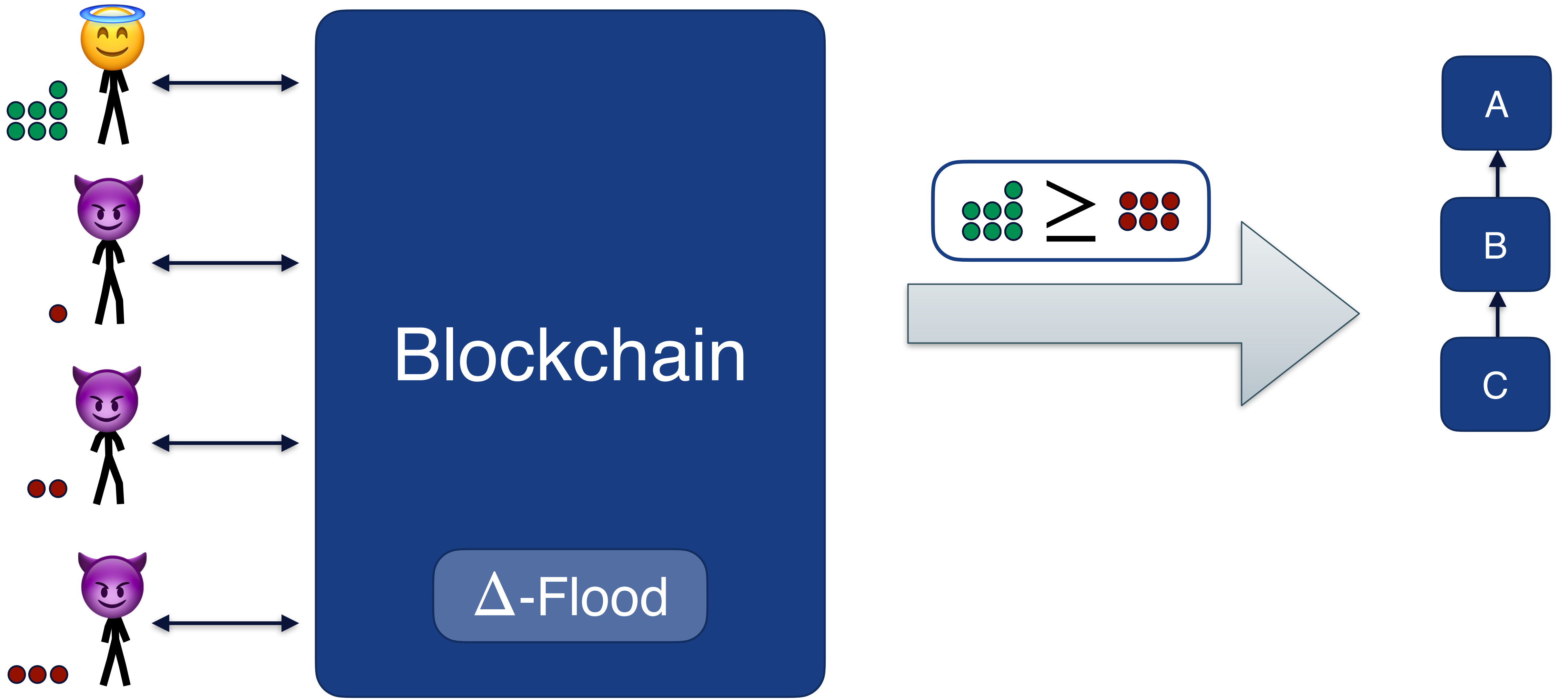
# BLOCKCHAINS



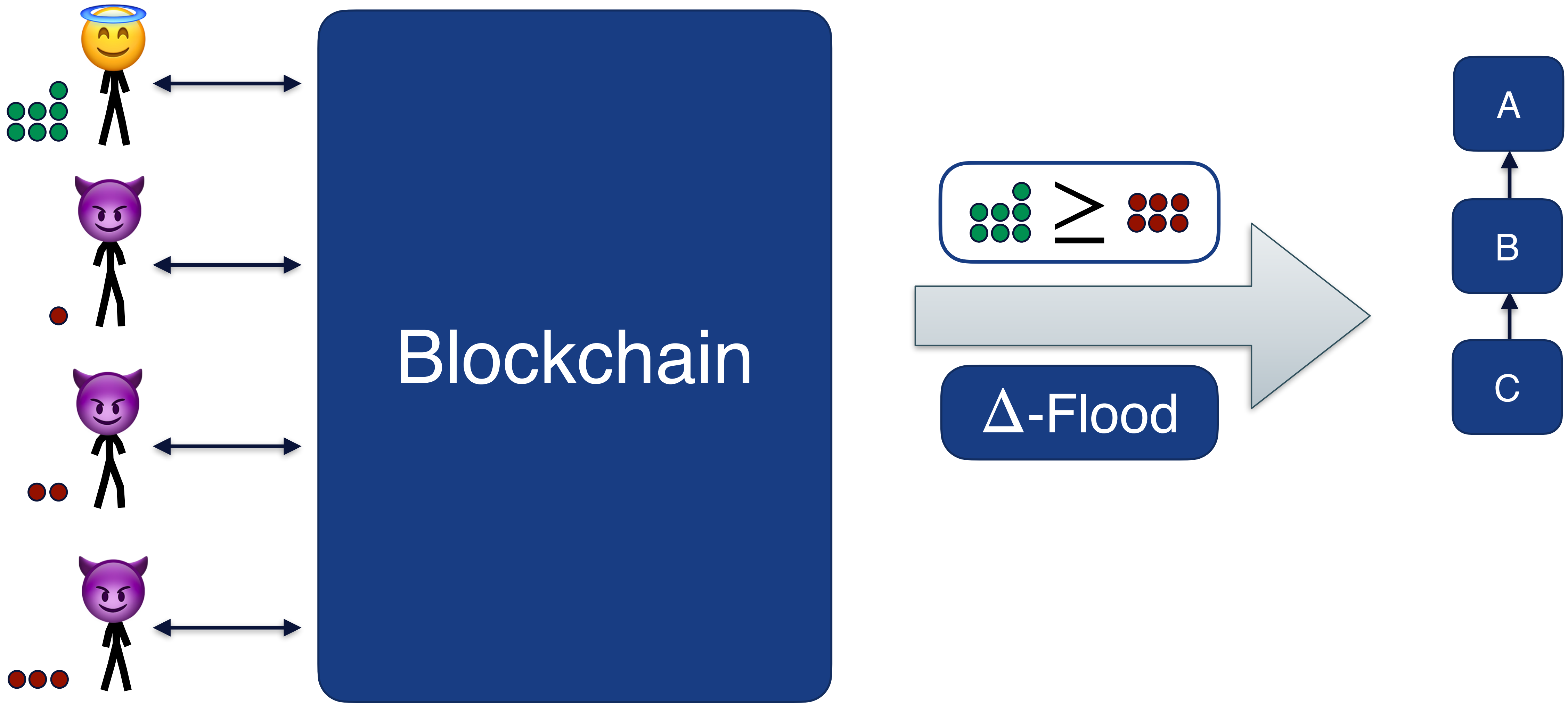
# BLOCKCHAINS



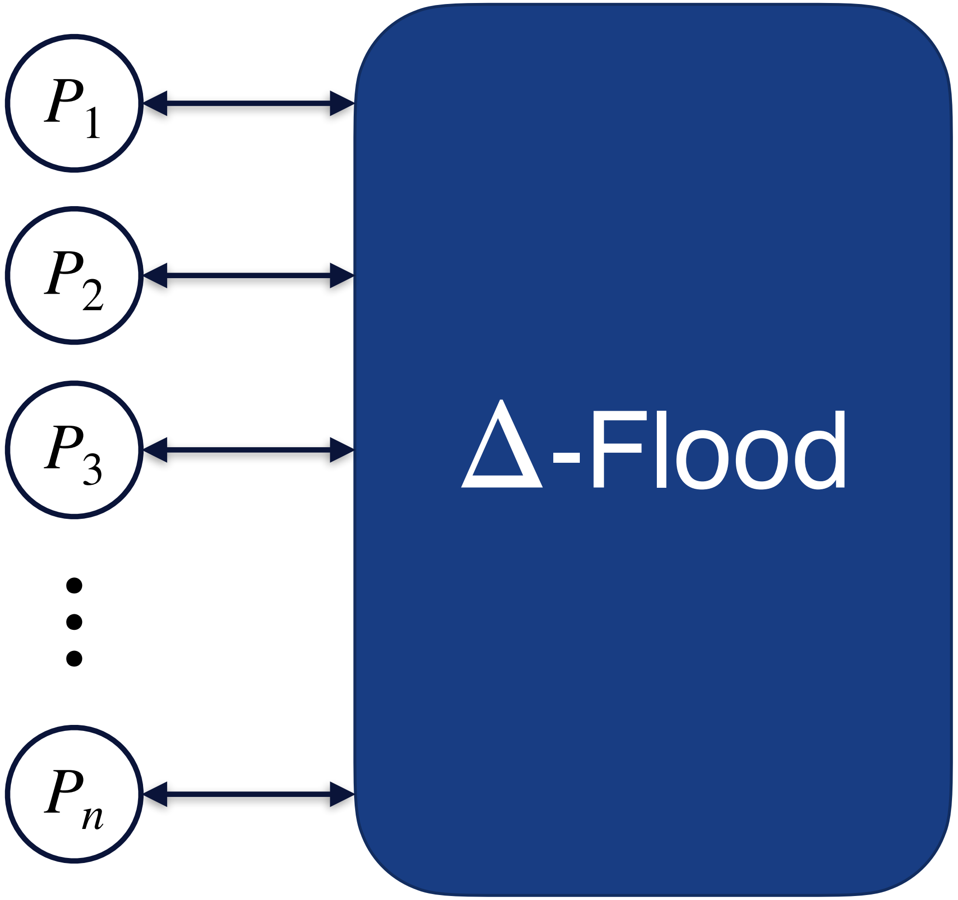
# BLOCKCHAINS



# BLOCKCHAINS

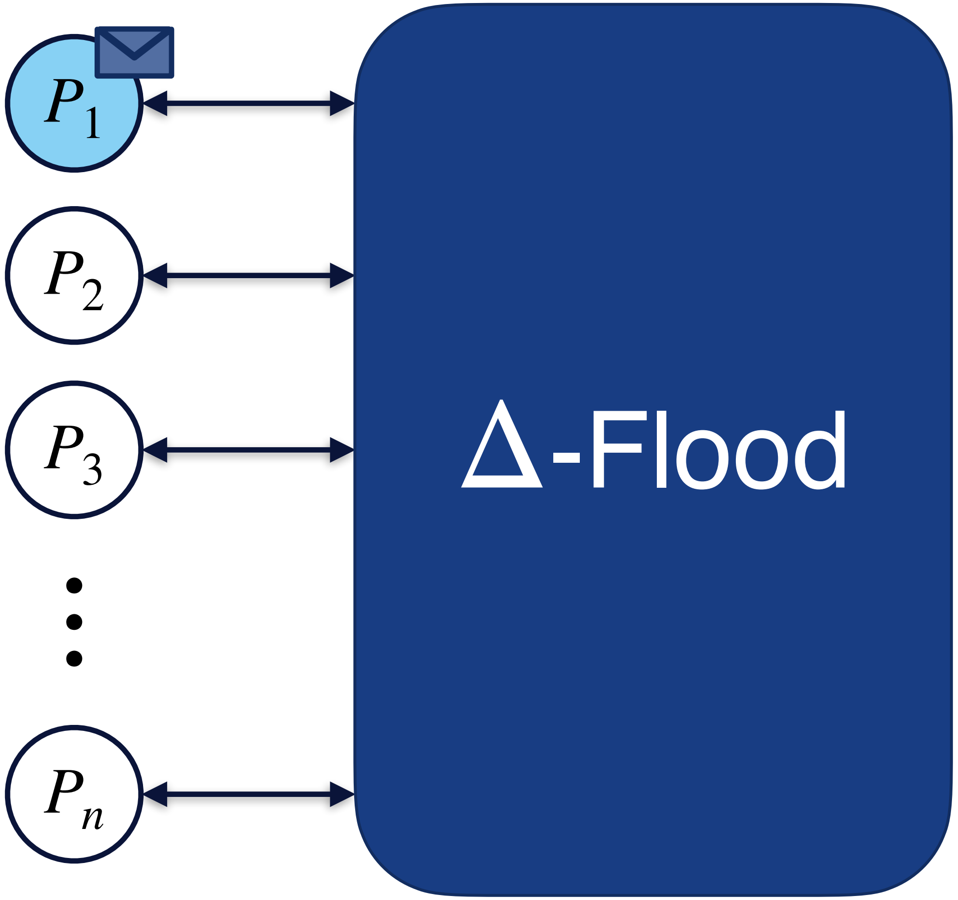


# FLOODING FOR BLOCKCHAINS



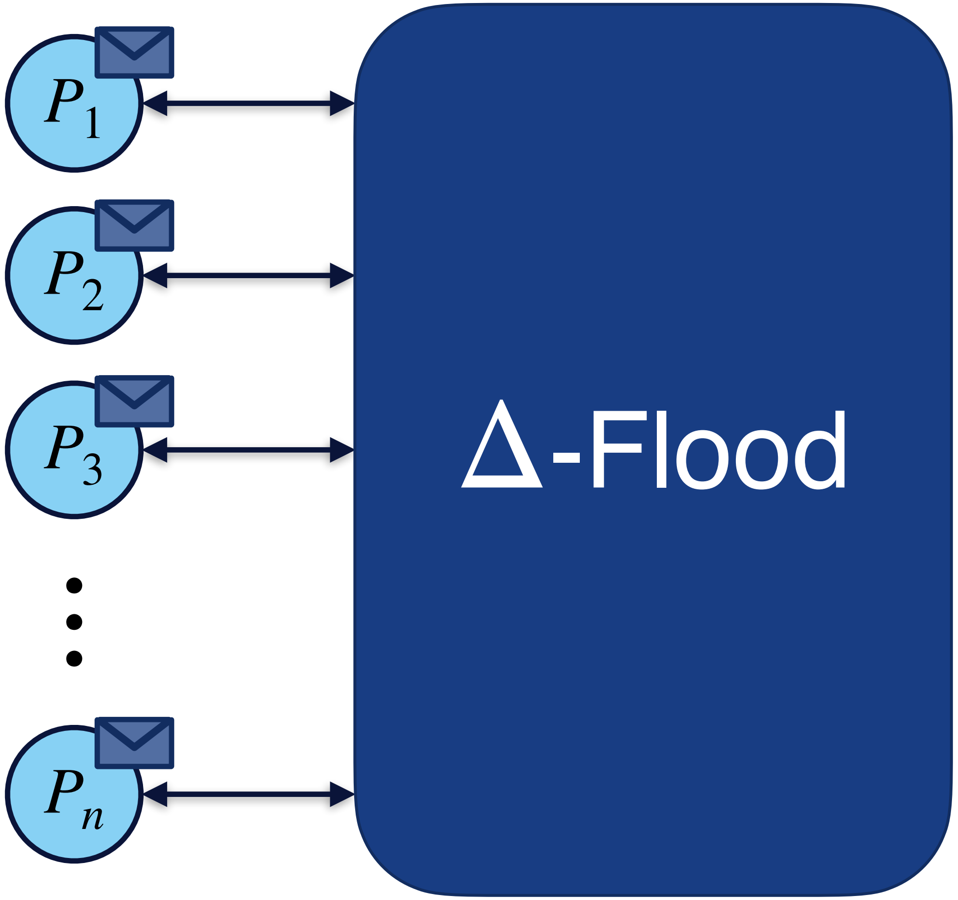
# FLOODING FOR BLOCKCHAINS

- Input messages must be delivered within Δ time.



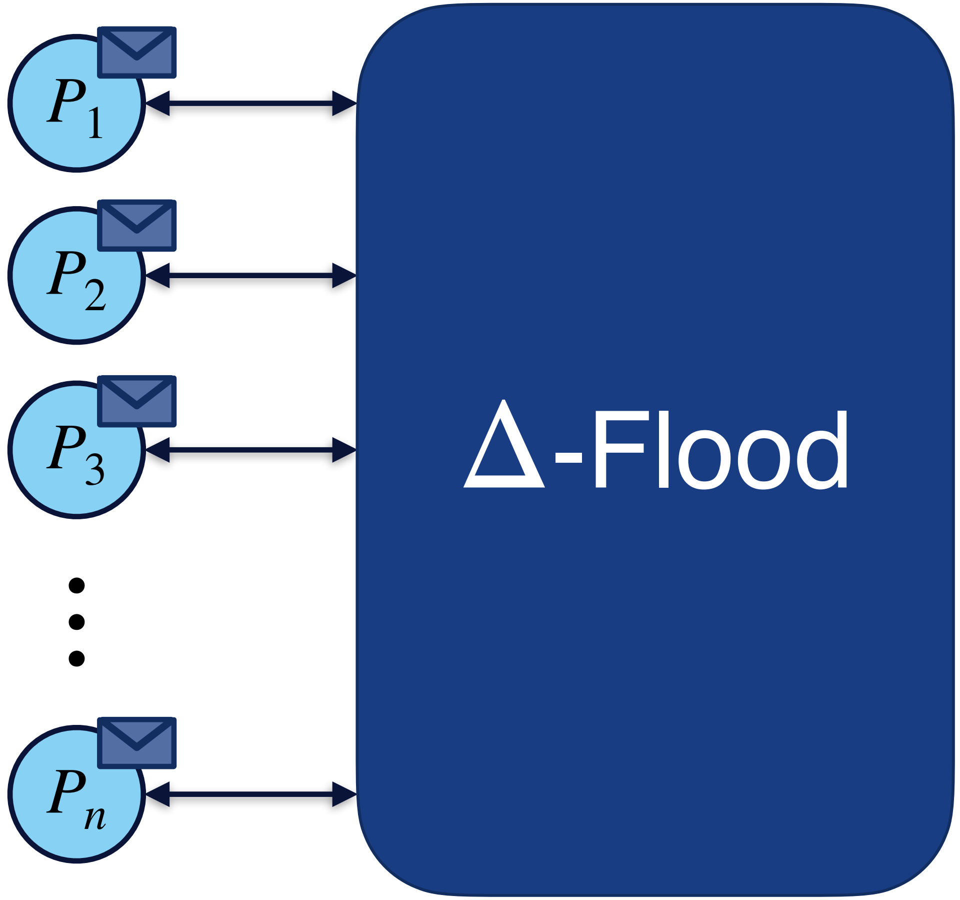
# FLOODING FOR BLOCKCHAINS

- Input messages must be delivered within $\Delta$ time.

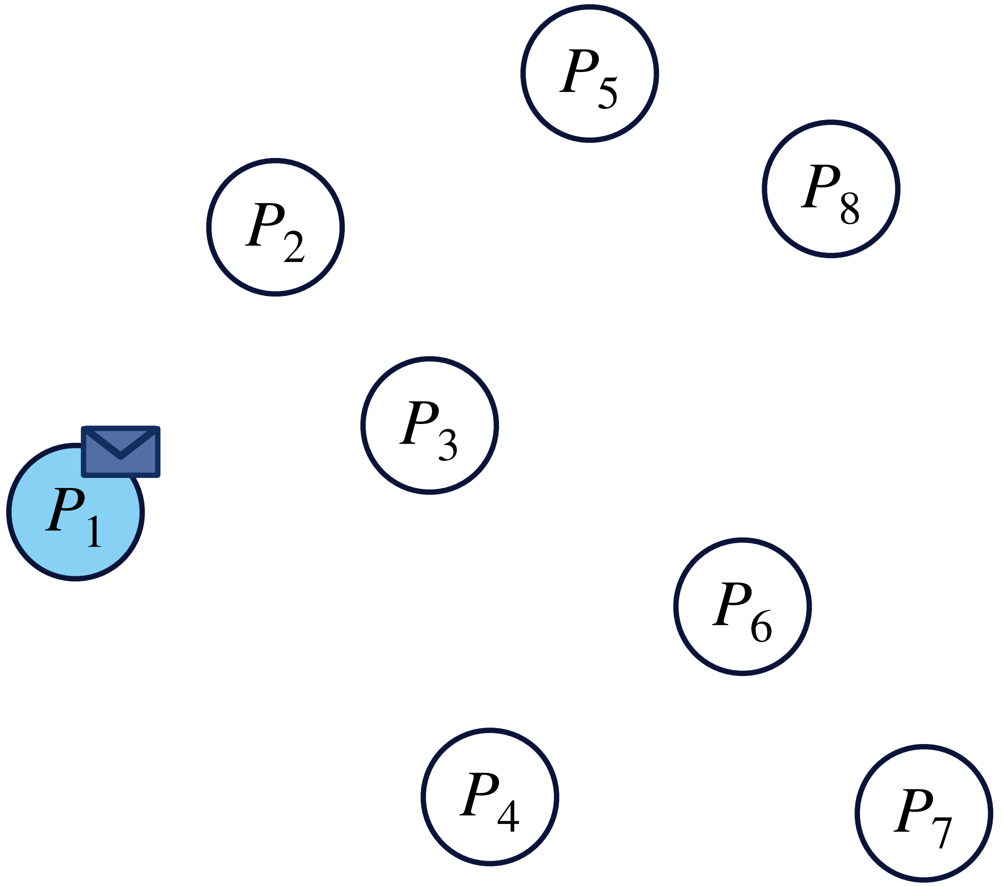


# FLOODING FOR BLOCKCHAINS

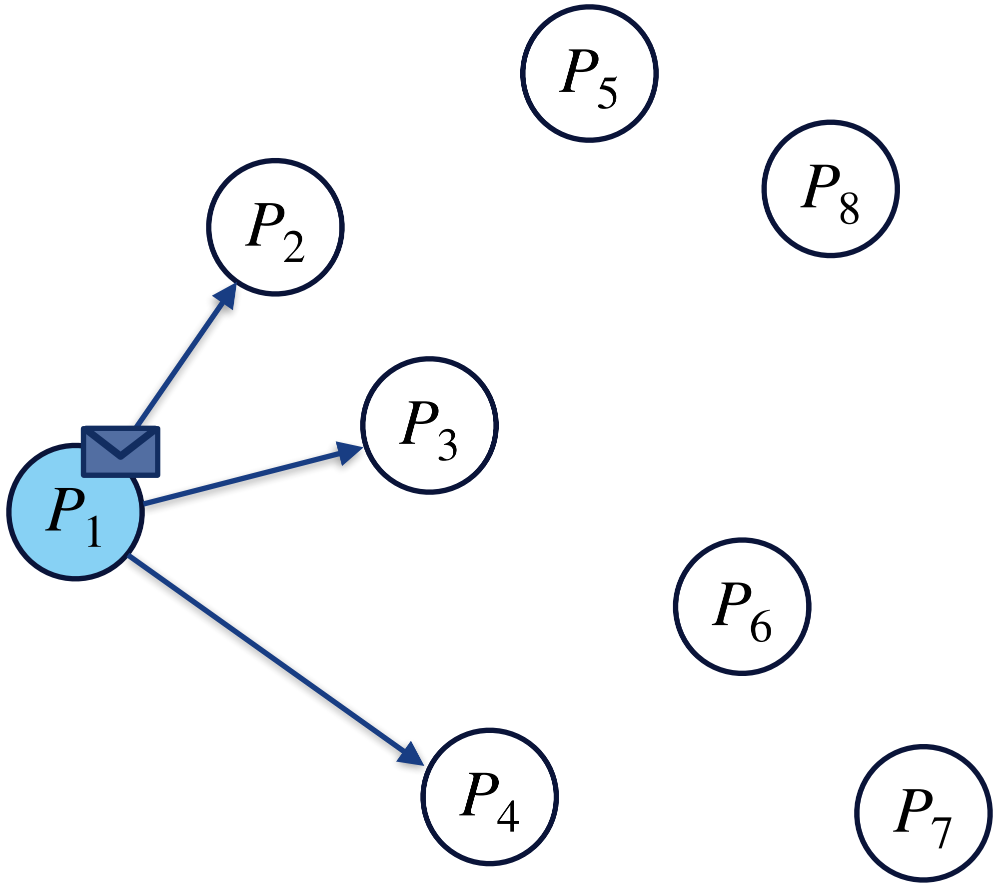
- Input messages must be delivered within $\Delta$ time.
  - Assumed to prove security of blockchains [GKL15,PS17,DGKR18,PS18,CM19,DMM+20].



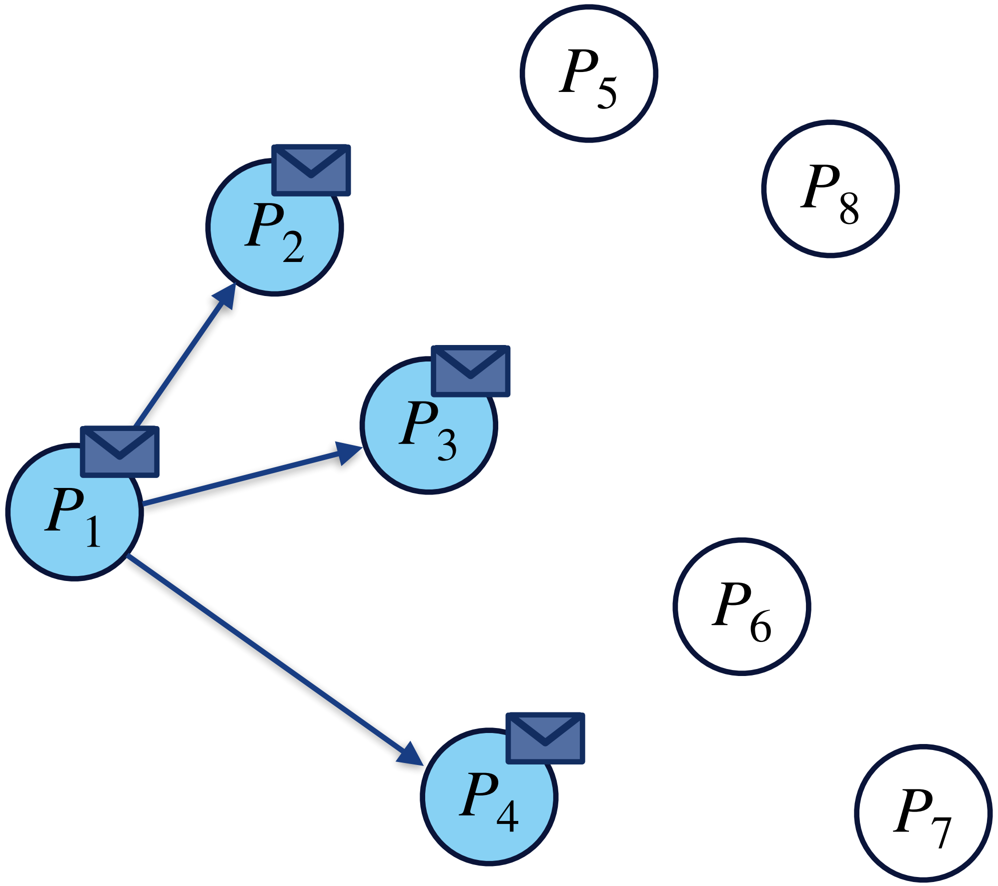
# FLOODING IN PRACTICE



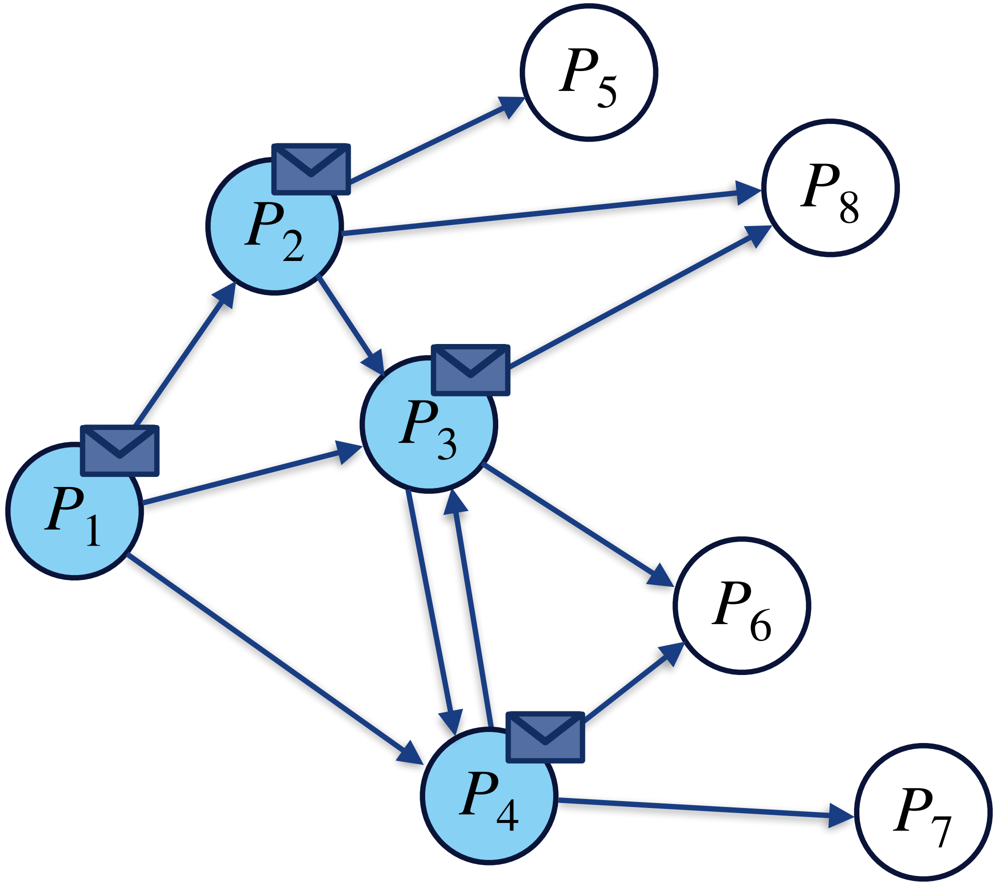
# FLOODING IN PRACTICE



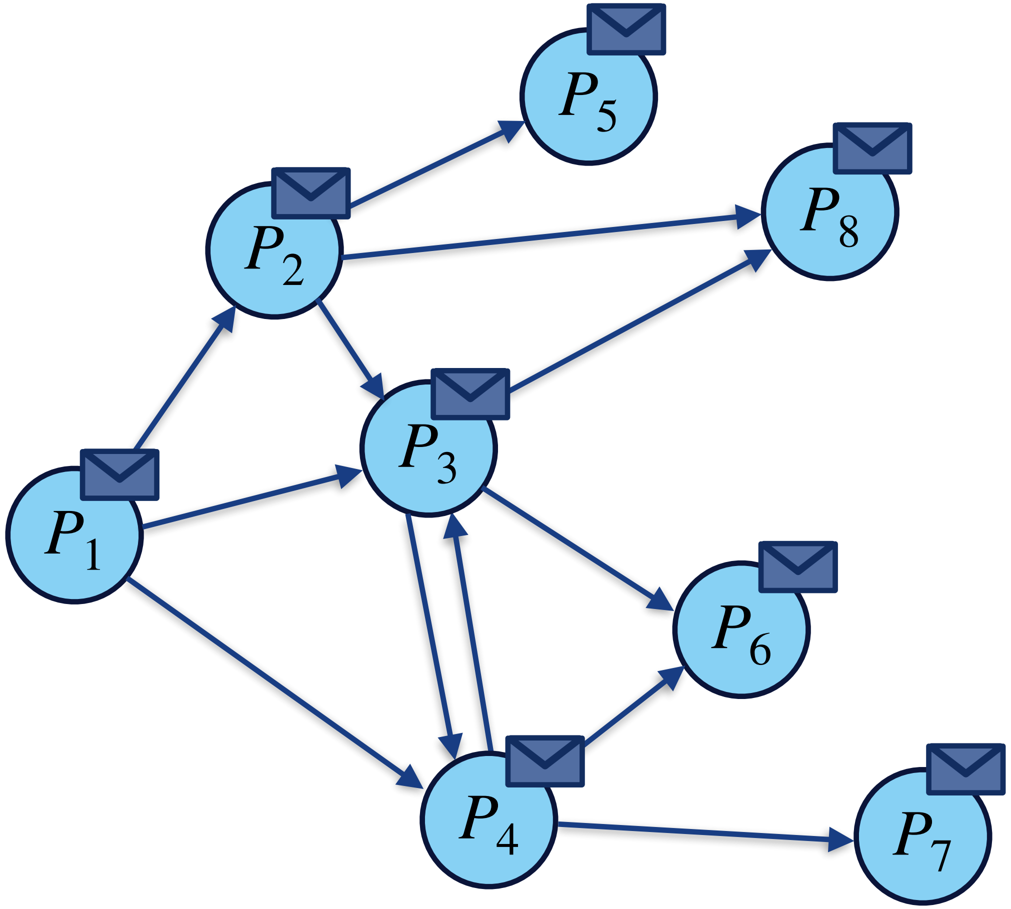
# FLOODING IN PRACTICE



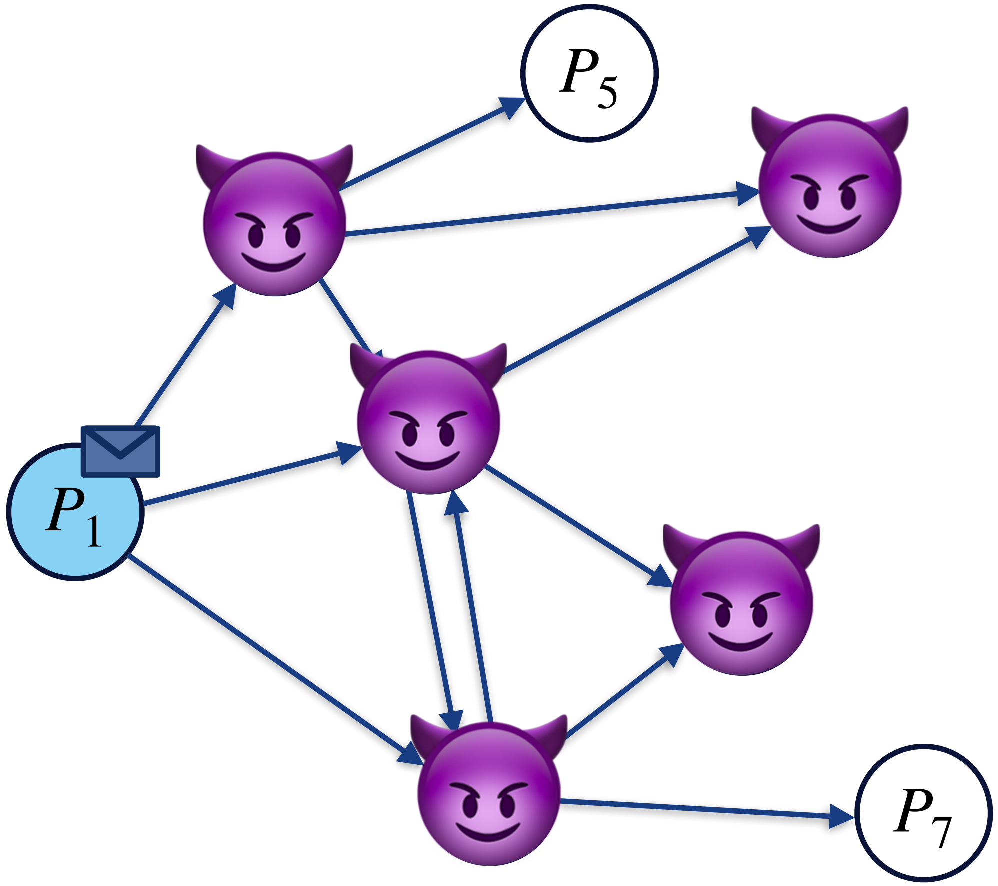
# FLOODING IN PRACTICE



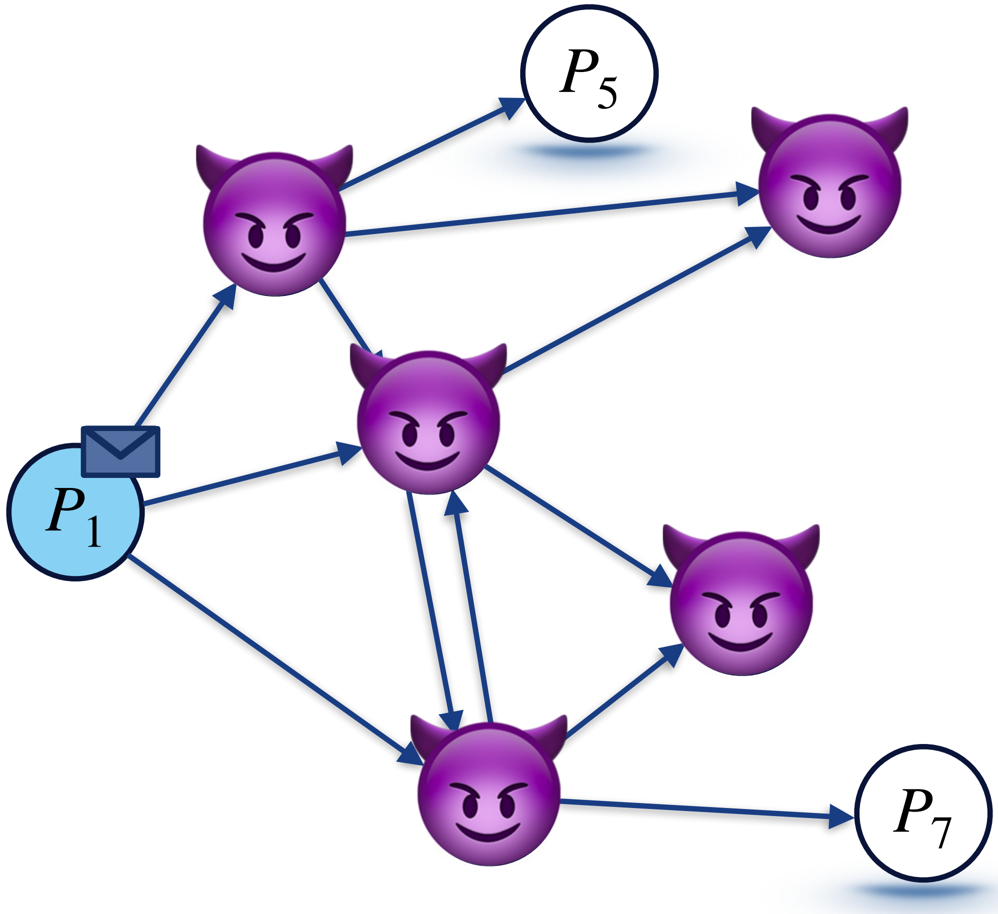
# FLOODING IN PRACTICE



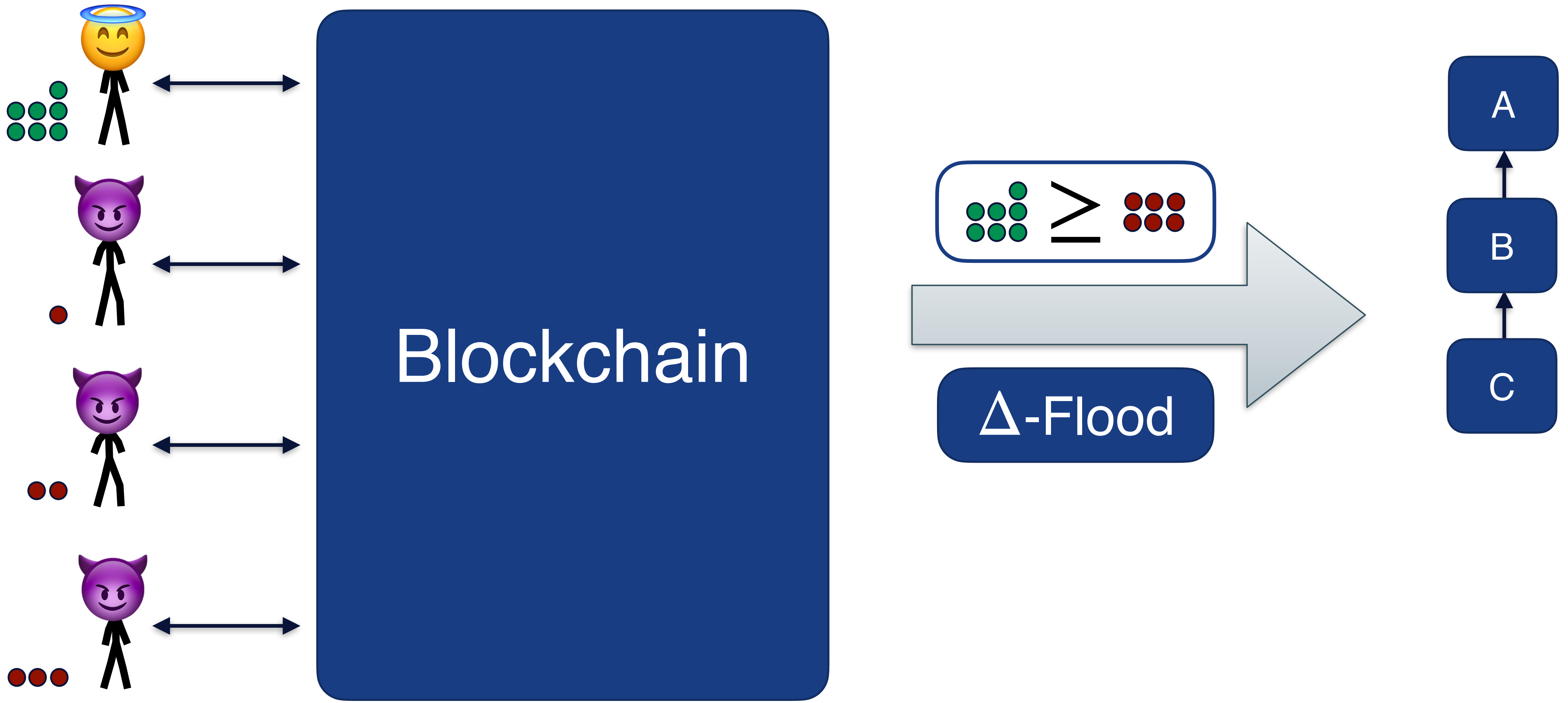
# FLOODING IN PRACTICE



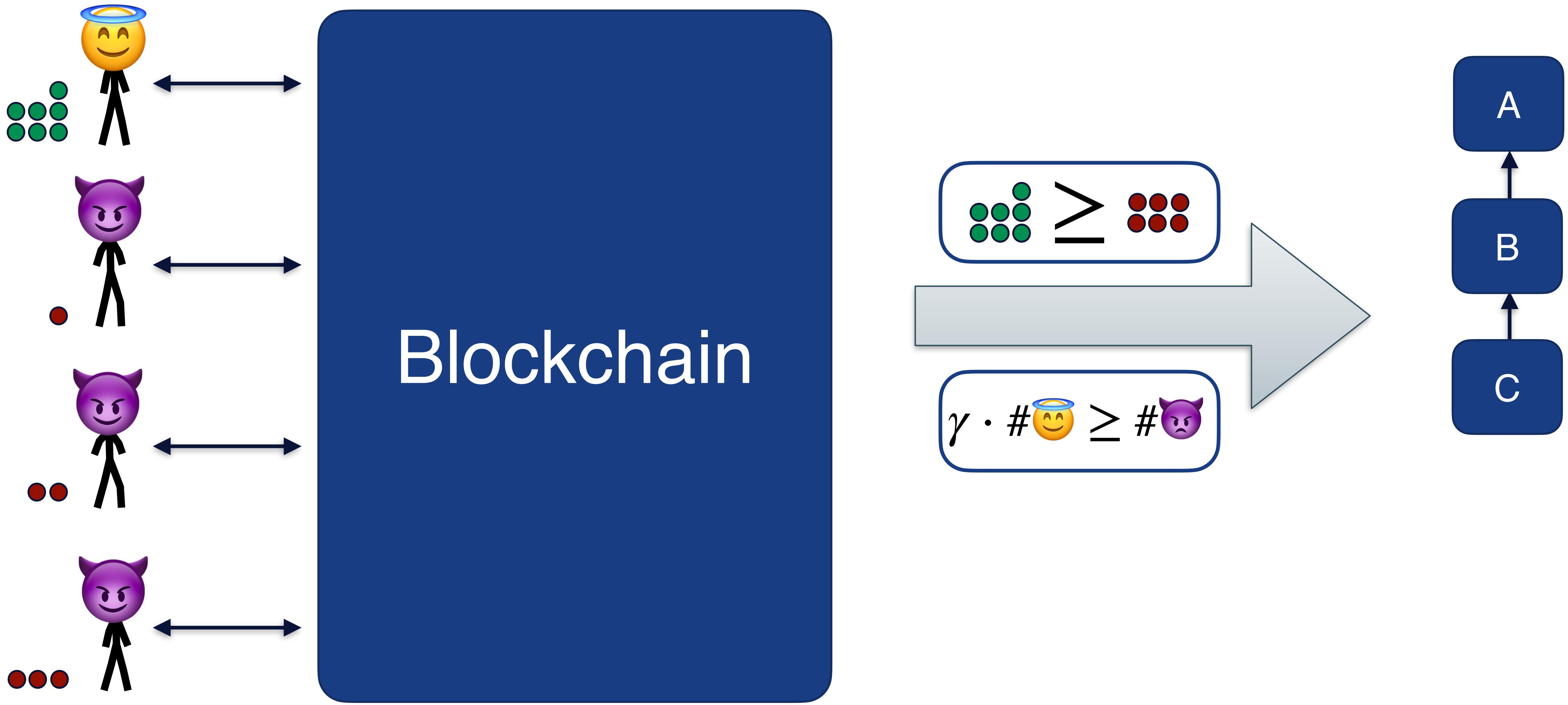
# FLOODING IN PRACTICE



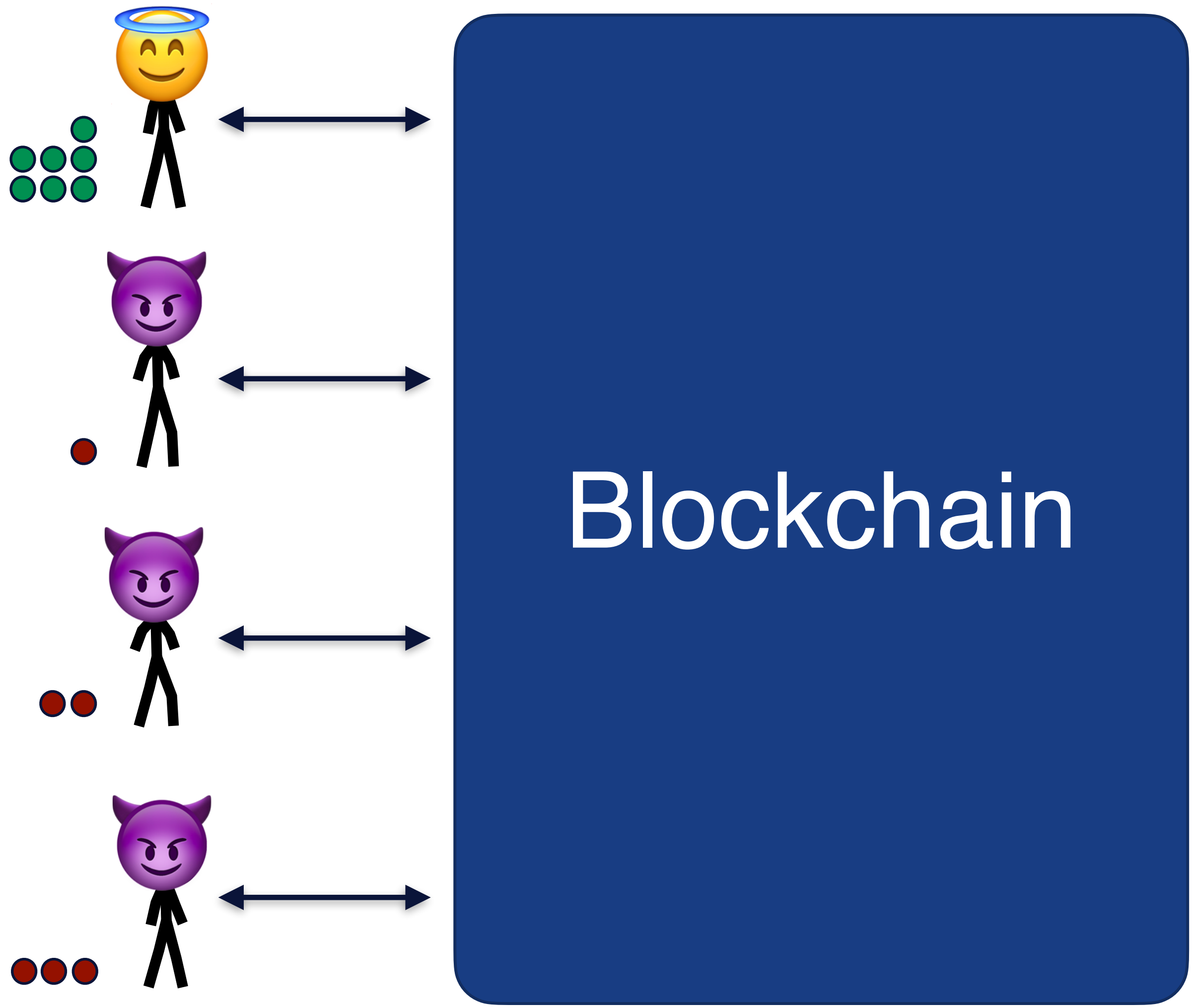
# BLOCKCHAINS



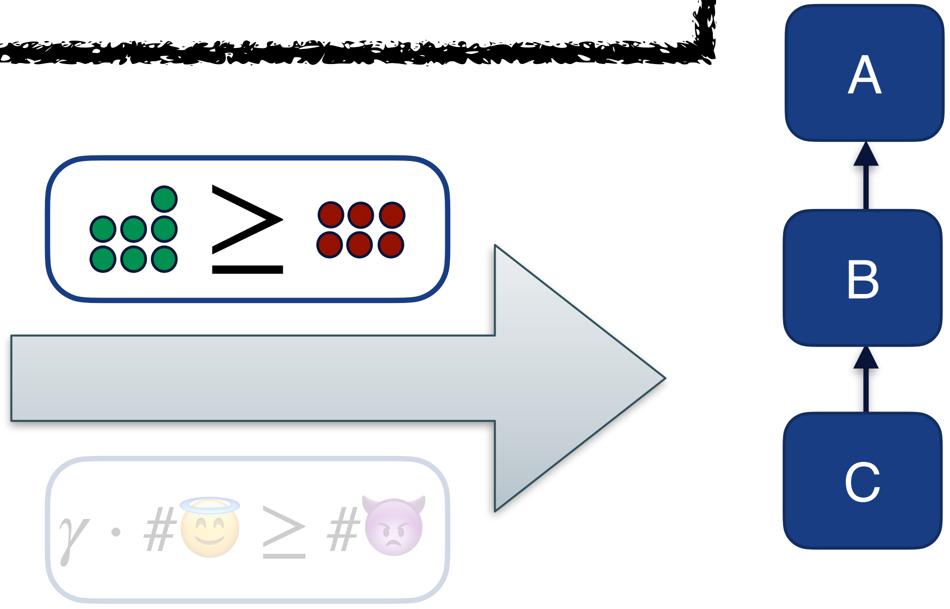
# BLOCKCHAINS



# BLOCKCHAINS




**Wanted!**



# OUR WORK

Q: Can efficient flooding be realized assuming a constant fraction of honest weight?

# OUR WORK

Q: Can efficient flooding be realized assuming a constant fraction of honest weight?

A: YES!

# CONTRIBUTIONS

## Practical Provably Secure Flooding for Blockchains

Chen-Da Liu-Zhang*[1], Christian Matt[2], Ueli Maurer[3], Guilherme Rito[3], and Søren Eller Thomsen†[4]

[1]NTT Research, USA
chen-da.liuzhang@ntt-research.com
[2]Concordium, Zurich, Switzerland
cm@concordium.com
[3]Department of Computer Science, ETH Zurich, Switzerland
{maurer, gteixeir}@inf.ethz.ch
[4]Concordium Blockchain Research Center, Aarhus University, Denmark
sethomsen@cs.au.dk

September 28, 2022

**Abstract**

In recent years, permisionless blockchains have received a lot of attention both from industry and academia, where substantial effort has been spent to develop consensus protocols that are secure under the assumption that less than half (or a third) of a given resource (e.g., stake or computing power) is controlled by corrupted parties. The security proofs of these consensus protocols usually assume the availability of a network functionality guaranteeing that a block sent by an honest party is received by all honest parties within some bounded time. To obtain an overall protocol that is secure under the same corruption assumption, it is therefore necessary to combine the consensus protocol with a network protocol that achieves this property under that assumption. In practice, however, the underlying network is typically implemented by flooding protocols that are not proven to be secure in the setting where a fraction of the considered total weight can be corrupted. This has led to many so-called eclipse attacks on existing protocols and tailor-made fixes against specific attacks.

To close this apparent gap, we present the first practical flooding protocol that provably delivers sent messages to all honest parties after a logarithmic number of steps. We prove security in the setting where all parties are publicly assigned a positive weight and the adversary can corrupt parties accumulating up to a constant fraction of the total weight. This can directly be used in the proof-of-stake setting, but is not limited to it. To prove the security of our protocol, we combine known results about the diameter of Erdős–Rényi graphs with reductions between different types of random graphs. We further show that the efficiency of our protocol is asymptotically optimal.

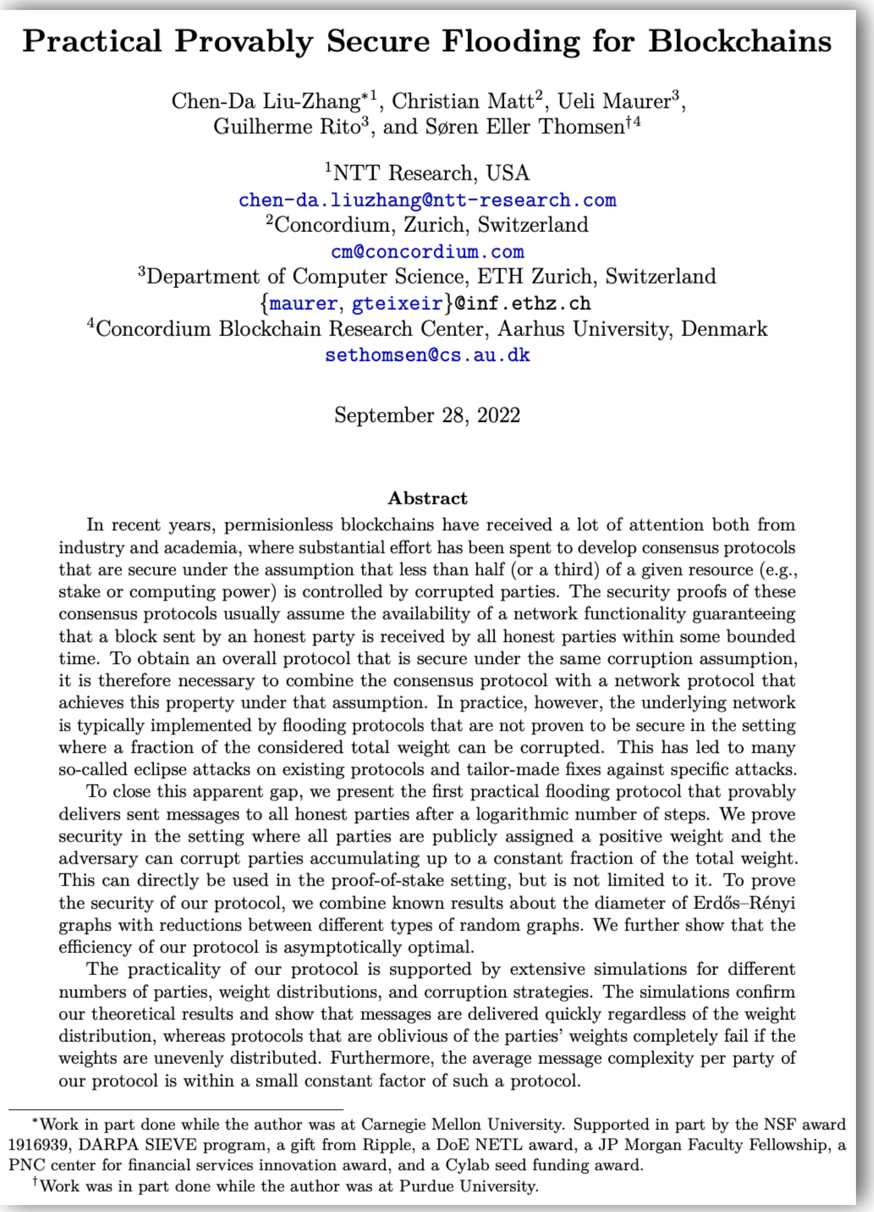
The practicality of our protocol is supported by extensive simulations for different numbers of parties, weight distributions, and corruption strategies. The simulations confirm our theoretical results and show that messages are delivered quickly regardless of the weight distribution, whereas protocols that are oblivious of the parties' weights completely fail if the weights are unevenly distributed. Furthermore, the average message complexity per party of our protocol is within a small constant factor of such a protocol.

*Work in part done while the author was at Carnegie Mellon University. Supported in part by the NSF award 1916939, DARPA SIEVE program, a gift from Ripple, a DoE NETL award, a JP Morgan Faculty Fellowship, a PNC center for financial services innovation award, and a Cylab seed funding award.

†Work was in part done while the author was at Purdue University.

# CONTRIBUTIONS

1. *Weighted Fanout Flooding* (WFF):



**Practical Provably Secure Flooding for Blockchains**

Chen-Da Liu-Zhang*[1], Christian Matt[2], Ueli Maurer[3], Guilherme Rito[3], and Søren Eller Thomsen†[4]

[1]NTT Research, USA
chen-da.liuzhang@ntt-research.com
[2]Concordium, Zurich, Switzerland
cm@concordium.com
[3]Department of Computer Science, ETH Zurich, Switzerland
{maurer, gteixeir}@inf.ethz.ch
[4]Concordium Blockchain Research Center, Aarhus University, Denmark
sethomsen@cs.au.dk

September 28, 2022

**Abstract**

In recent years, permisionless blockchains have received a lot of attention both from industry and academia, where substantial effort has been spent to develop consensus protocols that are secure under the assumption that less than half (or a third) of a given resource (e.g., stake or computing power) is controlled by corrupted parties. The security proofs of these consensus protocols usually assume the availability of a network functionality guaranteeing that a block sent by an honest party is received by all honest parties within some bounded time. To obtain an overall protocol that is secure under the same corruption assumption, it is therefore necessary to combine the consensus protocol with a network protocol that achieves this property under that assumption. In practice, however, the underlying network is typically implemented by flooding protocols that are not proven to be secure in the setting where a fraction of the considered total weight can be corrupted. This has led to many so-called eclipse attacks on existing protocols and tailor-made fixes against specific attacks.

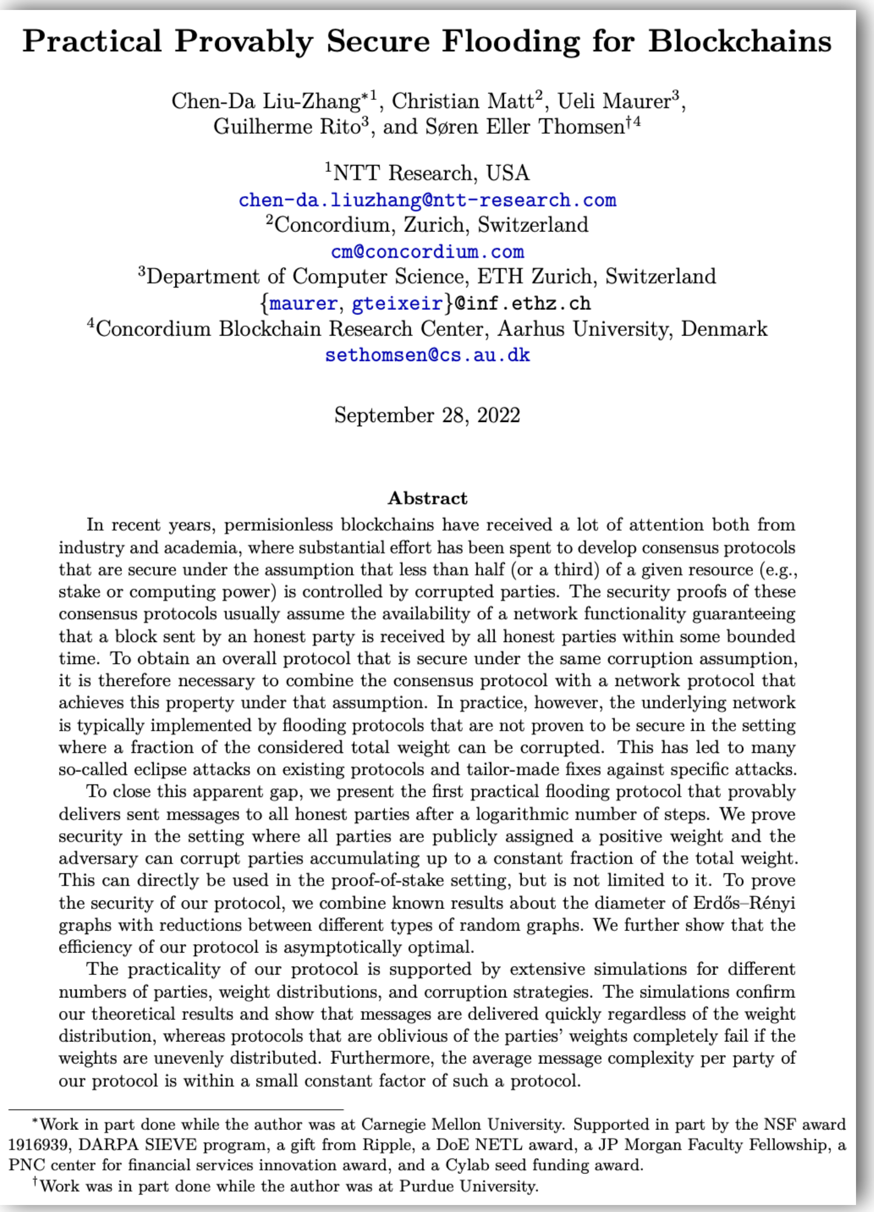
To close this apparent gap, we present the first practical flooding protocol that provably delivers sent messages to all honest parties after a logarithmic number of steps. We prove security in the setting where all parties are publicly assigned a positive weight and the adversary can corrupt parties accumulating up to a constant fraction of the total weight. This can directly be used in the proof-of-stake setting, but is not limited to it. To prove the security of our protocol, we combine known results about the diameter of Erdős–Rényi graphs with reductions between different types of random graphs. We further show that the efficiency of our protocol is asymptotically optimal.

The practicality of our protocol is supported by extensive simulations for different numbers of parties, weight distributions, and corruption strategies. The simulations confirm our theoretical results and show that messages are delivered quickly regardless of the weight distribution, whereas protocols that are oblivious of the parties' weights completely fail if the weights are unevenly distributed. Furthermore, the average message complexity per party of our protocol is within a small constant factor of such a protocol.

*Work in part done while the author was at Carnegie Mellon University. Supported in part by the NSF award 1916939, DARPA SIEVE program, a gift from Ripple, a DoE NETL award, a JP Morgan Faculty Fellowship, a PNC center for financial services innovation award, and a Cylab seed funding award.
†Work was in part done while the author was at Purdue University.

# CONTRIBUTIONS

1. *Weighted Fanout Flooding* (WFF):
   - Secure assuming any constant fraction $\gamma$ of *resources* being honest.



# Practical Provably Secure Flooding for Blockchains


Chen-Da Liu-Zhang*[1], Christian Matt[2], Ueli Maurer[3], Guilherme Rito[3], and Søren Eller Thomsen†[4]

[1]NTT Research, USA
chen-da.liuzhang@ntt-research.com
[2]Concordium, Zurich, Switzerland
cm@concordium.com
[3]Department of Computer Science, ETH Zurich, Switzerland
{maurer, gteixeir}@inf.ethz.ch
[4]Concordium Blockchain Research Center, Aarhus University, Denmark
sethomsen@cs.au.dk


September 28, 2022


**Abstract**

In recent years, permisionless blockchains have received a lot of attention both from industry and academia, where substantial effort has been spent to develop consensus protocols that are secure under the assumption that less than half (or a third) of a given resource (e.g., stake or computing power) is controlled by corrupted parties. The security proofs of these consensus protocols usually assume the availability of a network functionality guaranteeing that a block sent by an honest party is received by all honest parties within some bounded time. To obtain an overall protocol that is secure under the same corruption assumption, it is therefore necessary to combine the consensus protocol with a network protocol that achieves this property under that assumption. In practice, however, the underlying network is typically implemented by flooding protocols that are not proven to be secure in the setting where a fraction of the considered total weight can be corrupted. This has led to many so-called eclipse attacks on existing protocols and tailor-made fixes against specific attacks.

To close this apparent gap, we present the first practical flooding protocol that provably delivers sent messages to all honest parties after a logarithmic number of steps. We prove security in the setting where all parties are publicly assigned a positive weight and the adversary can corrupt parties accumulating up to a constant fraction of the total weight. This can directly be used in the proof-of-stake setting, but is not limited to it. To prove the security of our protocol, we combine known results about the diameter of Erdős–Rényi graphs with reductions between different types of random graphs. We further show that the efficiency of our protocol is asymptotically optimal.

The practicality of our protocol is supported by extensive simulations for different numbers of parties, weight distributions, and corruption strategies. The simulations confirm our theoretical results and show that messages are delivered quickly regardless of the weight distribution, whereas protocols that are oblivious of the parties' weights completely fail if the weights are unevenly distributed. Furthermore, the average message complexity per party of our protocol is within a small constant factor of such a protocol.

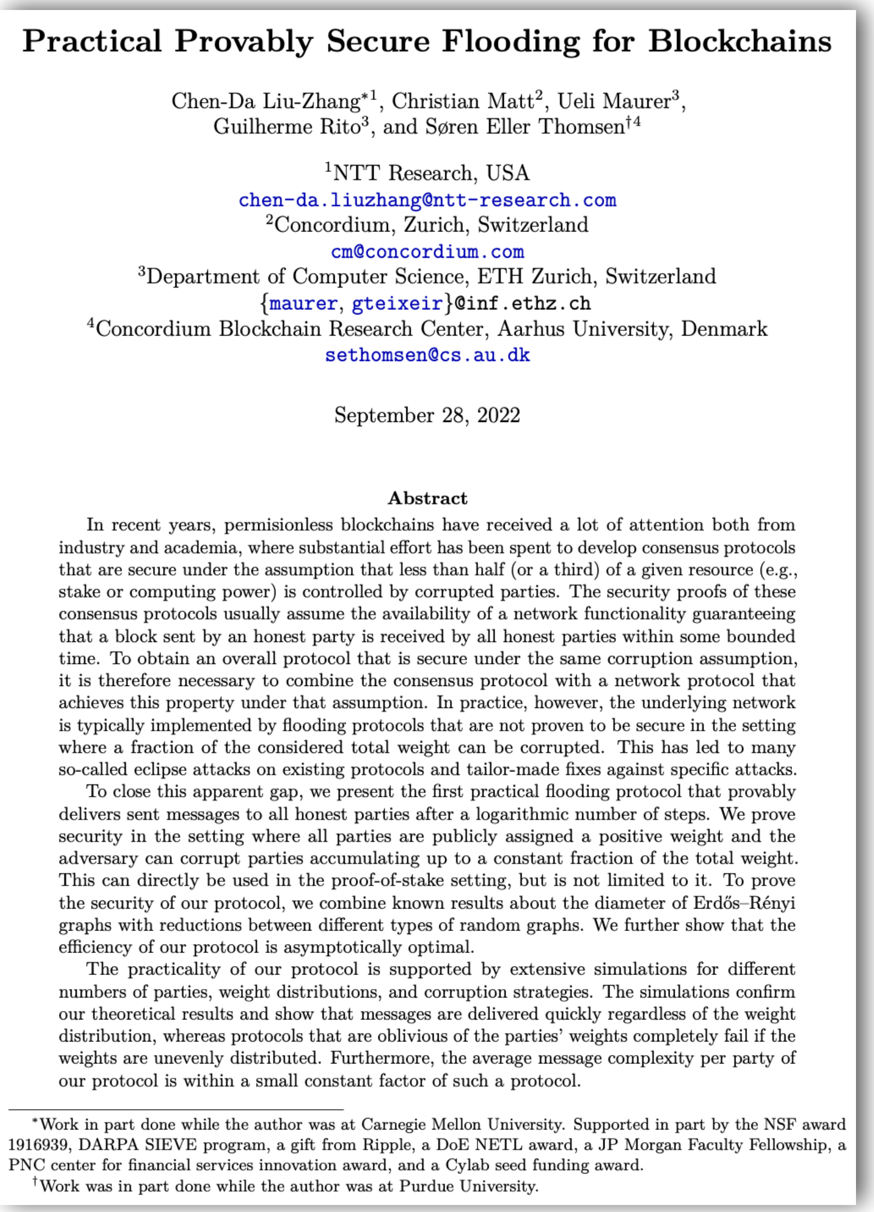


*Work in part done while the author was at Carnegie Mellon University. Supported in part by the NSF award 1916939, DARPA SIEVE program, a gift from Ripple, a DoE NETL award, a JP Morgan Faculty Fellowship, a PNC center for financial services innovation award, and a Cylab seed funding award.


†Work was in part done while the author was at Purdue University.

# CONTRIBUTIONS

1. *Weighted Fanout Flooding* (WFF):
    - Secure assuming any constant fraction $\gamma$ of *resources* being honest.
    - Diameter: $O\big(\log(n)\big)$ for $n$ parties.



## Practical Provably Secure Flooding for Blockchains


Chen-Da Liu-Zhang*[1], Christian Matt[2], Ueli Maurer[3], Guilherme Rito[3], and Søren Eller Thomsen†[4]

[1]NTT Research, USA
chen-da.liuzhang@ntt-research.com
[2]Concordium, Zurich, Switzerland
cm@concordium.com
[3]Department of Computer Science, ETH Zurich, Switzerland
{maurer, gteixeir}@inf.ethz.ch
[4]Concordium Blockchain Research Center, Aarhus University, Denmark
sethomsen@cs.au.dk


September 28, 2022


**Abstract**

In recent years, permisionless blockchains have received a lot of attention both from industry and academia, where substantial effort has been spent to develop consensus protocols that are secure under the assumption that less than half (or a third) of a given resource (e.g., stake or computing power) is controlled by corrupted parties. The security proofs of these consensus protocols usually assume the availability of a network functionality guaranteeing that a block sent by an honest party is received by all honest parties within some bounded time. To obtain an overall protocol that is secure under the same corruption assumption, it is therefore necessary to combine the consensus protocol with a network protocol that achieves this property under that assumption. In practice, however, the underlying network is typically implemented by flooding protocols that are not proven to be secure in the setting where a fraction of the considered total weight can be corrupted. This has led to many so-called eclipse attacks on existing protocols and tailor-made fixes against specific attacks.

To close this apparent gap, we present the first practical flooding protocol that provably delivers sent messages to all honest parties after a logarithmic number of steps. We prove security in the setting where all parties are publicly assigned a positive weight and the adversary can corrupt parties accumulating up to a constant fraction of the total weight. This can directly be used in the proof-of-stake setting, but is not limited to it. To prove the security of our protocol, we combine known results about the diameter of Erdős–Rényi graphs with reductions between different types of random graphs. We further show that the efficiency of our protocol is asymptotically optimal.

The practicality of our protocol is supported by extensive simulations for different numbers of parties, weight distributions, and corruption strategies. The simulations confirm our theoretical results and show that messages are delivered quickly regardless of the weight distribution, whereas protocols that are oblivious of the parties' weights completely fail if the weights are unevenly distributed. Furthermore, the average message complexity per party of our protocol is within a small constant factor of such a protocol.

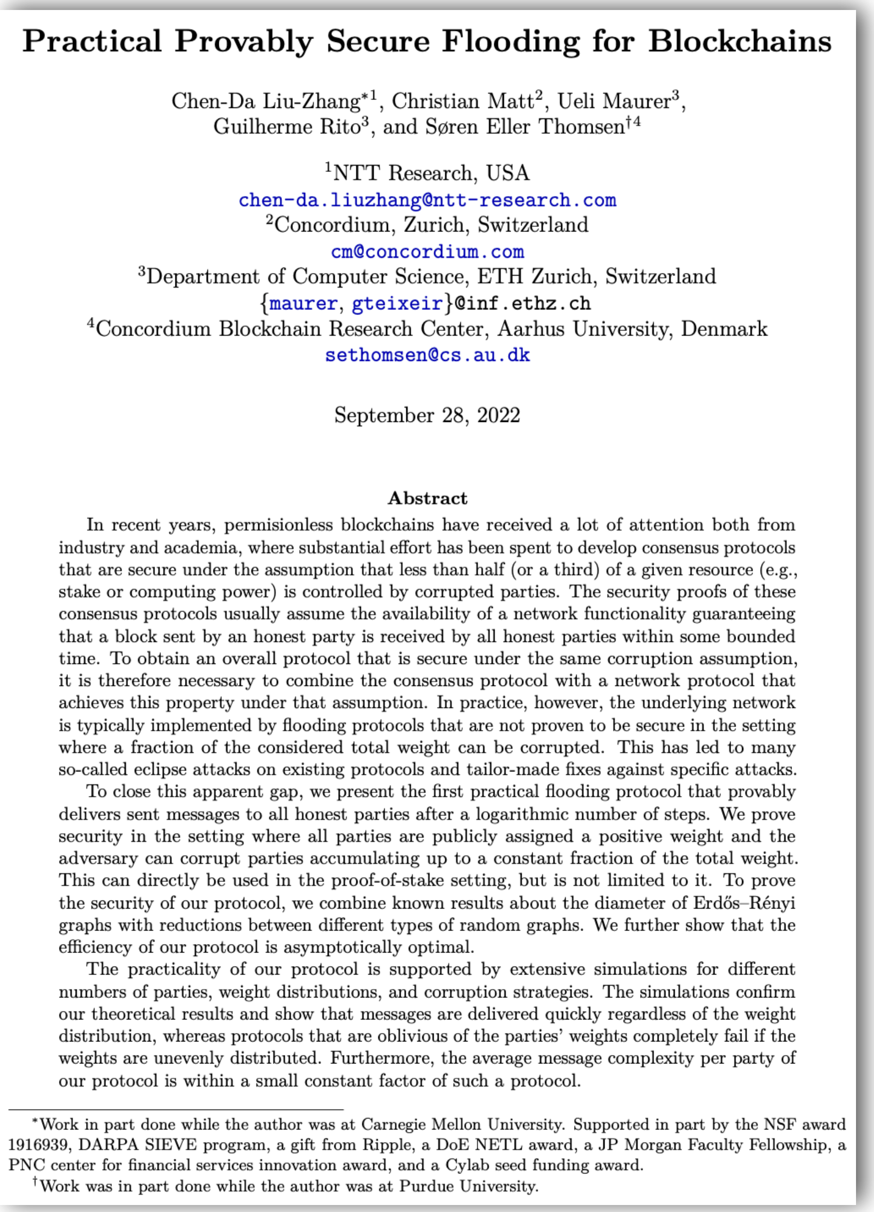


*Work in part done while the author was at Carnegie Mellon University. Supported in part by the NSF award 1916939, DARPA SIEVE program, a gift from Ripple, a DoE NETL award, a JP Morgan Faculty Fellowship, a PNC center for financial services innovation award, and a Cylab seed funding award.

†Work was in part done while the author was at Purdue University.

# CONTRIBUTIONS

1. *Weighted Fanout Flooding* (WFF):
   - Secure assuming any constant fraction $\gamma$ of *resources* being honest.
   - Diameter: $O\big(\log(n)\big)$ for $n$ parties.
   - Message complexity: $O\big(n \cdot \gamma^{-1} \cdot (\log(n) + \kappa)\big)$.



## Practical Provably Secure Flooding for Blockchains


Chen-Da Liu-Zhang*[1], Christian Matt[2], Ueli Maurer[3], Guilherme Rito[3], and Søren Eller Thomsen†[4]

[1]NTT Research, USA
chen-da.liuzhang@ntt-research.com
[2]Concordium, Zurich, Switzerland
cm@concordium.com
[3]Department of Computer Science, ETH Zurich, Switzerland
{maurer, gteixeir}@inf.ethz.ch
[4]Concordium Blockchain Research Center, Aarhus University, Denmark
sethomsen@cs.au.dk


September 28, 2022


**Abstract**

In recent years, permisionless blockchains have received a lot of attention both from industry and academia, where substantial effort has been spent to develop consensus protocols that are secure under the assumption that less than half (or a third) of a given resource (e.g., stake or computing power) is controlled by corrupted parties. The security proofs of these consensus protocols usually assume the availability of a network functionality guaranteeing that a block sent by an honest party is received by all honest parties within some bounded time. To obtain an overall protocol that is secure under the same corruption assumption, it is therefore necessary to combine the consensus protocol with a network protocol that achieves this property under that assumption. In practice, however, the underlying network is typically implemented by flooding protocols that are not proven to be secure in the setting where a fraction of the considered total weight can be corrupted. This has led to many so-called eclipse attacks on existing protocols and tailor-made fixes against specific attacks.

To close this apparent gap, we present the first practical flooding protocol that provably delivers sent messages to all honest parties after a logarithmic number of steps. We prove security in the setting where all parties are publicly assigned a positive weight and the adversary can corrupt parties accumulating up to a constant fraction of the total weight. This can directly be used in the proof-of-stake setting, but is not limited to it. To prove the security of our protocol, we combine known results about the diameter of Erdős–Rényi graphs with reductions between different types of random graphs. We further show that the efficiency of our protocol is asymptotically optimal.

The practicality of our protocol is supported by extensive simulations for different numbers of parties, weight distributions, and corruption strategies. The simulations confirm our theoretical results and show that messages are delivered quickly regardless of the weight distribution, whereas protocols that are oblivious of the parties' weights completely fail if the weights are unevenly distributed. Furthermore, the average message complexity per party of our protocol is within a small constant factor of such a protocol.

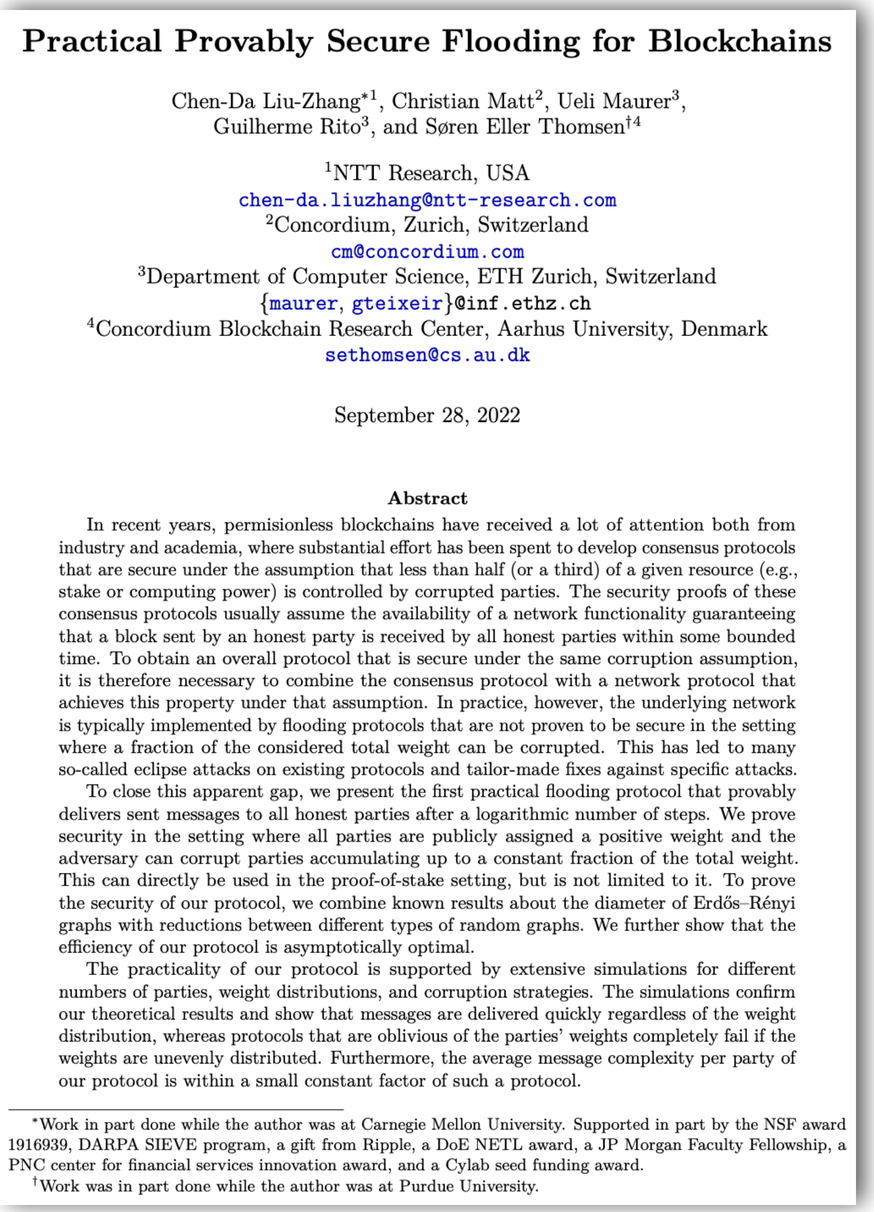


*Work in part done while the author was at Carnegie Mellon University. Supported in part by the NSF award 1916939, DARPA SIEVE program, a gift from Ripple, a DoE NETL award, a JP Morgan Faculty Fellowship, a PNC center for financial services innovation award, and a Cylab seed funding award.

†Work was in part done while the author was at Purdue University.

# CONTRIBUTIONS

1. *Weighted Fanout Flooding* (WFF):
   - Secure assuming any constant fraction $\gamma$ of *resources* being honest.
   - Diameter: $O\big(\log(n)\big)$ for $n$ parties.
   - Message complexity: $O\big(n \cdot \gamma^{-1} \cdot (\log(n) + \kappa)\big)$.
2. Extensive simulations of WFF.



## Practical Provably Secure Flooding for Blockchains


Chen-Da Liu-Zhang*[1], Christian Matt[2], Ueli Maurer[3], Guilherme Rito[3], and Søren Eller Thomsen†[4]

[1]NTT Research, USA
chen-da.liuzhang@ntt-research.com
[2]Concordium, Zurich, Switzerland
cm@concordium.com
[3]Department of Computer Science, ETH Zurich, Switzerland
{maurer, gteixeir}@inf.ethz.ch
[4]Concordium Blockchain Research Center, Aarhus University, Denmark
sethomsen@cs.au.dk


September 28, 2022


**Abstract**

In recent years, permisionless blockchains have received a lot of attention both from industry and academia, where substantial effort has been spent to develop consensus protocols that are secure under the assumption that less than half (or a third) of a given resource (e.g., stake or computing power) is controlled by corrupted parties. The security proofs of these consensus protocols usually assume the availability of a network functionality guaranteeing that a block sent by an honest party is received by all honest parties within some bounded time. To obtain an overall protocol that is secure under the same corruption assumption, it is therefore necessary to combine the consensus protocol with a network protocol that achieves this property under that assumption. In practice, however, the underlying network is typically implemented by flooding protocols that are not proven to be secure in the setting where a fraction of the considered total weight can be corrupted. This has led to many so-called eclipse attacks on existing protocols and tailor-made fixes against specific attacks.

To close this apparent gap, we present the first practical flooding protocol that provably delivers sent messages to all honest parties after a logarithmic number of steps. We prove security in the setting where all parties are publicly assigned a positive weight and the adversary can corrupt parties accumulating up to a constant fraction of the total weight. This can directly be used in the proof-of-stake setting, but is not limited to it. To prove the security of our protocol, we combine known results about the diameter of Erdős–Rényi graphs with reductions between different types of random graphs. We further show that the efficiency of our protocol is asymptotically optimal.

The practicality of our protocol is supported by extensive simulations for different numbers of parties, weight distributions, and corruption strategies. The simulations confirm our theoretical results and show that messages are delivered quickly regardless of the weight distribution, whereas protocols that are oblivious of the parties' weights completely fail if the weights are unevenly distributed. Furthermore, the average message complexity per party of our protocol is within a small constant factor of such a protocol.



*Work in part done while the author was at Carnegie Mellon University. Supported in part by the NSF award 1916939, DARPA SIEVE program, a gift from Ripple, a DoE NETL award, a JP Morgan Faculty Fellowship, a PNC center for financial services innovation award, and a Cylab seed funding award.

†Work was in part done while the author was at Purdue University.

# CONTRIBUTIONS

1. *Weighted Fanout Flooding* (WFF):
   - Secure assuming any constant fraction $\gamma$ of *resources* being honest.
   - Diameter: $O\big(\log(n)\big)$ for $n$ parties.
   - Message complexity: $O\big(n \cdot \gamma^{-1} \cdot (\log(n) + \kappa)\big)$.
2. Extensive simulations of WFF.
   - Confirms practicality protocol.

## Practical Provably Secure Flooding for Blockchains


Chen-Da Liu-Zhang*[1], Christian Matt[2], Ueli Maurer[3], Guilherme Rito[3], and Søren Eller Thomsen†[4]

[1]NTT Research, USA
chen-da.liuzhang@ntt-research.com
[2]Concordium, Zurich, Switzerland
cm@concordium.com
[3]Department of Computer Science, ETH Zurich, Switzerland
{maurer, gteixeir}@inf.ethz.ch
[4]Concordium Blockchain Research Center, Aarhus University, Denmark
sethomsen@cs.au.dk


September 28, 2022


**Abstract**

In recent years, permisionless blockchains have received a lot of attention both from industry and academia, where substantial effort has been spent to develop consensus protocols that are secure under the assumption that less than half (or a third) of a given resource (e.g., stake or computing power) is controlled by corrupted parties. The security proofs of these consensus protocols usually assume the availability of a network functionality guaranteeing that a block sent by an honest party is received by all honest parties within some bounded time. To obtain an overall protocol that is secure under the same corruption assumption, it is therefore necessary to combine the consensus protocol with a network protocol that achieves this property under that assumption. In practice, however, the underlying network is typically implemented by flooding protocols that are not proven to be secure in the setting where a fraction of the considered total weight can be corrupted. This has led to many so-called eclipse attacks on existing protocols and tailor-made fixes against specific attacks.

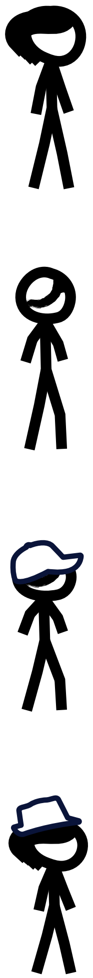
To close this apparent gap, we present the first practical flooding protocol that provably delivers sent messages to all honest parties after a logarithmic number of steps. We prove security in the setting where all parties are publicly assigned a positive weight and the adversary can corrupt parties accumulating up to a constant fraction of the total weight. This can directly be used in the proof-of-stake setting, but is not limited to it. To prove the security of our protocol, we combine known results about the diameter of Erdős–Rényi graphs with reductions between different types of random graphs. We further show that the efficiency of our protocol is asymptotically optimal.

The practicality of our protocol is supported by extensive simulations for different numbers of parties, weight distributions, and corruption strategies. The simulations confirm our theoretical results and show that messages are delivered quickly regardless of the weight distribution, whereas protocols that are oblivious of the parties' weights completely fail if the weights are unevenly distributed. Furthermore, the average message complexity per party of our protocol is within a small constant factor of such a protocol.

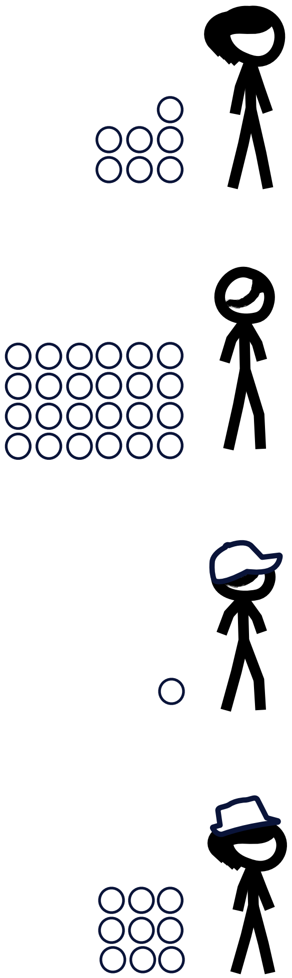


*Work in part done while the author was at Carnegie Mellon University. Supported in part by the NSF award 1916939, DARPA SIEVE program, a gift from Ripple, a DoE NETL award, a JP Morgan Faculty Fellowship, a PNC center for financial services innovation award, and a Cylab seed funding award.
†Work was in part done while the author was at Purdue University.

# MODEL

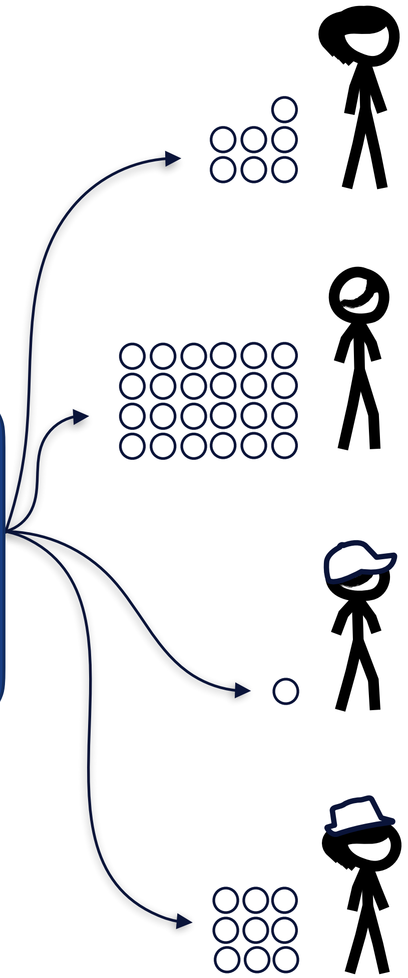


# MODEL



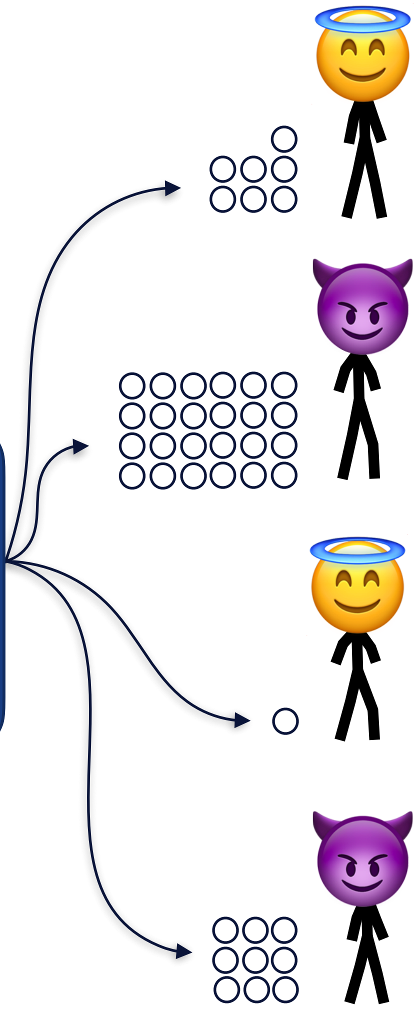
# MODEL

Each party $p_i$ has a publicly known weight $w_i > 0$.



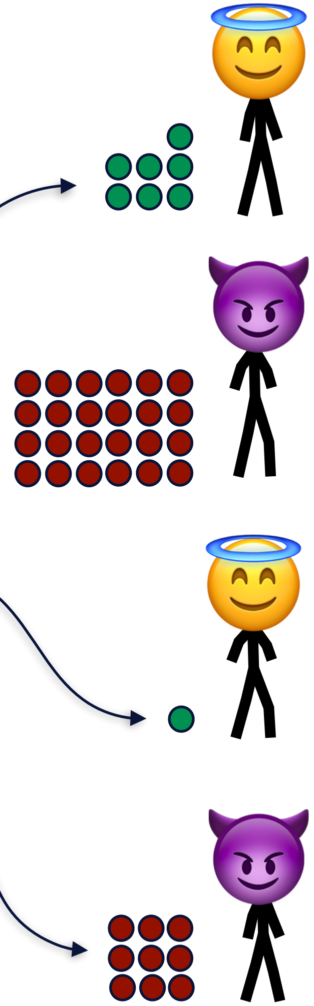
# MODEL

Each party $p_i$ has a publicly known weight $w_i > 0$.



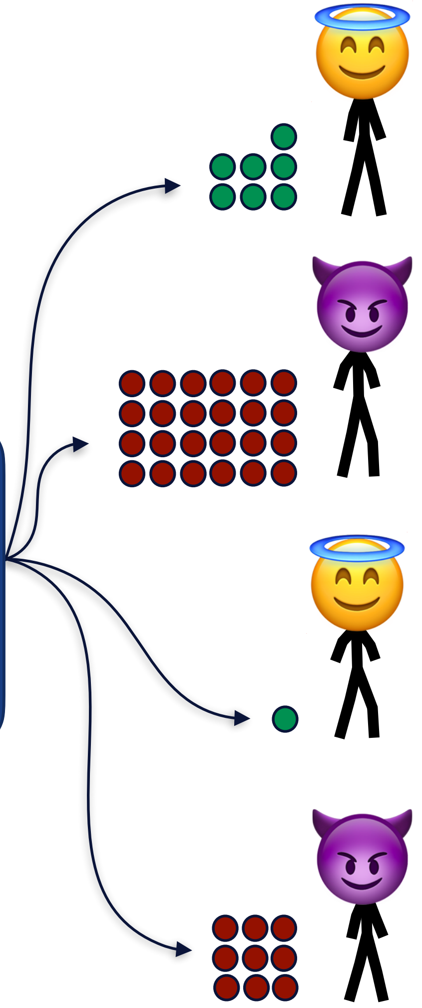
# MODEL

Each party $p_i$ has a publicly known weight $w_i > 0$.



# MODEL

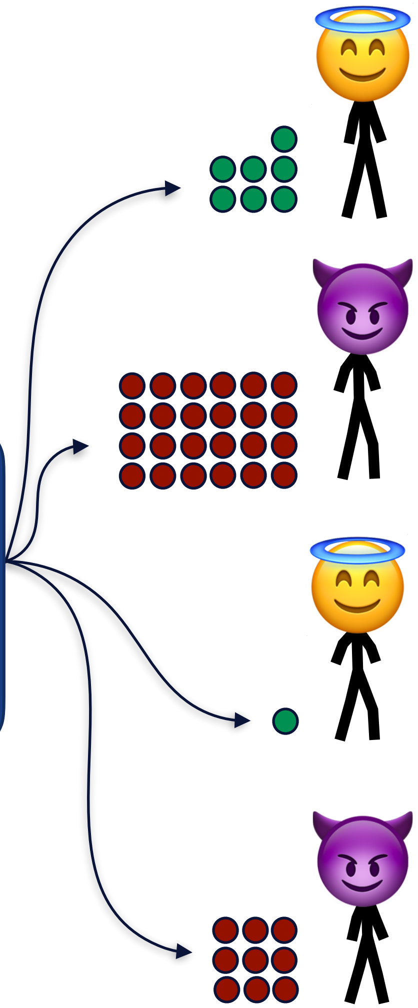
Each party $p_i$ has a publicly known weight $w_i > 0$.



**Assumption:** $\exists \gamma \in (0,1]$, *s.t.*

$$\# \bullet \geq \gamma \cdot ( \# \bullet + \# \bullet ).$$

# MODEL



Each party $p_i$ has a publicly known weight $w_i > 0$.

**Assumption:** $\exists \gamma \in (0,1]$, *s.t.*

$$\# \textcolor{green}{\bullet} \geq \gamma \cdot ( \# \textcolor{green}{\bullet} + \# \textcolor{red}{\bullet} ).$$

Implied by the standard PoS assumption.

# WARMUP: A SIMPLE INEFFICIENT SOLUTION

# WARMUP: A SIMPLE INEFFICIENT SOLUTION

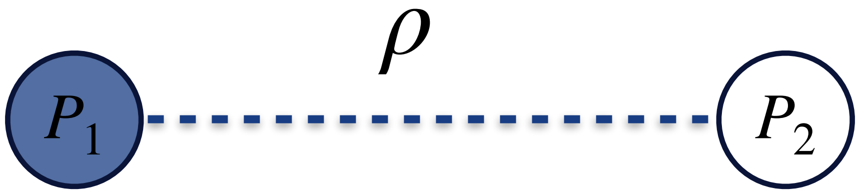
💡 Use existing flooding protocol where parties behave proportionally to their weight.

# WARMUP: A SIMPLE INEFFICIENT SOLUTION

- 💡 Use existing flooding protocol where parties behave proportionally to their weight.
- 💡 [MNT22]: "Forward to each party with a probability $\rho$" ensures logarithmic diameter.

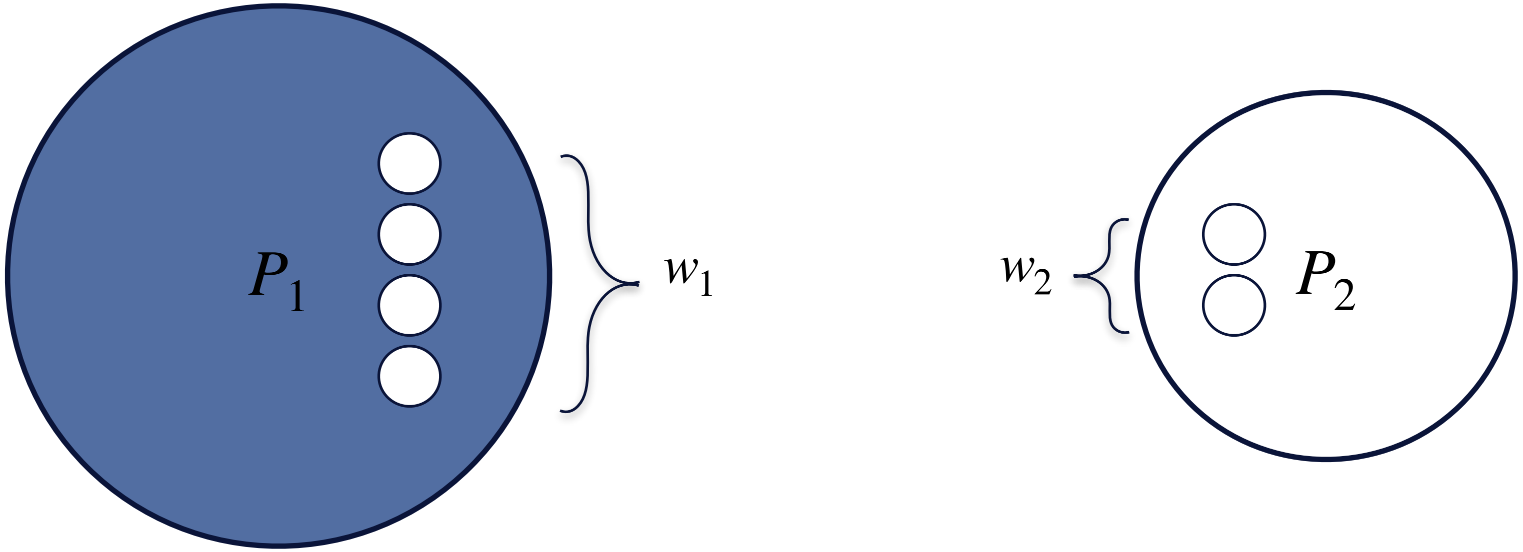
# WARMUP: A SIMPLE INEFFICIENT SOLUTION

- Use existing flooding protocol where parties behave proportionally to their weight.
- [MNT22]: "Forward to each party with a probability $\rho$" ensures logarithmic diameter.



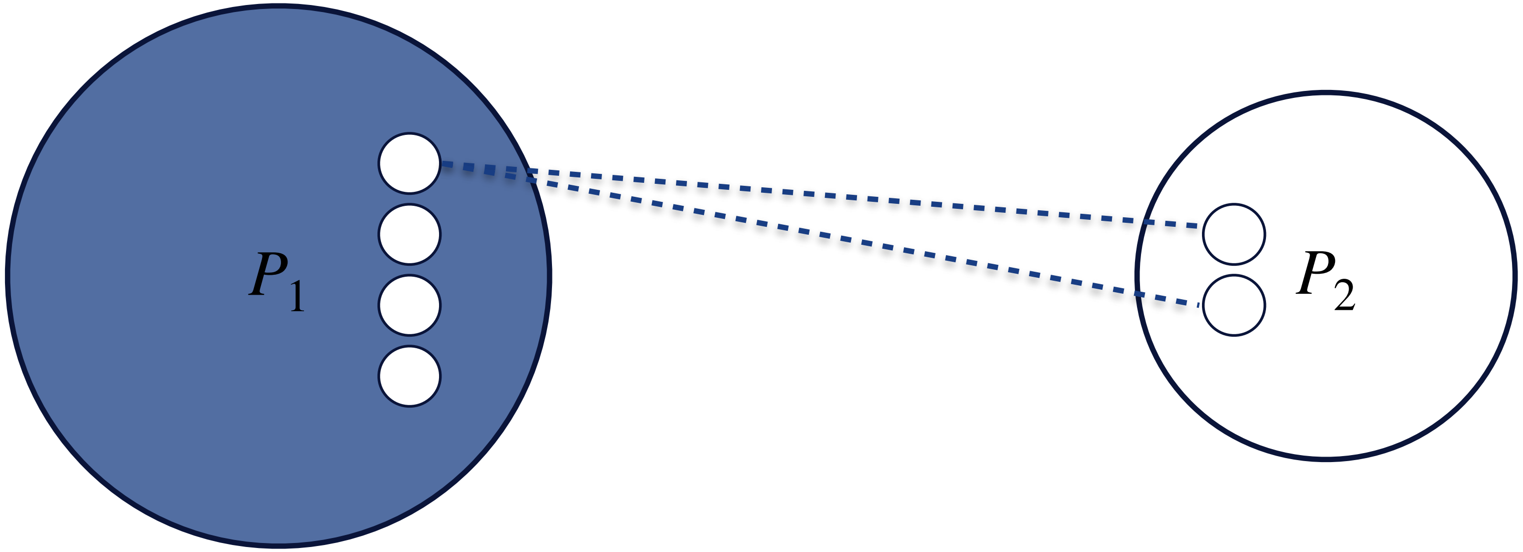
# WARMUP: A SIMPLE INEFFICIENT SOLUTION

- Use existing flooding protocol where parties behave proportionally to their weight.
- [MNT22]: "Forward to each party with a probability $\rho$" ensures logarithmic diameter.



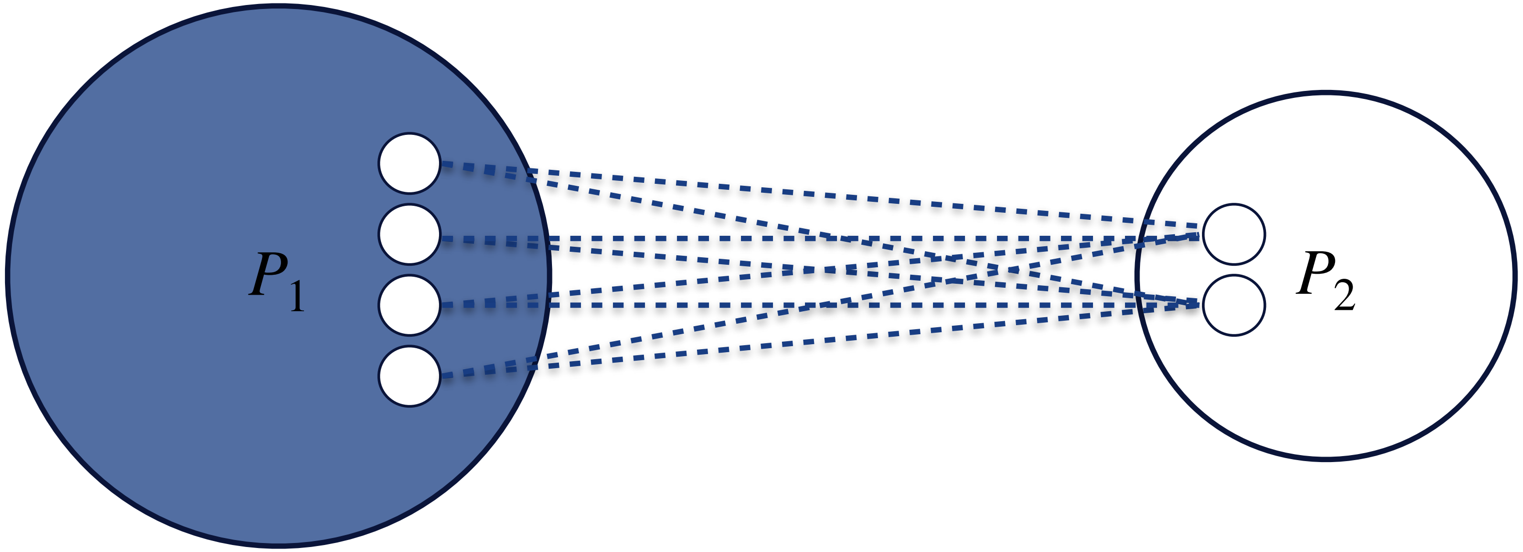
# WARMUP: A SIMPLE INEFFICIENT SOLUTION

- 💡 Use existing flooding protocol where parties behave proportionally to their weight.
- 💡 [MNT22]: “Forward to each party with a probability $\rho$” ensures logarithmic diameter.



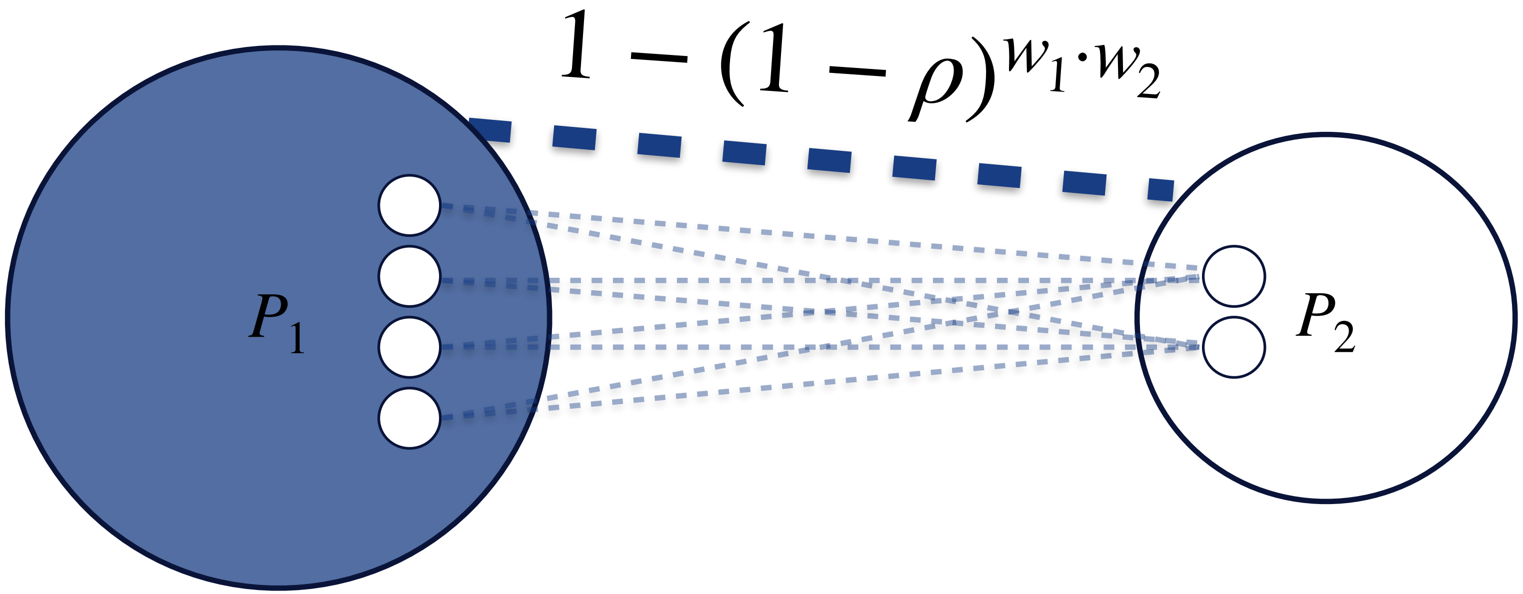
# WARMUP: A SIMPLE INEFFICIENT SOLUTION

- Use existing flooding protocol where parties behave proportionally to their weight.
- [MNT22]: "Forward to each party with a probability $\rho$" ensures logarithmic diameter.



# WARMUP: A SIMPLE INEFFICIENT SOLUTION

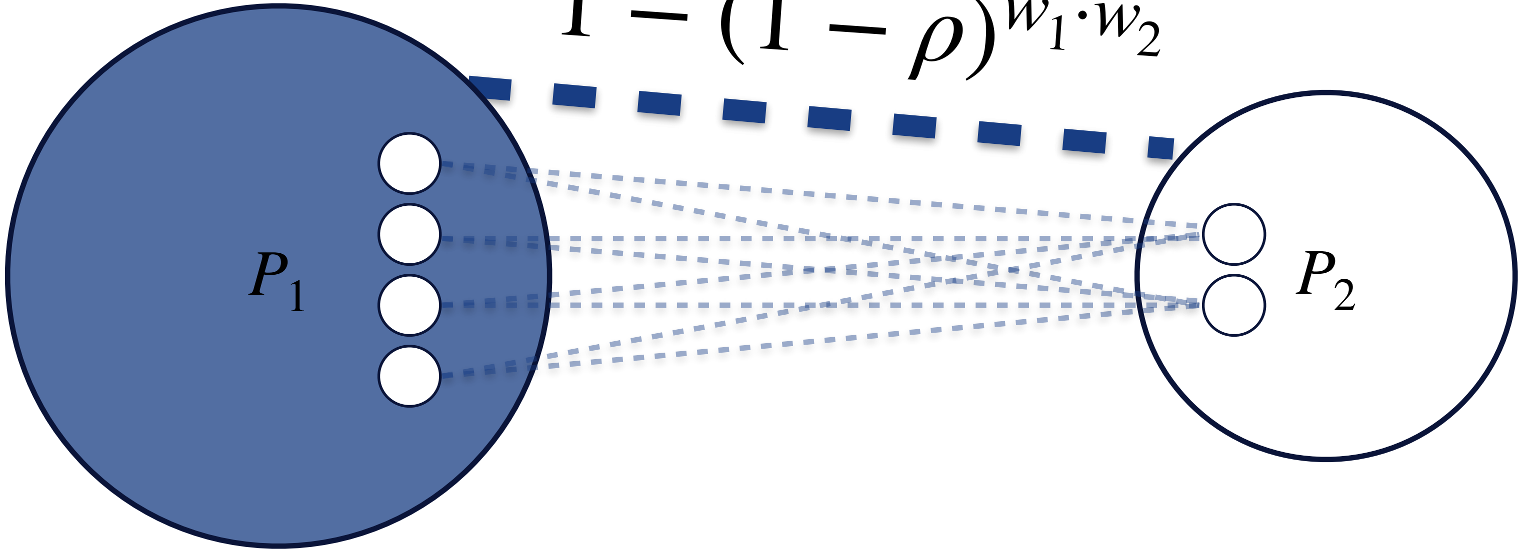
- 💡 Use existing flooding protocol where parties behave proportionally to their weight.
- 💡 [MNT22]: "Forward to each party with a probability $\rho$" ensures logarithmic diameter.



# WARMUP: A SIMPLE INEFFICIENT SOLUTION

**Wanted: Scaling invariance!**

$$1 - (1 - \rho)^{w_1 \cdot w_2}$$

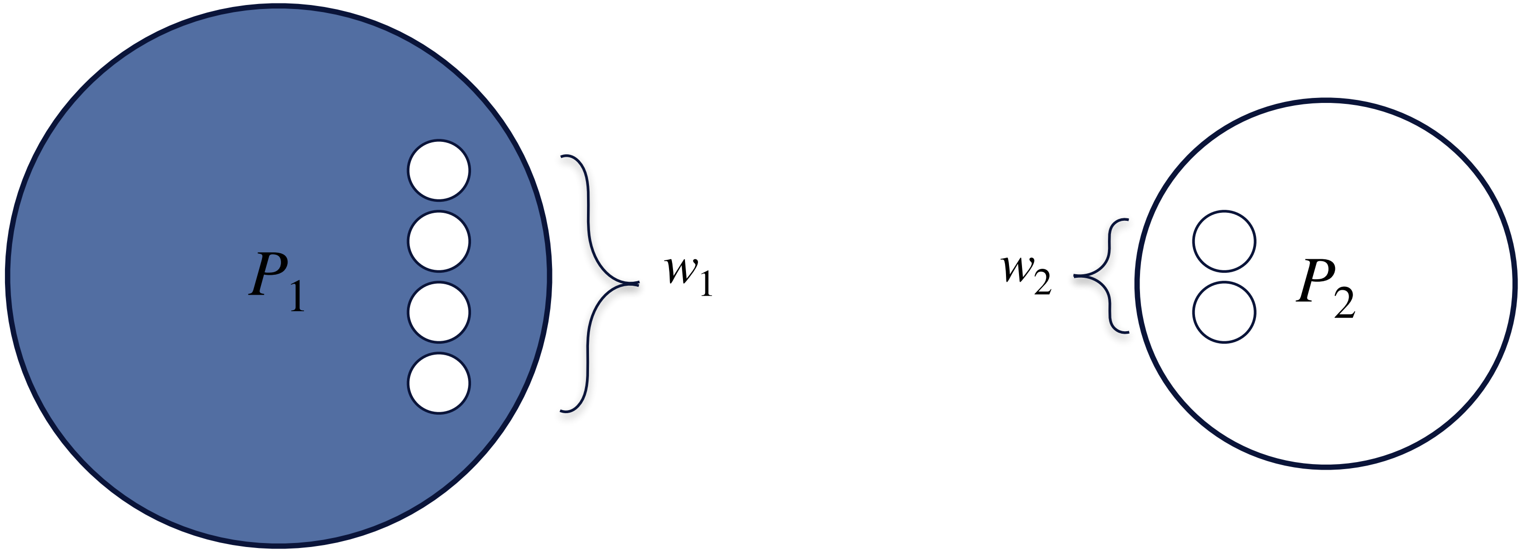


# DEVELOPING THE IDEA

💡 A function $E(p)$ that determines how many nodes each party should emulate.

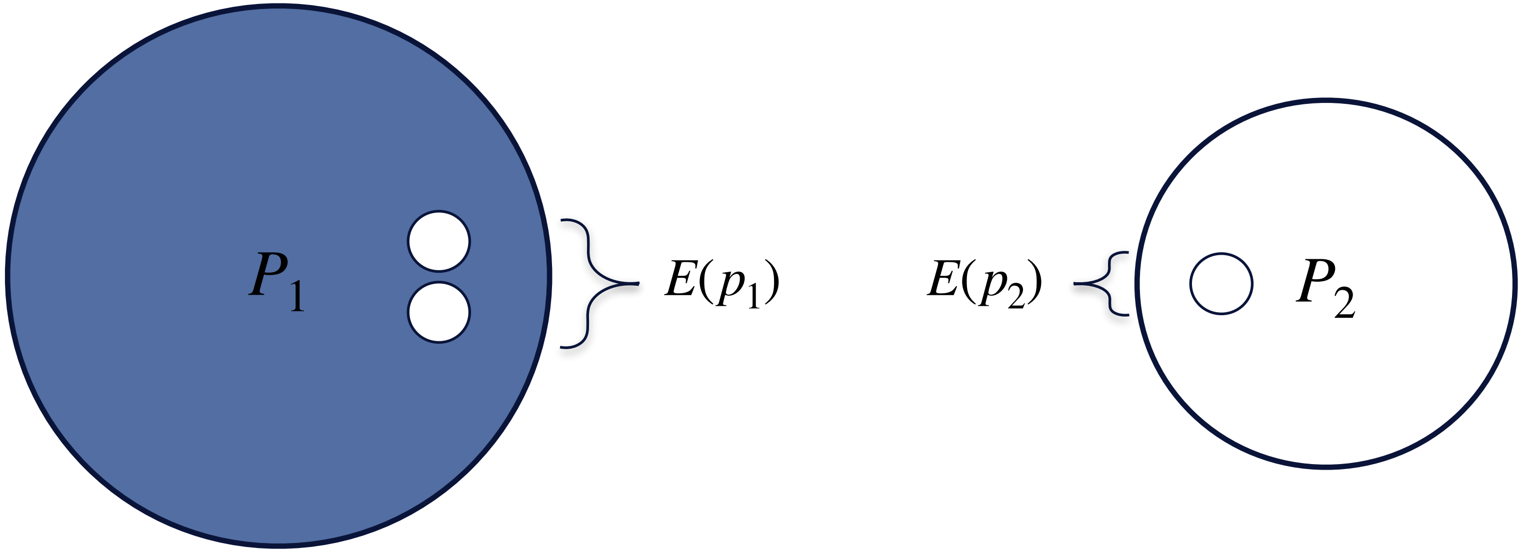
# DEVELOPING THE IDEA

💡 A function $E(p)$ that determines how many nodes each party should emulate.



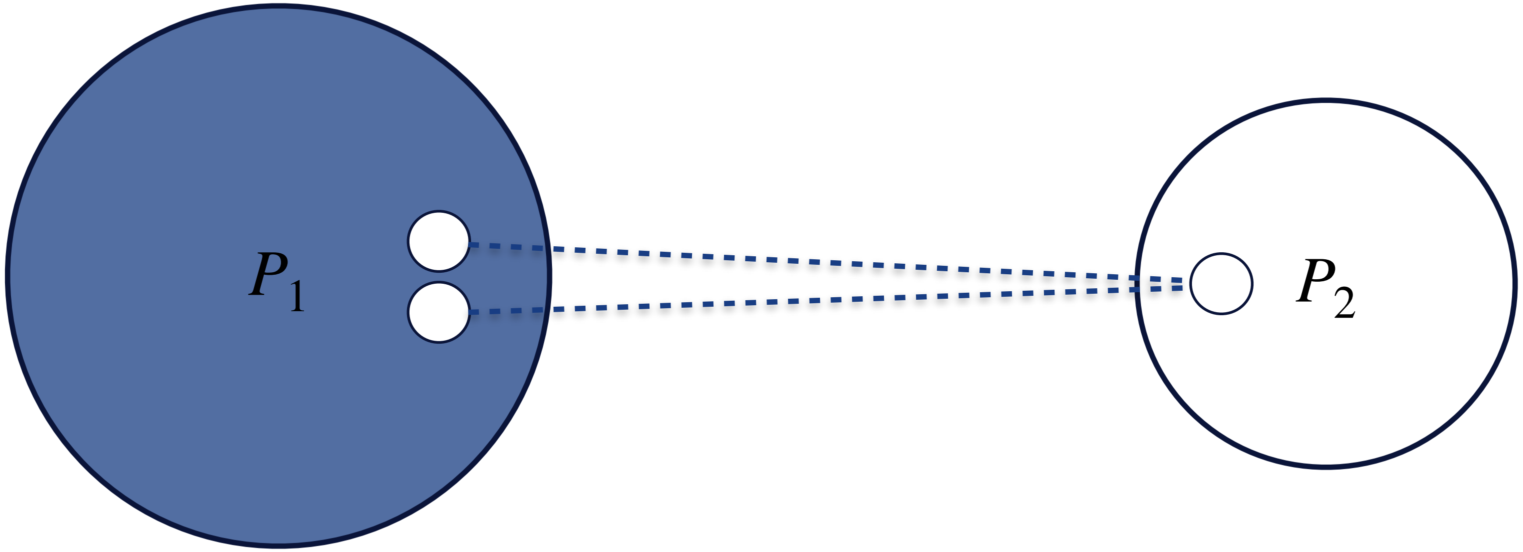
# DEVELOPING THE IDEA

💡 A function $E(p)$ that determines how many nodes each party should emulate.



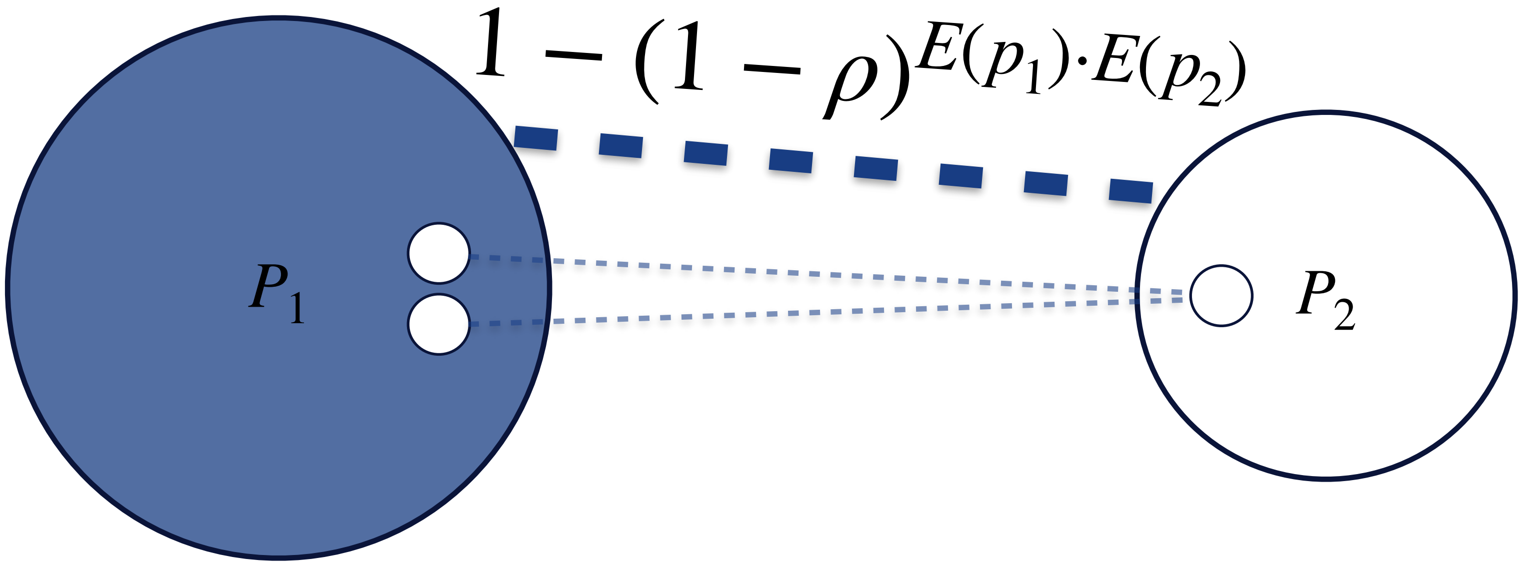
# DEVELOPING THE IDEA

💡 A function $E(p)$ that determines how many nodes each party should emulate.



# DEVELOPING THE IDEA

💡 A function $E(p)$ that determines how many nodes each party should emulate.



# PROPERTIES OF A GOOD EMULATION FUNCTION

# PROPERTIES OF A GOOD EMULATION FUNCTION

- Invariant to scaling of weights.

# PROPERTIES OF A GOOD EMULATION FUNCTION

- Invariant to scaling of weights.
- Any party should emulate at least one node.

# PROPERTIES OF A GOOD EMULATION FUNCTION

- Invariant to scaling of weights.
- Any party should emulate at least one node.

Message complexity of [MNT22] is linear in $n$ and $\gamma^{-1}$.

# PROPERTIES OF A GOOD EMULATION FUNCTION

- Invariant to scaling of weights.
- Any party should emulate at least one node.
- Number of emulated nodes should be low.

Message complexity of [MNT22] is linear in $n$ and $\gamma^{-1}$.

# PROPERTIES OF A GOOD EMULATION FUNCTION

- Invariant to scaling of weights.
- Any party should emulate at least one node.
- Number of emulated nodes should be low.
- Fraction of honestly emulated nodes should be high.

Message complexity of [MNT22] is linear in $n$ and $\gamma^{-1}$.

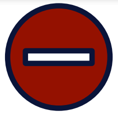
# CANDIDATES?

| | |
|---|---|
| ‣ Invariant to scaling of weight. | |
| ‣ Any party should emulate at least one node. | |
| ‣ Number of emulated nodes should be low. | |
| ‣ Fraction of honestly emulated nodes should be high. | |

# CANDIDATES?

| | $E(p) \triangleq w_p$ |
|---|---|
| ‣ Invariant to scaling of weight. | |
| ‣ Any party should emulate at least one node. | |
| ‣ Number of emulated nodes should be low. | |
| ‣ Fraction of honestly emulated nodes should be high. | |

# CANDIDATES?

| | $E(p) \triangleq w_p$ |
|---|---|
| ‣ Invariant to scaling of weight. |  |
| ‣ Any party should emulate at least one node. | |
| ‣ Number of emulated nodes should be low. | |
| ‣ Fraction of honestly emulated nodes should be high. | |

# CANDIDATES?

| | $E(p) \triangleq \alpha_p$ |
|---|---|
| ‣ Invariant to scaling of weight. | |
| ‣ Any party should emulate at least one node. | |
| ‣ Number of emulated nodes should be low. | |
| ‣ Fraction of honestly emulated nodes should be high. | |

# CANDIDATES?

Fraction of weight owned by party $p$.

| | $E(p) \triangleq \alpha_p$ |
|---|---|
| ‣ Invariant to scaling of weight. | |
| ‣ Any party should emulate at least one node. | |
| ‣ Number of emulated nodes should be low. | |
| ‣ Fraction of honestly emulated nodes should be high. | |

# CANDIDATES?

Fraction of weight owned by party $p$.

| | $E(p) \triangleq \alpha_p$ |
|---|---|
| ‣ Invariant to scaling of weight. | ✓ |
| ‣ Any party should emulate at least one node. | |
| ‣ Number of emulated nodes should be low. | |
| ‣ Fraction of honestly emulated nodes should be high. | |

# CANDIDATES?

Fraction of weight owned by party $p$.

| | $E(p) \triangleq \alpha_p$ |
|---|---|
| ‣ Invariant to scaling of weight. | ✓ |
| ‣ Any party should emulate at least one node. | ⛔ |
| ‣ Number of emulated nodes should be low. | |
| ‣ Fraction of honestly emulated nodes should be high. | |

# CANDIDATES?

| | $E(p) \triangleq \lceil \alpha_p \rceil$ |
|---|---|
| ‣ Invariant to scaling of weight. | |
| ‣ Any party should emulate at least one node. | |
| ‣ Number of emulated nodes should be low. | |
| ‣ Fraction of honestly emulated nodes should be high. | |

# CANDIDATES?

| | $E(p) \triangleq \lceil \alpha_p \rceil$ |
|---|---|
| ‣ Invariant to scaling of weight. | ✓ |
| ‣ Any party should emulate at least one node. | |
| ‣ Number of emulated nodes should be low. | |
| ‣ Fraction of honestly emulated nodes should be high. | |

# CANDIDATES?

| | $E(p) \triangleq \lceil \alpha_p \rceil$ |
|---|---|
| ‣ Invariant to scaling of weight. | ✓ |
| ‣ Any party should emulate at least one node. | ✓ |
| ‣ Number of emulated nodes should be low. | |
| ‣ Fraction of honestly emulated nodes should be high. | |

# CANDIDATES?

| | $E(p) \triangleq \lceil \alpha_p \rceil$ |
|---|---|
| ‣ Invariant to scaling of weight. | ✓ |
| ‣ Any party should emulate at least one node. | ✓ |
| ‣ Number of emulated nodes should be low. | ✓ |
| ‣ Fraction of honestly emulated nodes should be high. | |

# CANDIDATES?

| | $E(p) \triangleq \lceil \alpha_p \rceil$ |
|---|---|
| ‣ Invariant to scaling of weight. | ✓ |
| ‣ Any party should emulate at least one node. | ✓ |
| ‣ Number of emulated nodes should be low. | ✓ |
| ‣ Fraction of honestly emulated nodes should be high. | ⛔ |

# CANDIDATES?

| | $E(p) \triangleq 1$ |
|---|---|
| ‣ Invariant to scaling of weight. | ✓ |
| ‣ Any party should emulate at least one node. | ✓ |
| ‣ Number of emulated nodes should be low. | ✓ |
| ‣ Fraction of honestly emulated nodes should be high. | |

# CANDIDATES?

| | $E(p) \triangleq \lceil \alpha_p \cdot n \rceil$ |
|---|---|
| ‣ Invariant to scaling of weight. | |
| ‣ Any party should emulate at least one node. | |
| ‣ Number of emulated nodes should be low. | |
| ‣ Fraction of honestly emulated nodes should be high. | |

# CANDIDATES?

| | $E(p) \triangleq \lceil \alpha_p \cdot n \rceil$ |
|---|---|
| ‣ Invariant to scaling of weight. | ✔ |
| ‣ Any party should emulate at least one node. | |
| ‣ Number of emulated nodes should be low. | |
| ‣ Fraction of honestly emulated nodes should be high. | |

# CANDIDATES?

| | $E(p) \triangleq \lceil \alpha_p \cdot n \rceil$ |
|---|---|
| ‣ Invariant to scaling of weight. | ✓ |
| ‣ Any party should emulate at least one node. | ✓ |
| ‣ Number of emulated nodes should be low. | |
| ‣ Fraction of honestly emulated nodes should be high. | |

# CANDIDATES?

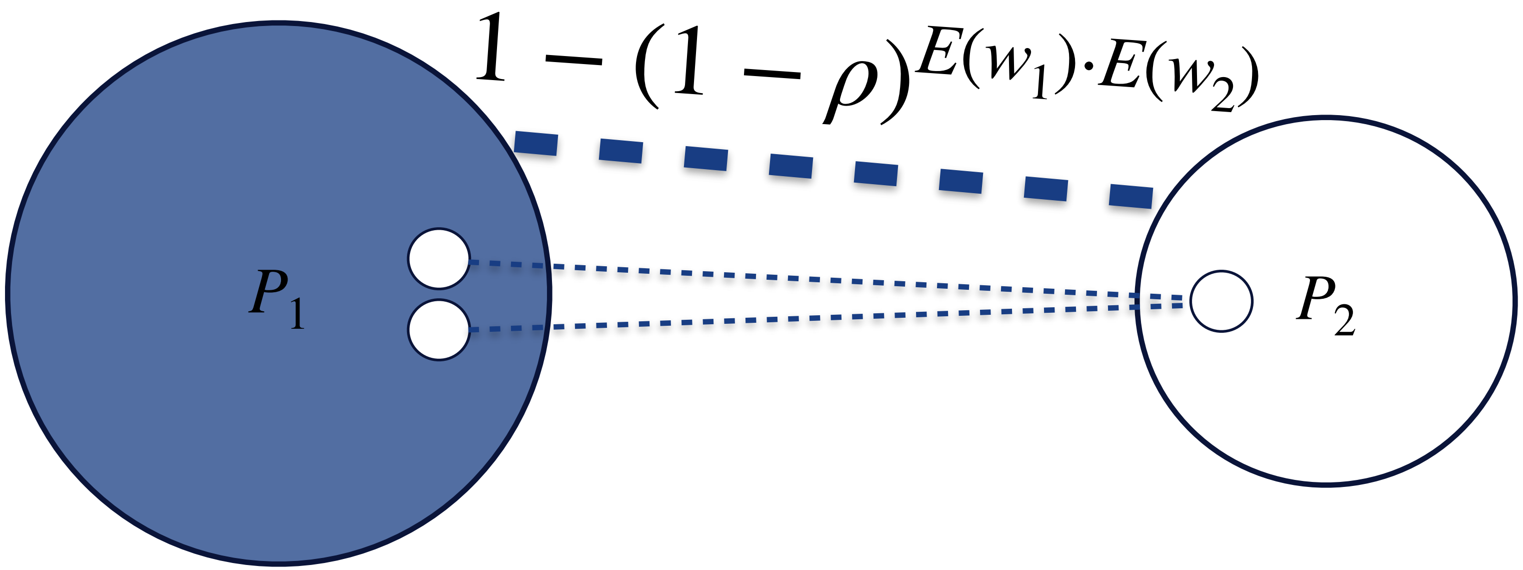
| | $E(p) \triangleq \lceil \alpha_p \cdot n \rceil$ |
|---|---|
| ‣ Invariant to scaling of weight. | ✓ |
| ‣ Any party should emulate at least one node. | ✓ |
| ‣ Number of emulated nodes should be low. | ✓ $( \leq 2 \cdot n)$ |
| ‣ Fraction of honestly emulated nodes should be high. | |

# CANDIDATES?

| | $E(p) \triangleq \lceil \alpha_p \cdot n \rceil$ |
|---|---|
| ‣ Invariant to scaling of weight. | ✔ |
| ‣ Any party should emulate at least one node. | ✔ |
| ‣ Number of emulated nodes should be low. | ✔ $( \leq 2 \cdot n)$ |
| ‣ Fraction of honestly emulated nodes should be high. | ✔ $( \geq 2^{-1} \cdot \gamma)$ |

# A FEW ISSUES REMAIN

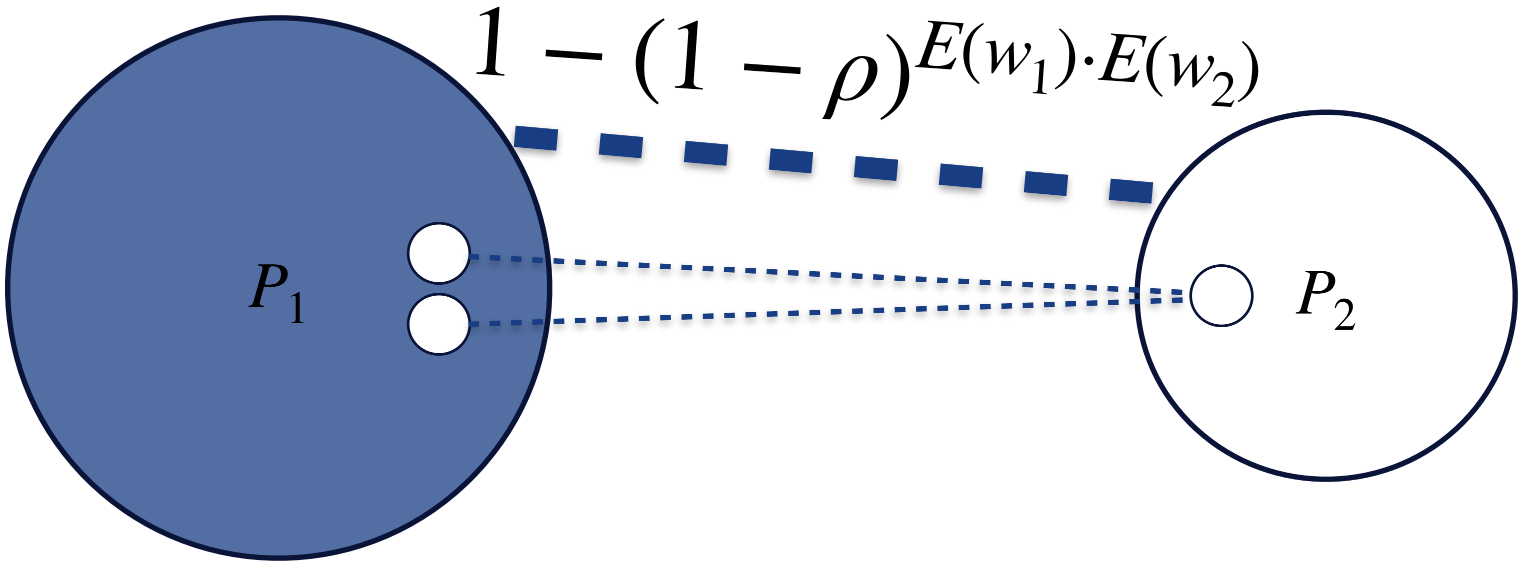
$$E(p) \triangleq \lceil \alpha_p \cdot n \rceil$$



# A FEW ISSUES REMAIN

- Selection of neighbors requires $n$ coinflips.

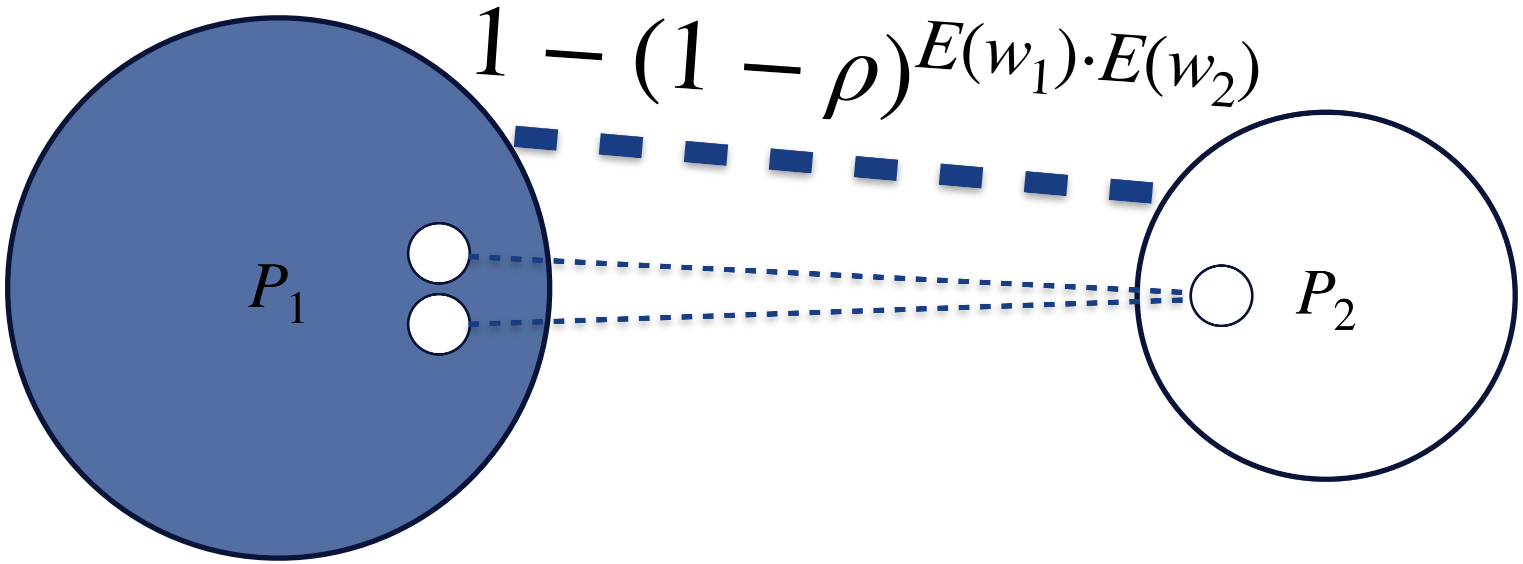
$$E(p) \triangleq \lceil \alpha_p \cdot n \rceil$$



# A FEW ISSUES REMAIN

- Selection of neighbors requires $n$ coinflips.
- Unknown number of neighbors is not very practical.

$$E(p) \triangleq \lceil \alpha_p \cdot n \rceil$$



# WEIGHTED FANOUT FLOODING (WFF)

# WEIGHTED FANOUT FLOODING (WFF)

1. $E(p) \triangleq \lceil \alpha_p \cdot n \rceil$

# WEIGHTED FANOUT FLOODING (WFF)

1. $E(p) \triangleq \lceil \alpha_p \cdot n \rceil$
2. Party $p$ selects $K = k \cdot E(p)$ neighbors.

# WEIGHTED FANOUT FLOODING (WFF)

1. $E(p) \triangleq \lceil \alpha_p \cdot n \rceil$
2. Party $p$ selects $K = k \cdot E(p)$ neighbors.

Parameter of protocol.

# WEIGHTED FANOUT FLOODING (WFF)

1. $E(p) \triangleq \lceil \alpha_p \cdot n \rceil$

Parameter of protocol.

2. Party $p$ selects $K = k \cdot E(p)$ neighbors.

3. Neighbors are selected by weighted sampling without replacement where each party $q$ is weighted by $E(q)$.

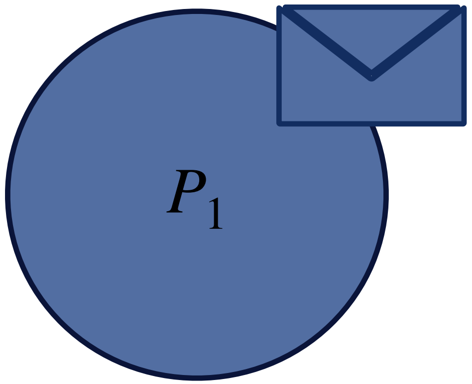
# WEIGHTED FANOUT FLOODING (WFF)

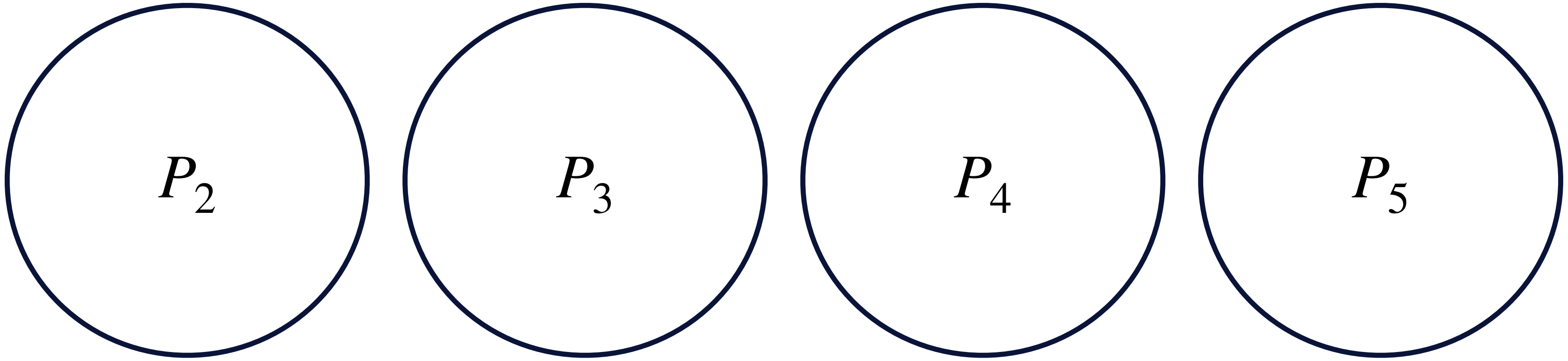
1. $E(p) \triangleq \lceil \alpha_p \cdot n \rceil$

Parameter of protocol.

2. Party $p$ selects $K = k \cdot E(p)$ neighbors.

3. Neighbors are selected by weighted sampling without replacement where each party $q$ is weighted by $E(q)$.

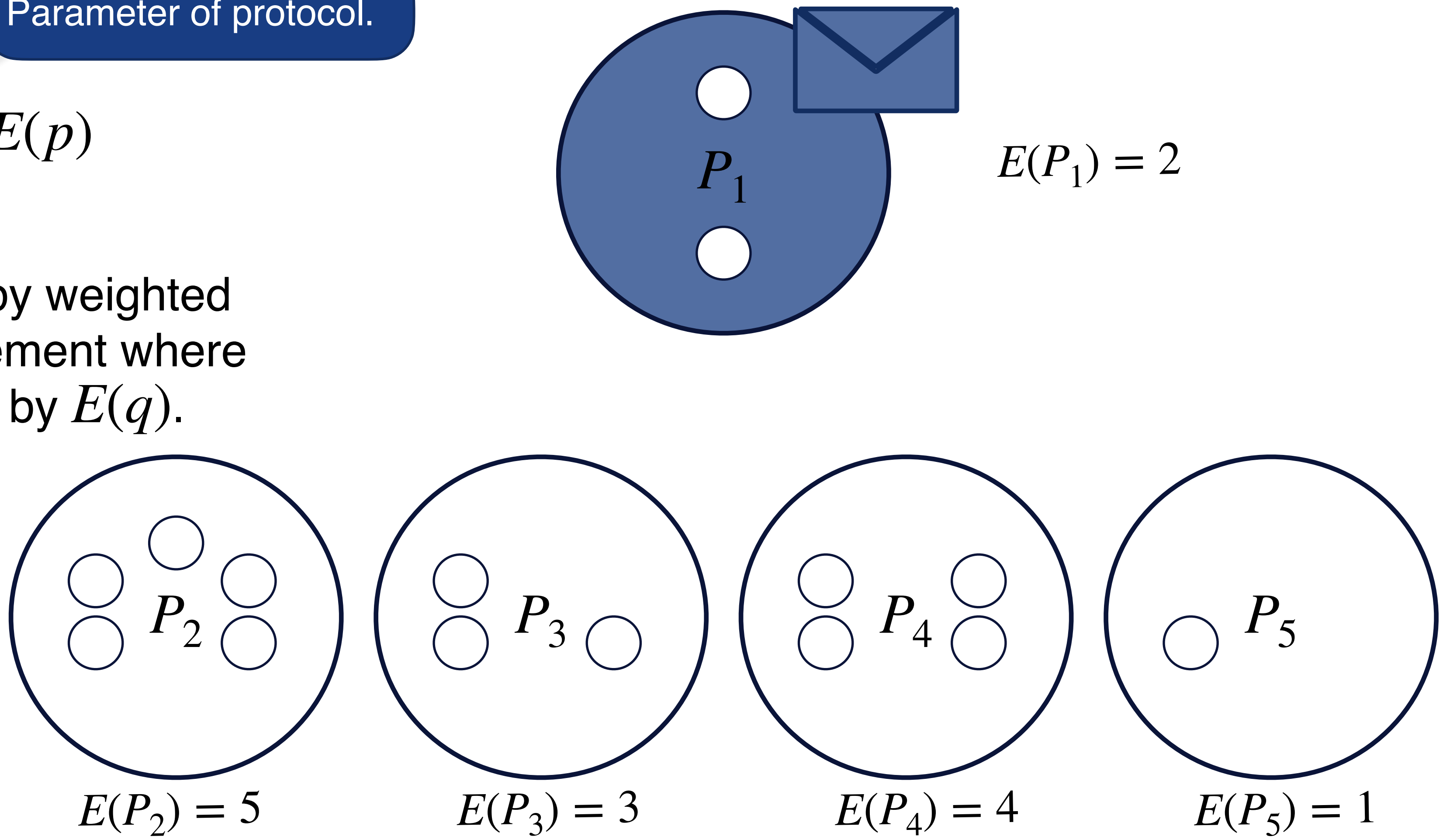


# WEIGHTED FANOUT FLOODING (WFF)

1. $E(p) \triangleq \lceil \alpha_p \cdot n \rceil$
2. Party $p$ selects $K = k \cdot E(p)$ neighbors.
3. Neighbors are selected by weighted sampling without replacement where each party $q$ is weighted by $E(q)$.

Parameter of protocol.



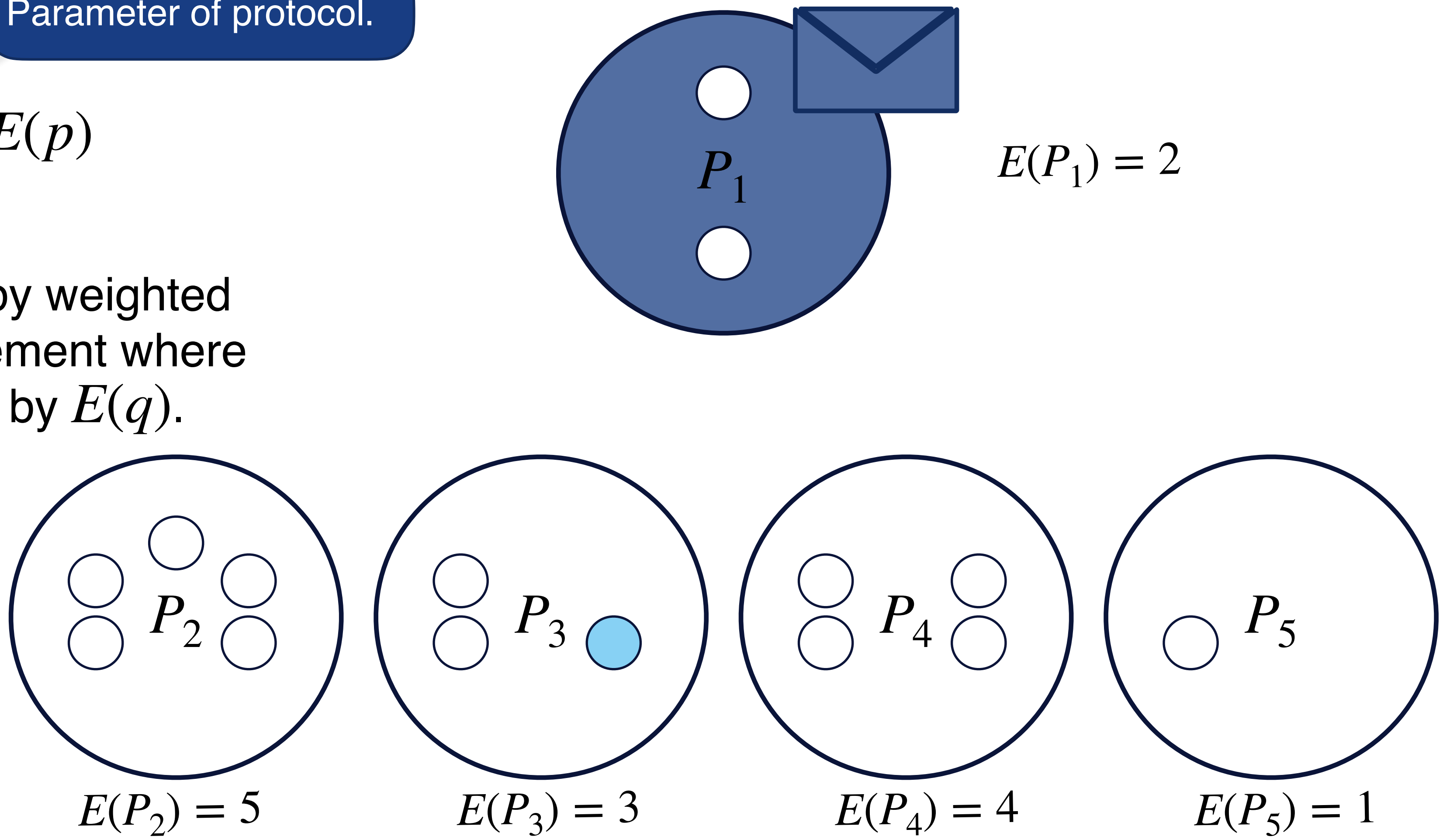
# WEIGHTED FANOUT FLOODING (WFF)

1. $E(p) \triangleq \lceil \alpha_p \cdot n \rceil$

2. Party $p$ selects $K = k \cdot E(p)$ neighbors.

Parameter of protocol.

3. Neighbors are selected by weighted sampling without replacement where each party $q$ is weighted by $E(q)$.



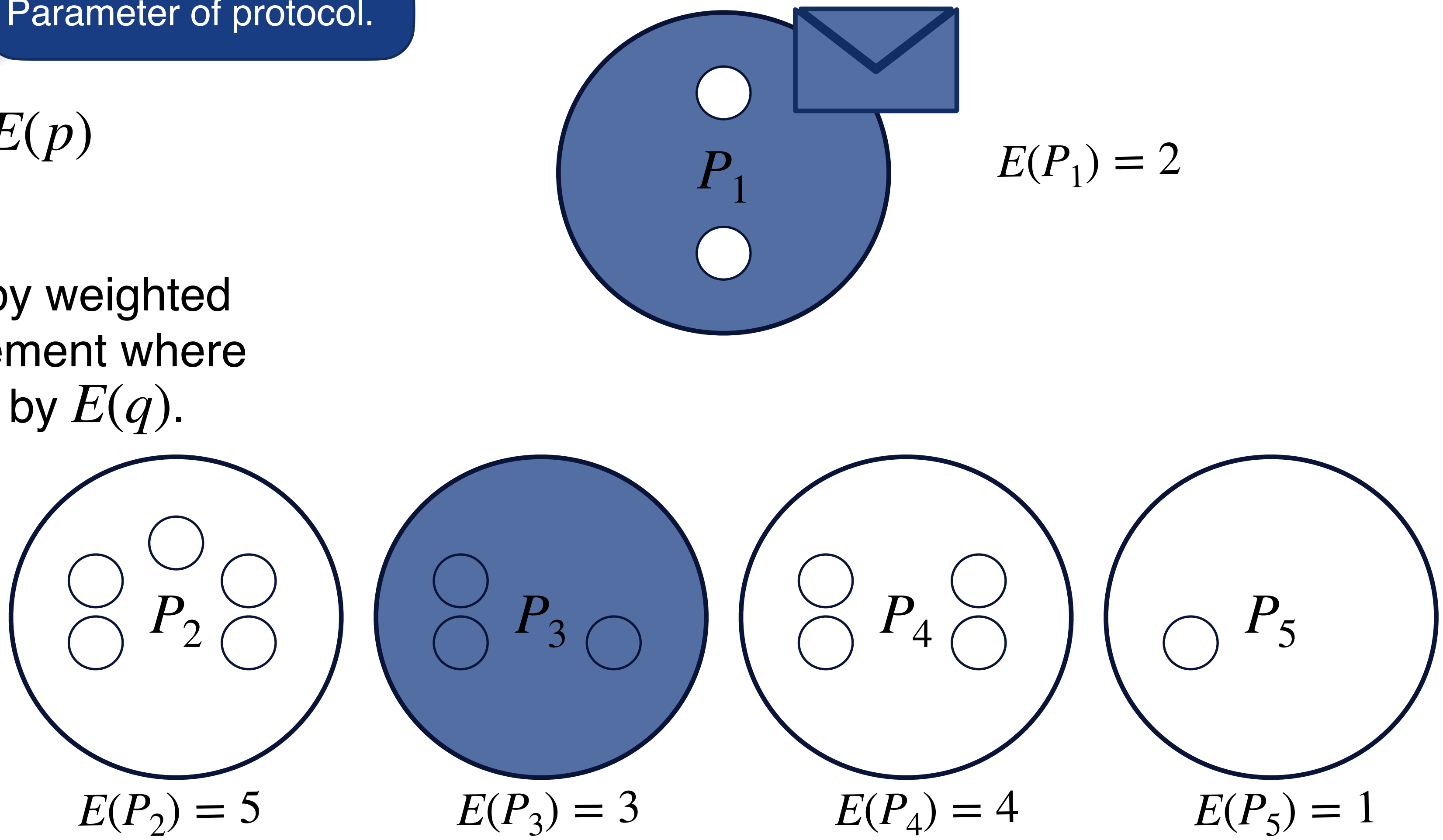
# WEIGHTED FANOUT FLOODING (WFF)

1. $E(p) \triangleq \lceil \alpha_p \cdot n \rceil$

Parameter of protocol.

2. Party $p$ selects $K = k \cdot E(p)$ neighbors.

3. Neighbors are selected by weighted sampling without replacement where each party $q$ is weighted by $E(q)$.



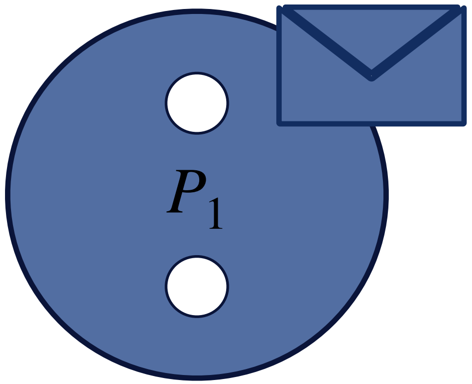
# WEIGHTED FANOUT FLOODING (WFF)

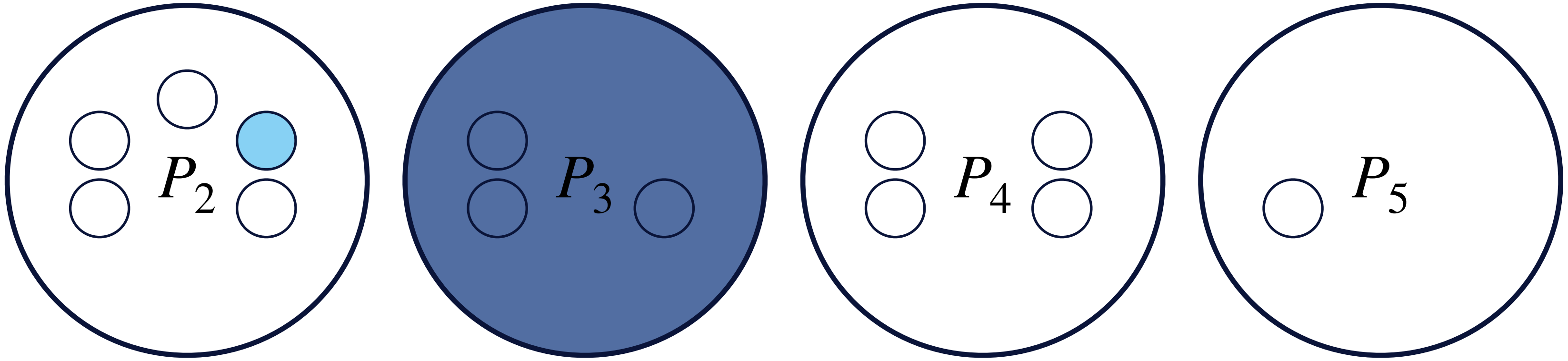
1. $E(p) \triangleq \lceil \alpha_p \cdot n \rceil$

Parameter of protocol.

2. Party $p$ selects $K = k \cdot E(p)$ neighbors.

3. Neighbors are selected by weighted sampling without replacement where each party $q$ is weighted by $E(q)$.



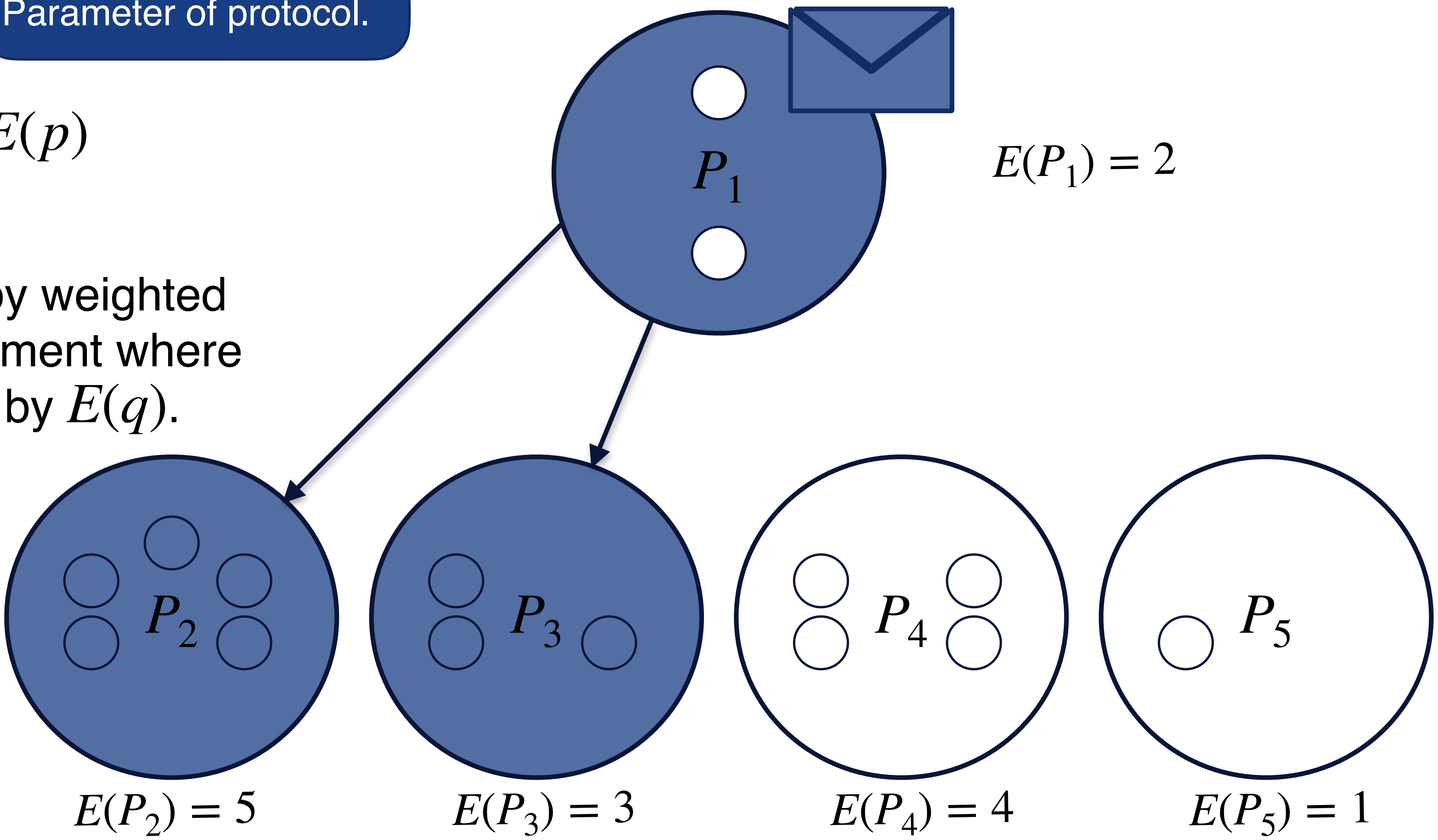
# WEIGHTED FANOUT FLOODING (WFF)

1. $E(p) \triangleq \lceil \alpha_p \cdot n \rceil$

Parameter of protocol.

2. Party $p$ selects $K = k \cdot E(p)$ neighbors.

3. Neighbors are selected by weighted sampling without replacement where each party $q$ is weighted by $E(q)$.



# MAIN RESULT

# MAIN RESULT

**Theorem (informal).**

*For* $k = O\left((\log(n) + \kappa) \cdot \gamma^{-1}\right)$ *and* $\Delta = O(\log(n) \cdot \delta)$ *WFF(k) is a* $\Delta$*-Flood protocol.*

$\kappa$ = security parameter.
$\gamma$ = fraction of honest weight.
$\delta$ = delay on underlying channels.

# MAIN RESULT

**Theorem (informal).**

*For* $k = O\left((\log(n) + \kappa) \cdot \gamma^{-1}\right)$ *and* $\Delta = O(\log(n) \cdot \delta)$ *WFF(k) is a* $\Delta$*-Flood protocol.*

$\kappa$ = security parameter.
$\gamma$ = fraction of honest weight.
$\delta$ = delay on underlying channels.

- Message complexity: $O(k \cdot n)$.

# MAIN RESULT

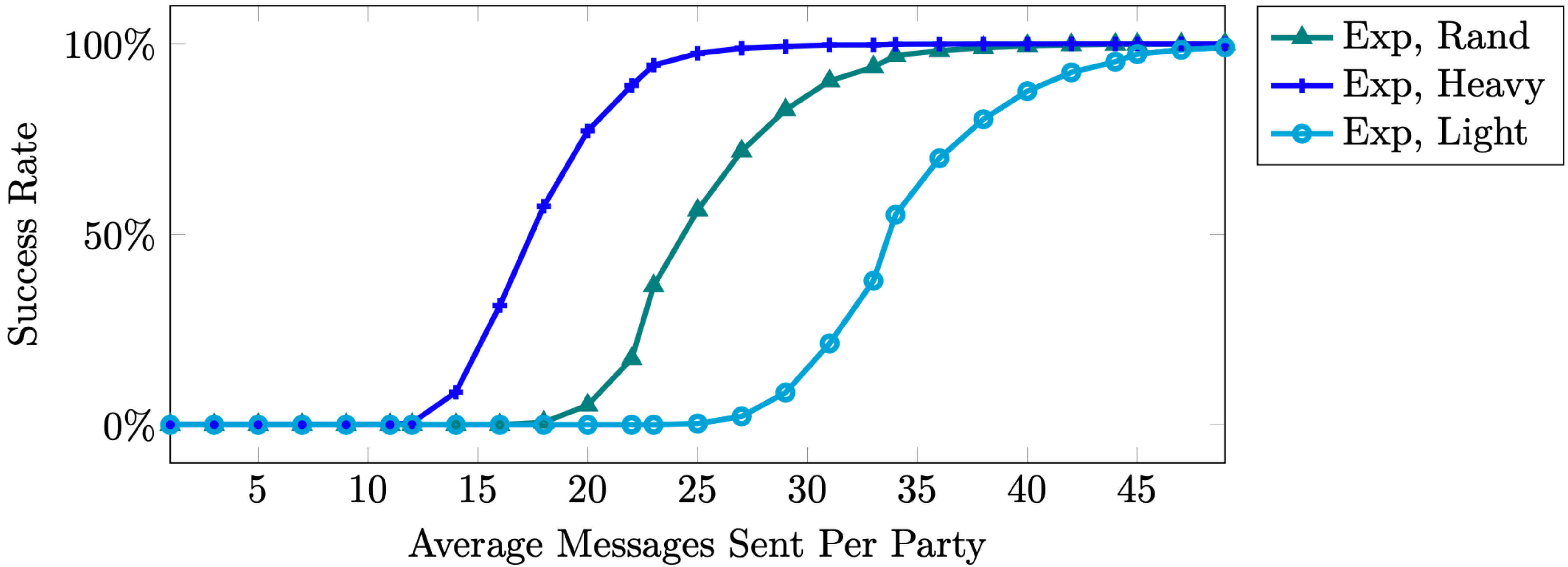
**Theorem (informal).**

*For* $k = O\left((\log(n) + \kappa) \cdot \gamma^{-1}\right)$ *and* $\Delta = O(\log(n) \cdot \delta)$ *WFF(k) is a* $\Delta$*-Flood protocol.*

$\kappa$ = security parameter.
$\gamma$ = fraction of honest weight.
$\delta$ = delay on underlying channels.

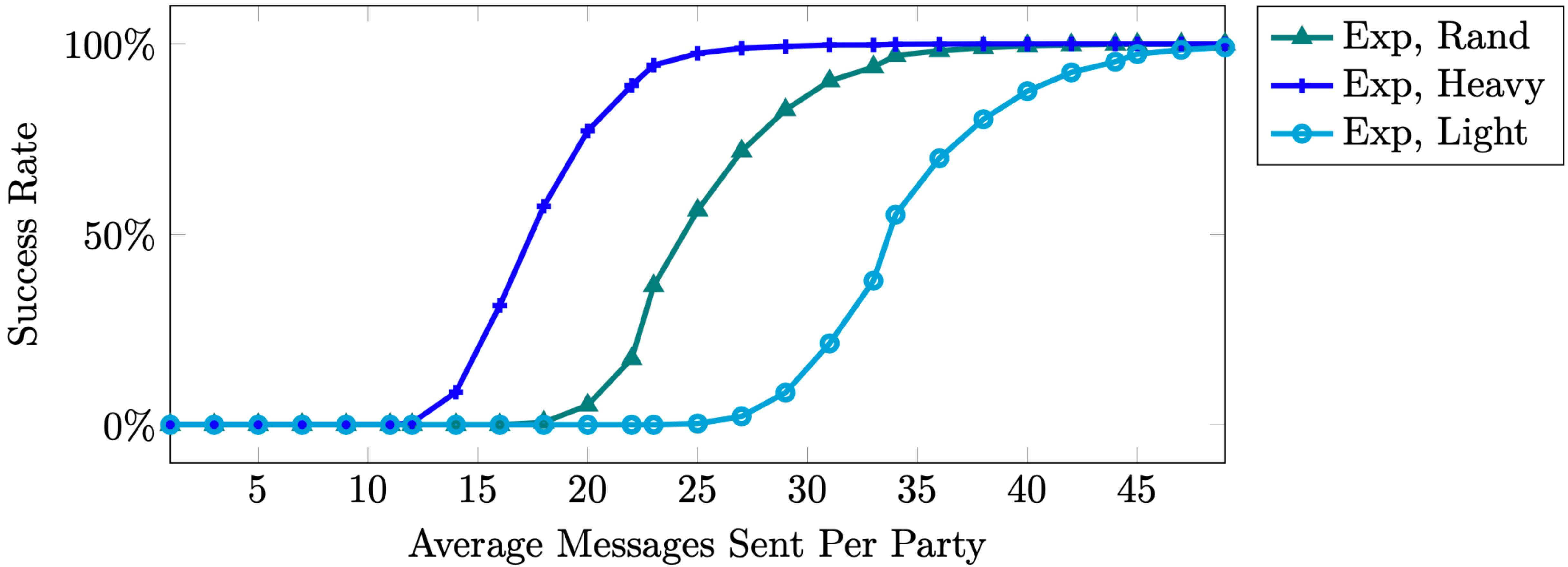
- Message complexity: $O(k \cdot n)$.
- Neighbors of a party $p$: $O\left(k \cdot \lceil \alpha_p \cdot n \rceil\right)$.

# PRACTICALITY OF WFF



# PRACTICALITY OF WFF

Exp = Exponentially distributed weights.
Rand = Random corruptions.
Heavy = Corrupt heavy nodes first.
Light = Corrupt light nodes first.



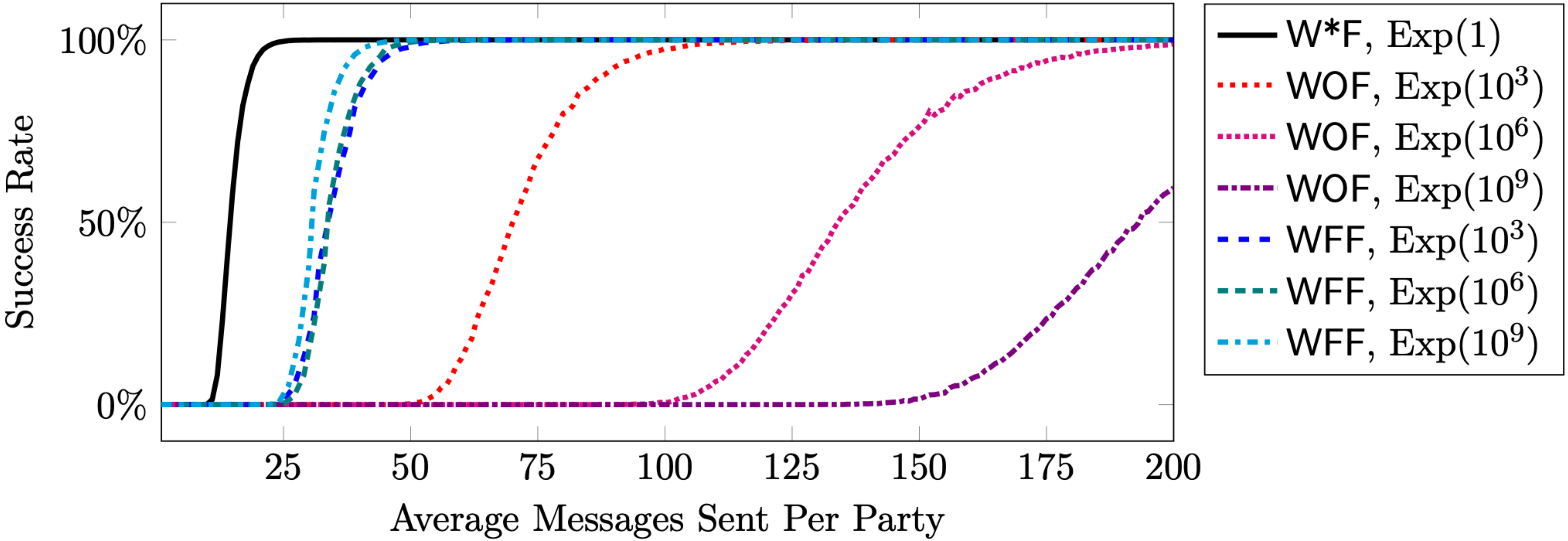
# WFF VS WOF

# WFF VS WOF

“Weight Oblivious Flooding”

# WFF VS WOF

"Weight Oblivious Flooding"



# CONCLUSION

# CONCLUSION

- We present the first provably secure flooding protocol in the weighted setting and demonstrate its practicality using probabilistic simulations.

# CONCLUSION

- We present the first provably secure flooding protocol in the weighted setting and demonstrate its practicality using probabilistic simulations.
- Many more details and additional results: https://eprint.iacr.org/2022/608.

# CONCLUSION

- We present the first provably secure flooding protocol in the weighted setting and demonstrate its practicality using probabilistic simulations.

- Many more details and additional results: https://eprint.iacr.org/2022/608.
  - Necessity of increasing neighborhood for heavy parties.

# CONCLUSION

- We present the first provably secure flooding protocol in the weighted setting and demonstrate its practicality using probabilistic simulations.

- Many more details and additional results: https://eprint.iacr.org/2022/608.
    - Necessity of increasing neighborhood for heavy parties.
    - Necessity of $\log(n)$ neighborhood for fan out flooding.

# CONCLUSION

- We present the first provably secure flooding protocol in the weighted setting and demonstrate its practicality using probabilistic simulations.

- Many more details and additional results: https://eprint.iacr.org/2022/608.
  - Necessity of increasing neighborhood for heavy parties.
  - Necessity of $\log(n)$ neighborhood for fan out flooding.
  - Delivery to parties with zero weight.

# CONCLUSION

- We present the first provably secure flooding protocol in the weighted setting and demonstrate its practicality using probabilistic simulations.
- Many more details and additional results: https://eprint.iacr.org/2022/608.
  - Necessity of increasing neighborhood for heavy parties.
  - Necessity of $\log(n)$ neighborhood for fan out flooding.
  - Delivery to parties with zero weight.
  - Additional simulations.

# CONCLUSION

- We present the first provably secure flooding protocol in the weighted setting and demonstrate its practicality using probabilistic simulations.

- Many more details and additional results: https://eprint.iacr.org/2022/608.
  - Necessity of increasing neighborhood for heavy parties.
  - Necessity of $\log(n)$ neighborhood for fan out flooding.
  - Delivery to parties with zero weight.
  - Additional simulations.

- Contact: sethomsen@cs.au.dk.

# REFERENCES


[GKL15]: Juan A. Garay, Aggelos Kiayias, and Nikos Leonardos. The bitcoin backbone protocol: Analysis and applications. In EUROCRYPT (2), volume 9057 of Lecture Notes in Computer Science, pages 281–310. Springer, 2015.

[PS17]: Rafael Pass and Elaine Shi. Fruitchains: A fair blockchain. In PODC, pages 315–324. ACM, 2017.

[DGKR18]: Bernardo David, Peter Gazi, Aggelos Kiayias, and Alexander Russell. Ouroboros praos: An adaptively-secure, semi-synchronous proof-of-stake blockchain. In EUROCRYPT (2), volume 10821 of Lecture Notes in Computer Science, pages 66–98. Springer, 2018.

[CM19]: Jing Chen and Silvio Micali. Algorand: A secure and efficient distributed ledger. Theor. Comput. Sci., 777:155–183, 2019.

[DMM+20]: Thomas Dinsdale-Young, Bernardo Magri, Christian Matt, Jesper Buus Nielsen, and Daniel Tschudi. Afgjort: A partially synchronous finality layer for blockchains. In SCN, volume 12238 of Lecture Notes in Computer Science, pages 24–44. Springer, 2020.

[MNT22]: Christian Matt, Jesper Buus Nielsen, and Søren Eller Thomsen. Formalizing delayed adaptive corruptions and the security of flooding networks. In Advances in Cryptology – CRYPTO 2022. Springer, 2022.

# SCALABILITY OF WFF