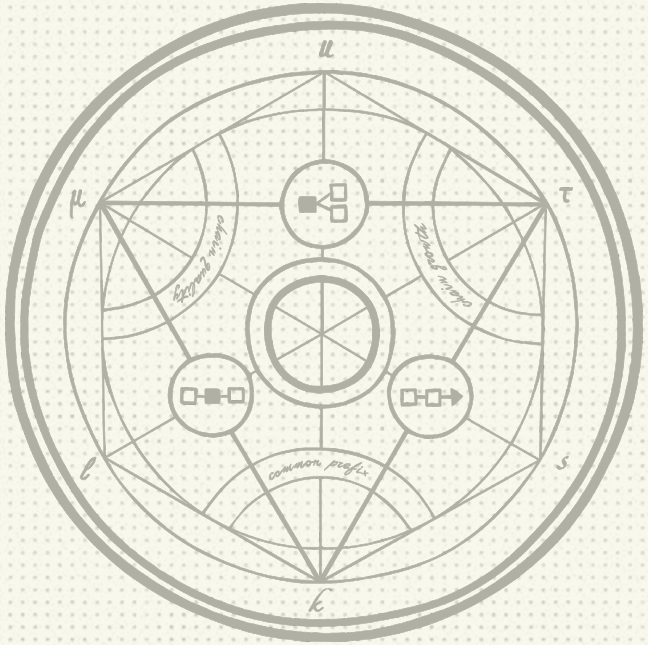
20. Aug 2025

CRYPTO 2025, Santa Barbara



Leader Election with Poly-logarithmic Communication Per Party

Sravya Yandamuri

joint work with:

Amey Bhangale (UC Riverside)

Julian Loss (CISPA)

Chen-Da Liu-Zhang (Lucerne University of Applied Sciences & Arts, Web 3 Foundation)

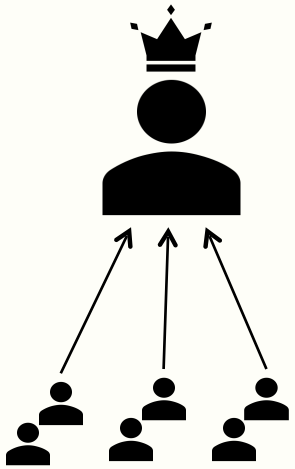
Kartik Nayak (Duke University)

Common ␣ Prefix

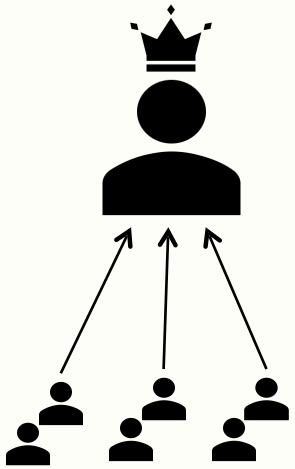
Funded by the European Union

Leader Election

Leader Election

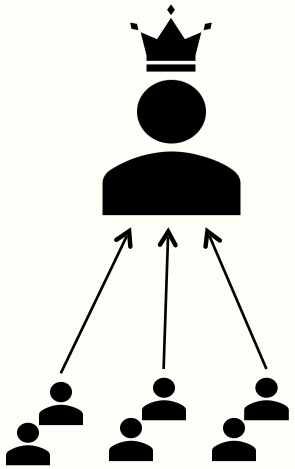


Leader Election



(Agreement) honest parties do not disagree on elected leader

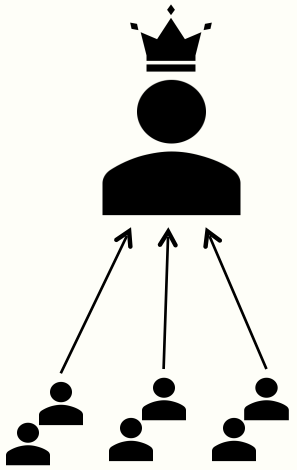
Leader Election



(Agreement) honest parties do not disagree on elected leader

(Validity) with constant probability the elected leader is honest

Leader Election



(Agreement) honest parties do not disagree on elected leader

(Validity) with constant probability the elected leader is honest

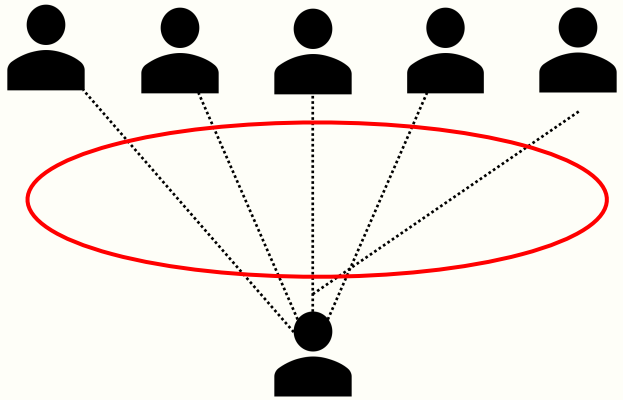
(Termination) all honest parties output a leader

Scalability

DR85: Deterministic consensus requires $O(n^2)$ messages

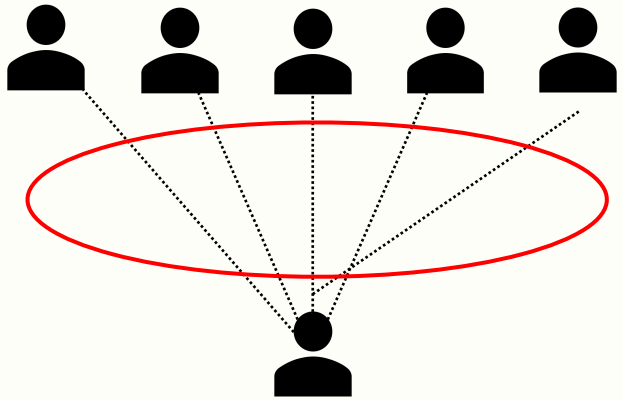
Scalability

DR85: Deterministic consensus requires $O(n^2)$ messages



Scalability

DR85: Deterministic consensus requires $O(n^2)$ messages



Goal: sub-linear communication <u>per party</u>

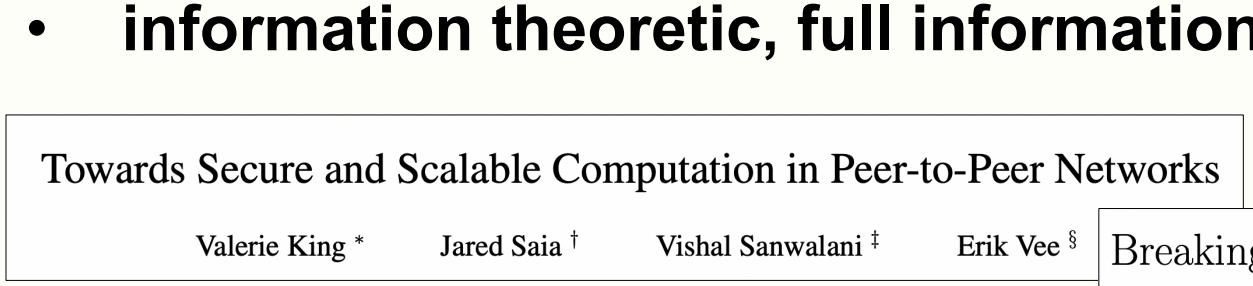
KSSV SODA 2006

- n **parties,** t**<(1/3−**ε**)**n **faulty for** ε**=**O**(1/lnln** n **)**
- $polylog(n)$ **communication complexity per party**
- **all but** o**(1) honest parties know the leader**
- **completes in** $polylog(n)$ **rounds**
- **information theoretic, full information**

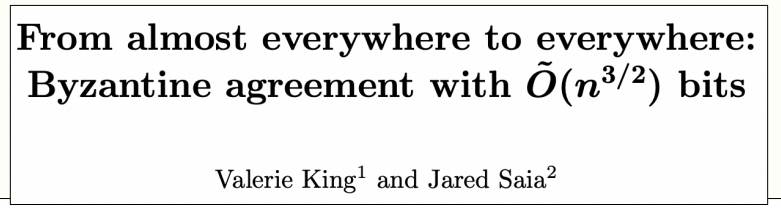
King, Valerie, Jared Saia, Vishal Sanwalani, and Erik Vee. . "Scalable leader election." *SODA*. Vol. 6. 2006.

KSSV SODA 2006

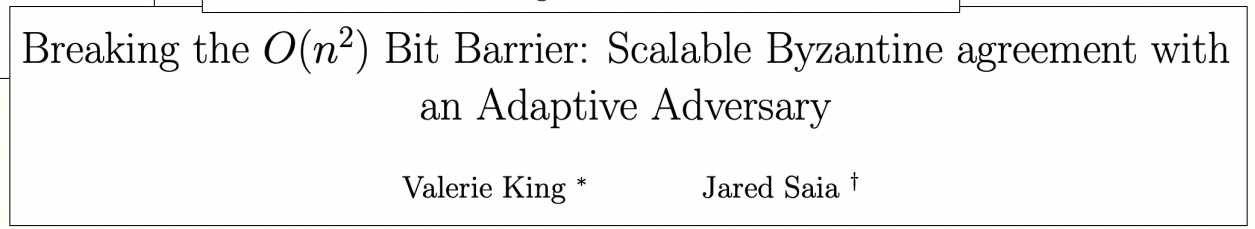
- n **parties,** t**<(1/3−**ε**)**n **faulty for** ε**=**O**(1/lnln** n **)**
- $polylog(n)$ **communication complexity per party**
- **all but** o**(1) honest parties know the leader**
- **completes in** $polylog(n)$ **rounds**
- **information theoretic, full information**


Towards Secure and Scalable Computation in Peer-to-Peer Networks

Valerie King * Jared Saia † Vishal Sanwalani ‡ Erik Vee §


From almost everywhere to everywhere: Byzantine agreement with $\tilde{O}(n^{3/2})$ bits

Valerie King[1] and Jared Saia[2]


Breaking the $O(n^2)$ Bit Barrier: Scalable Byzantine agreement with an Adaptive Adversary

Valerie King * Jared Saia †

King, Valerie, Jared Saia, Vishal Sanwalani, and Erik Vee. . "Scalable leader election." *SODA*. Vol. 6. 2006.

Our Work

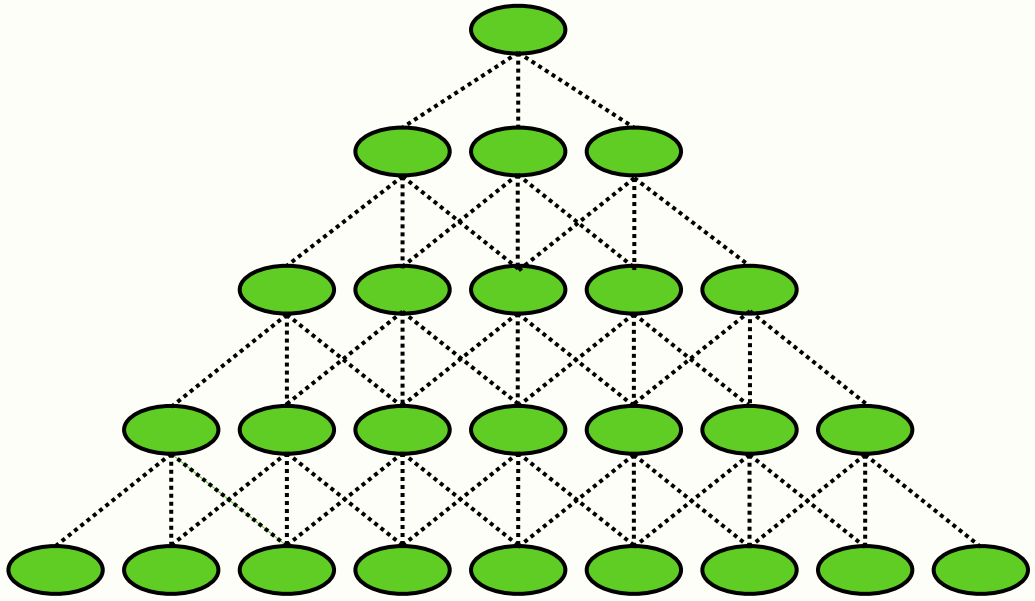
Leader election:

- n **parties,** t**<(1/3−**ε**)**n **faulty for** ε**=**O**(1/lnln** n**)**
- $polylog(n)$ **communication complexity per party**
- **all but** o**(1) honest parties know the leader**
- **completes in** $polylog(n)$ **rounds**
- **information theoretic, full information**

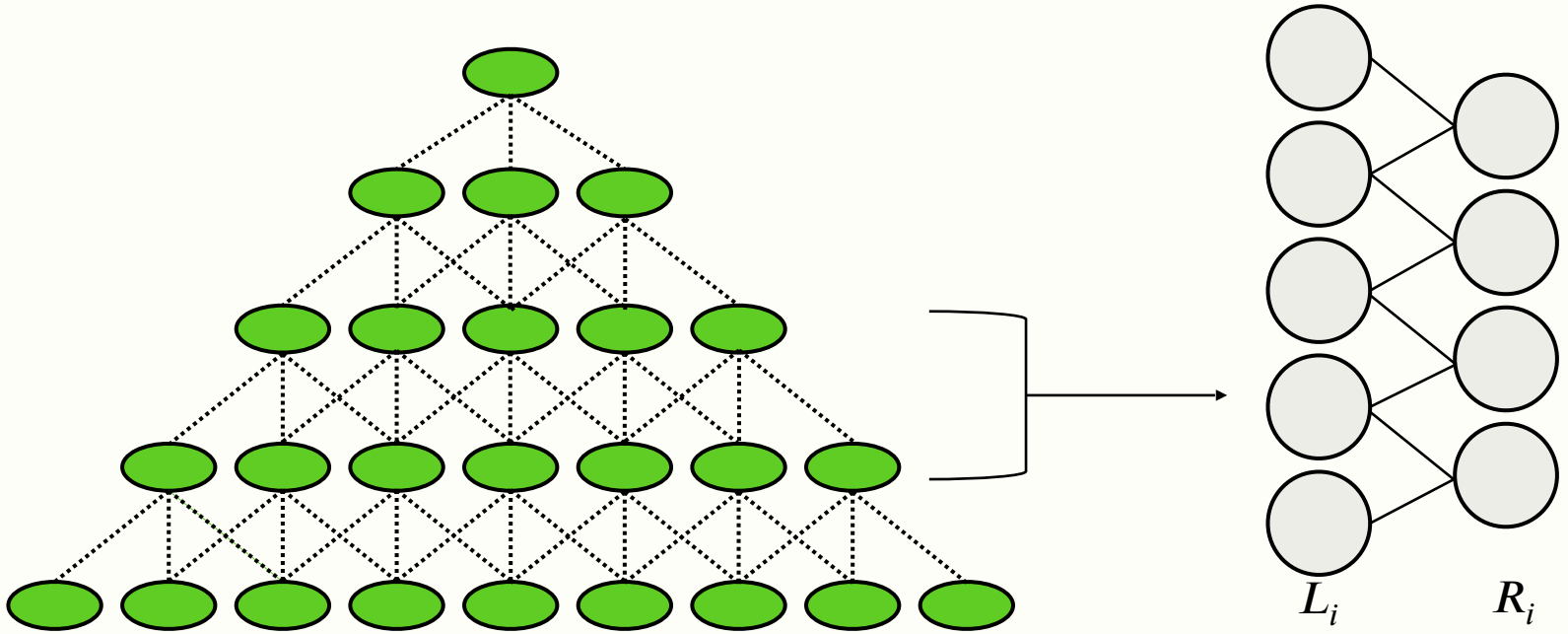
w.h.p. (1 - $n^{-\omega(1)}$*)*

Approach of KSSV

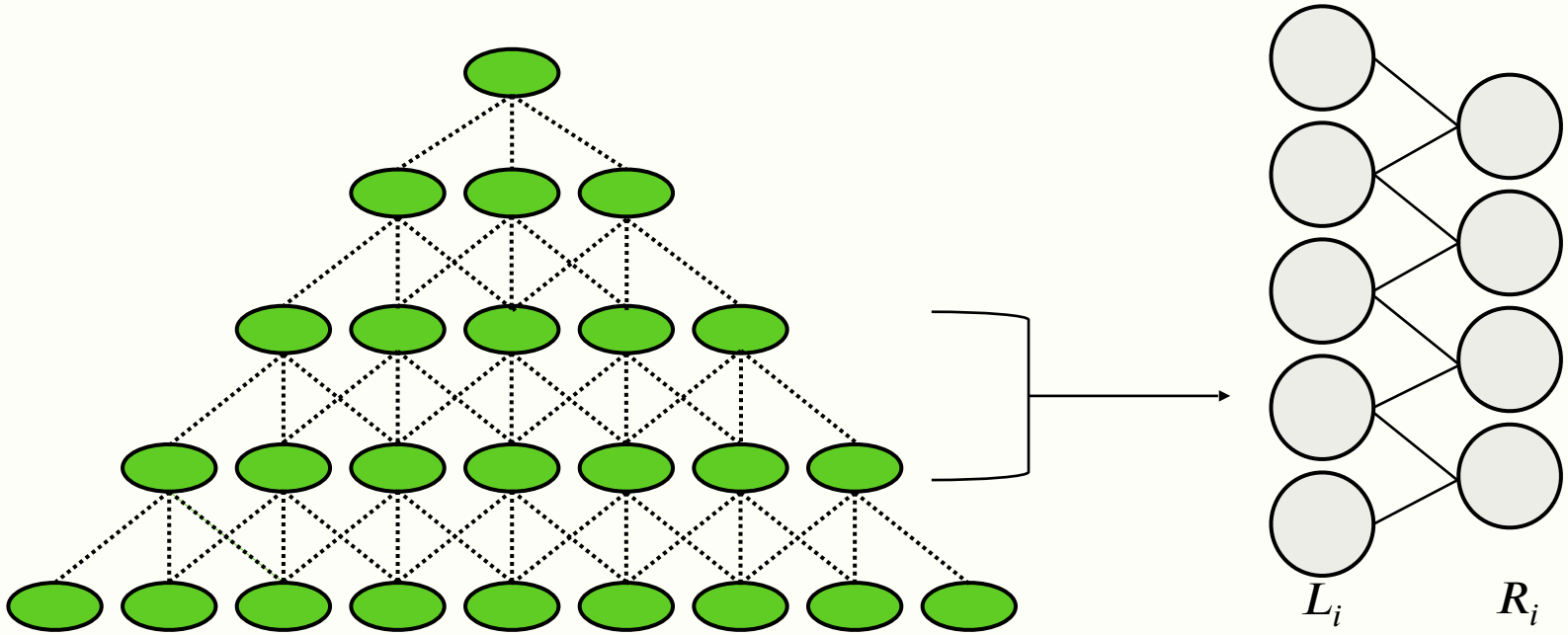
Approach of KSSV



Approach of KSSV



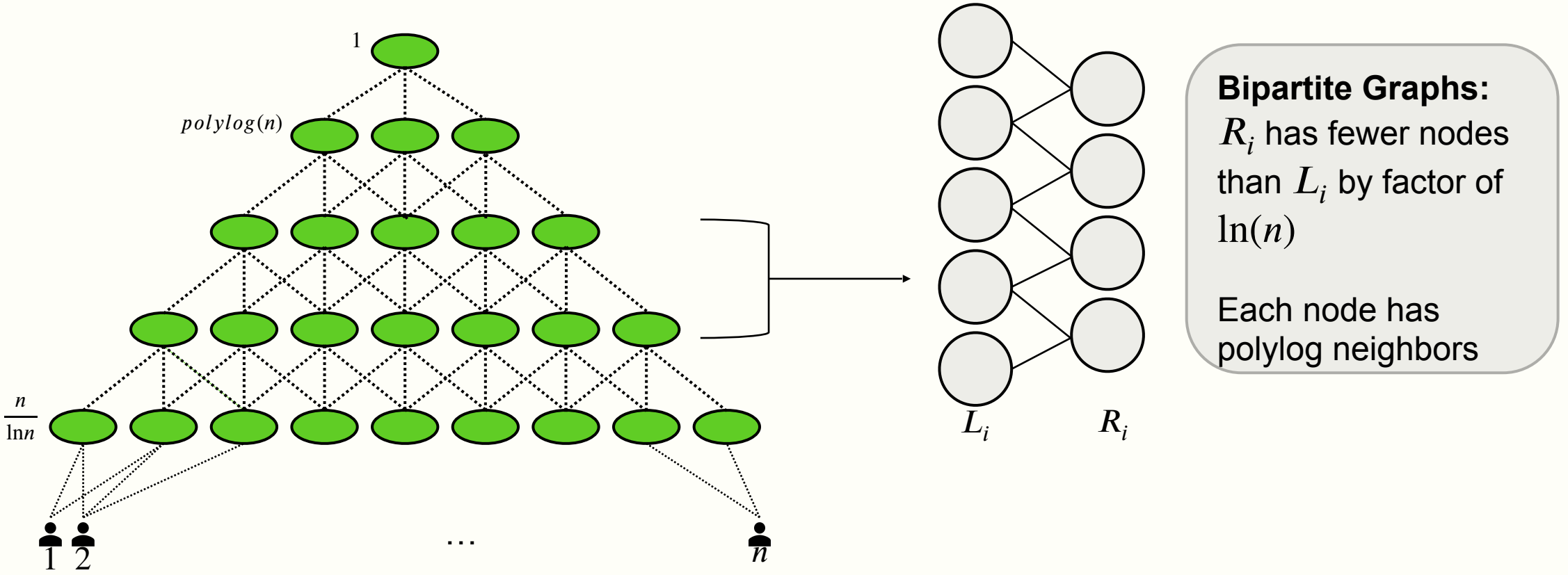
Approach of KSSV



Bipartite Graphs:
R_i has fewer nodes than L_i by factor of $\ln(n)$

Each node has polylog neighbors

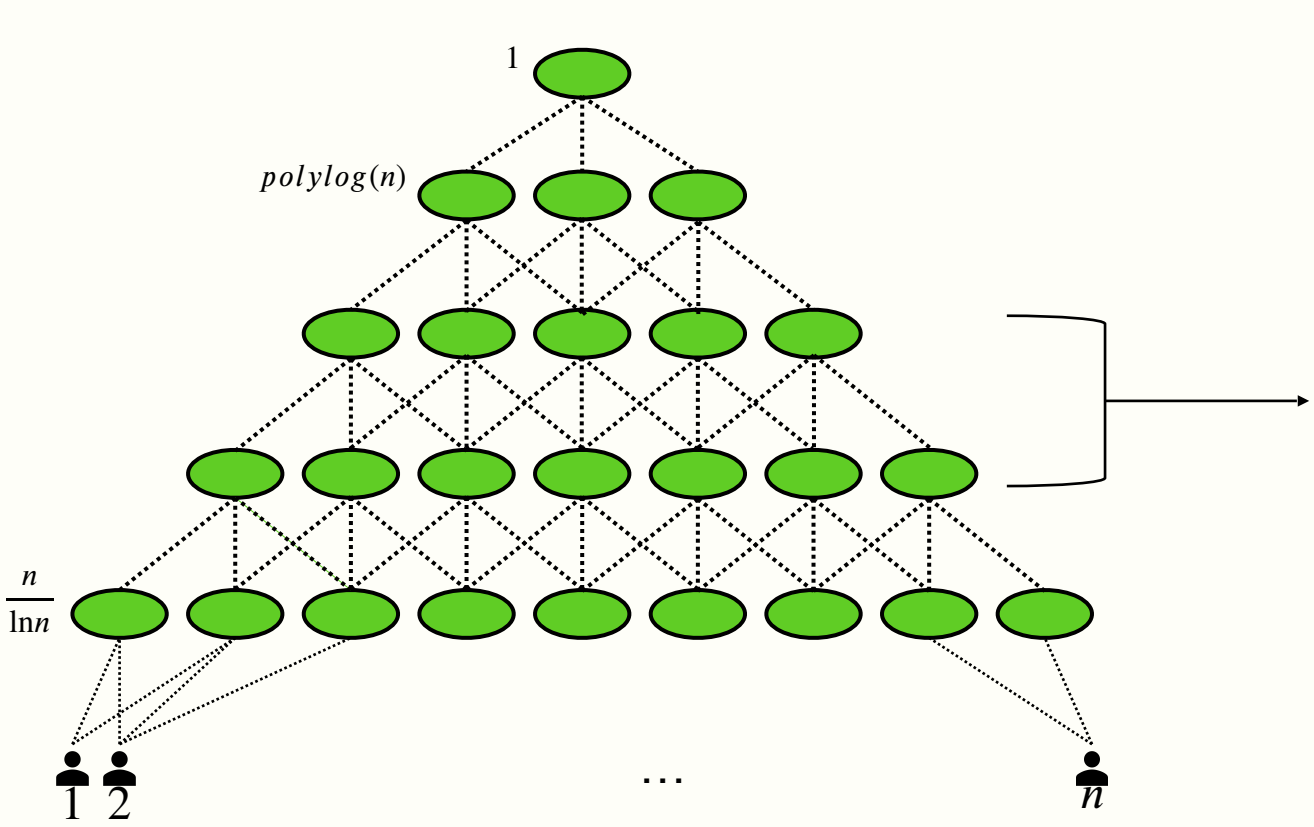
Approach of KSSV

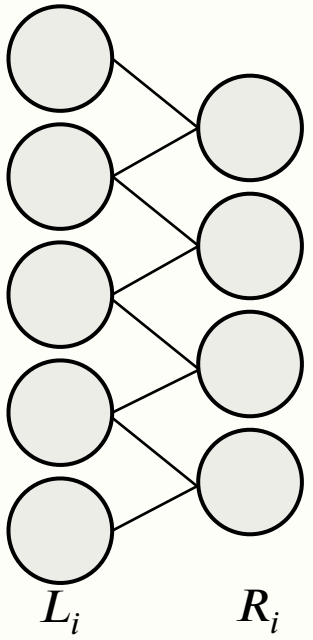


Bipartite Graphs:
R_i has fewer nodes than L_i by factor of $\ln(n)$

Each node has polylog neighbors

Approach of KSSV



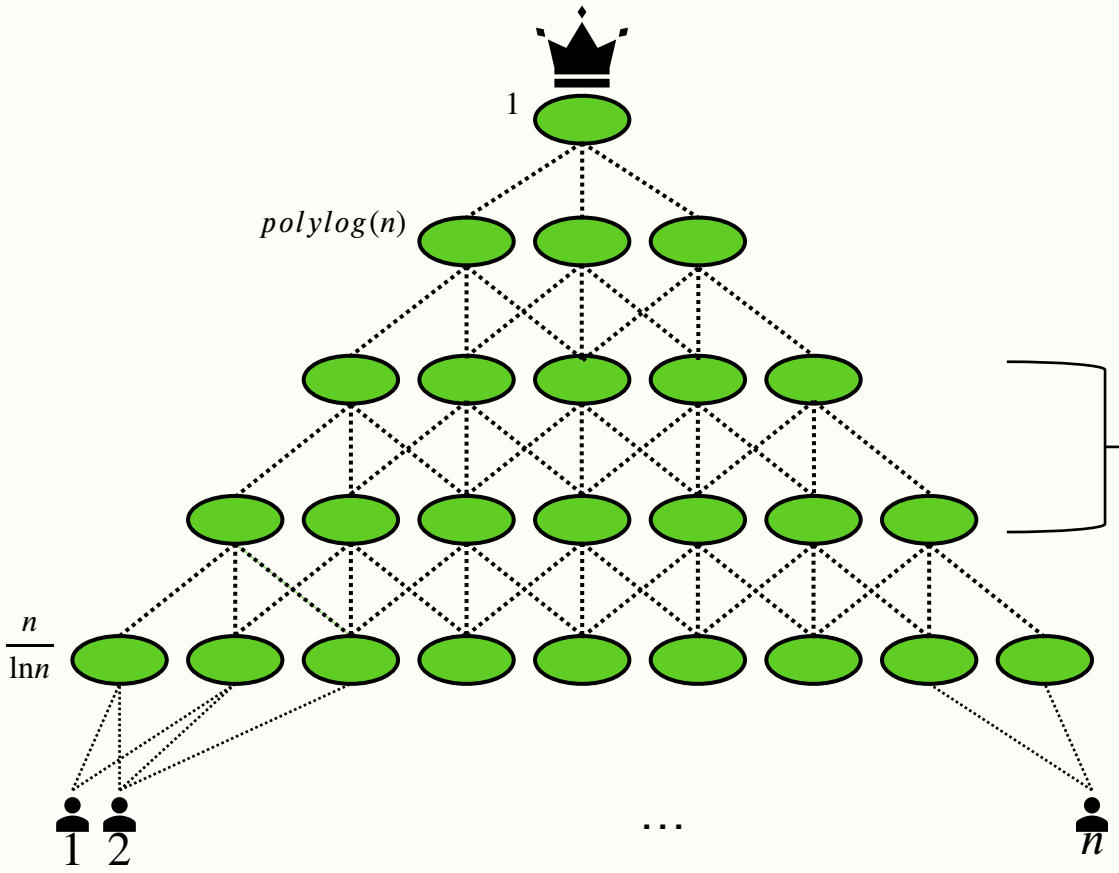


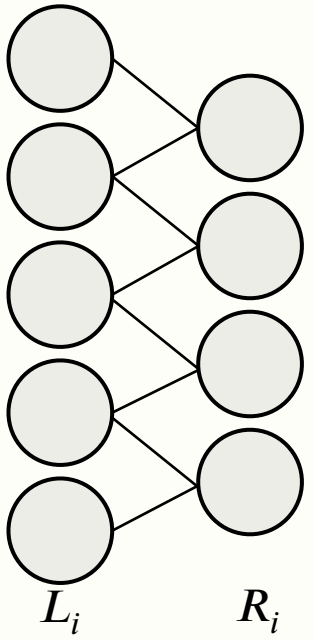
Bipartite Graphs:
R_i has fewer nodes than L_i by factor of $\ln(n)$

Each node has polylog neighbors

polylog parties are elected to nodes up the network

Approach of KSSV





Bipartite Graphs:
R_i has fewer nodes than L_i by factor of $\ln(n)$

Each node has polylog neighbors

polylog parties are elected to nodes up the network

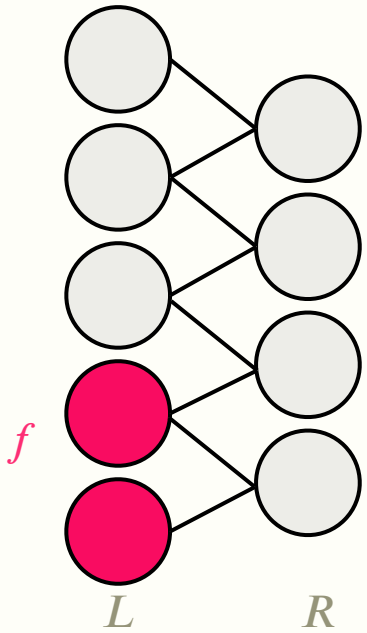
Details

Details

samplers

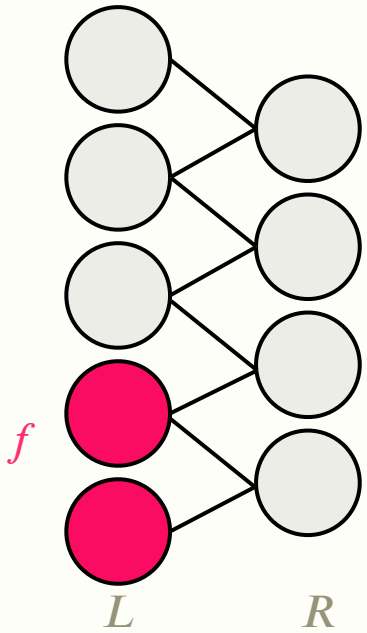
Details

samplers



Details

samplers

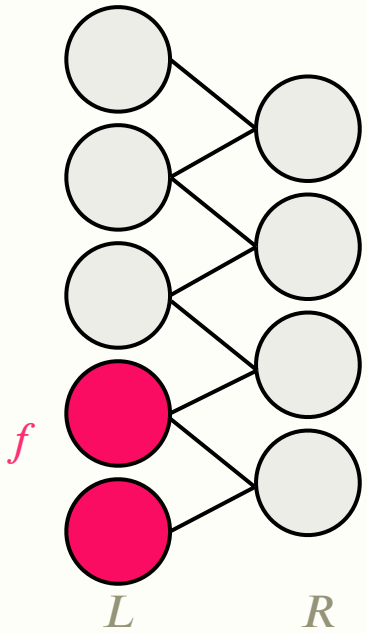


$0 < \delta < 1$

At most δ fraction of the nodes in R have more than $f + \delta$ fraction of pink neighbors

Details

samplers



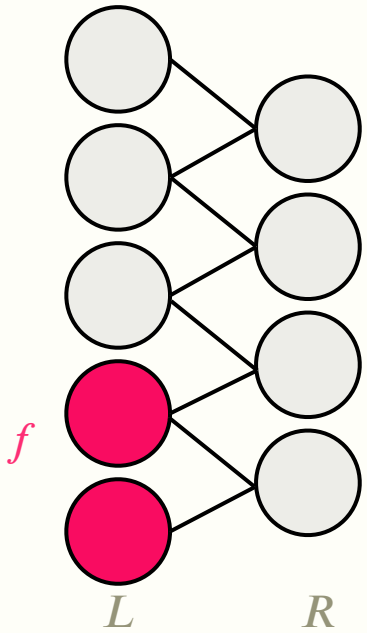
$0 < \delta < 1$

At most δ fraction of the nodes in R have more than $f + \delta$ fraction of pink neighbors

(up to δ nodes on each layer may be “bad”)

Details

samplers



$0 < \delta < 1$

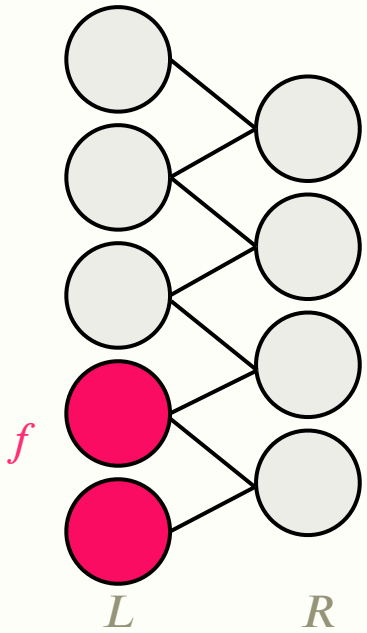
At most δ fraction of the nodes in R have more than $f + \delta$ fraction of pink neighbors

(up to δ nodes on each layer may be “bad”)

communication tree

Details

samplers

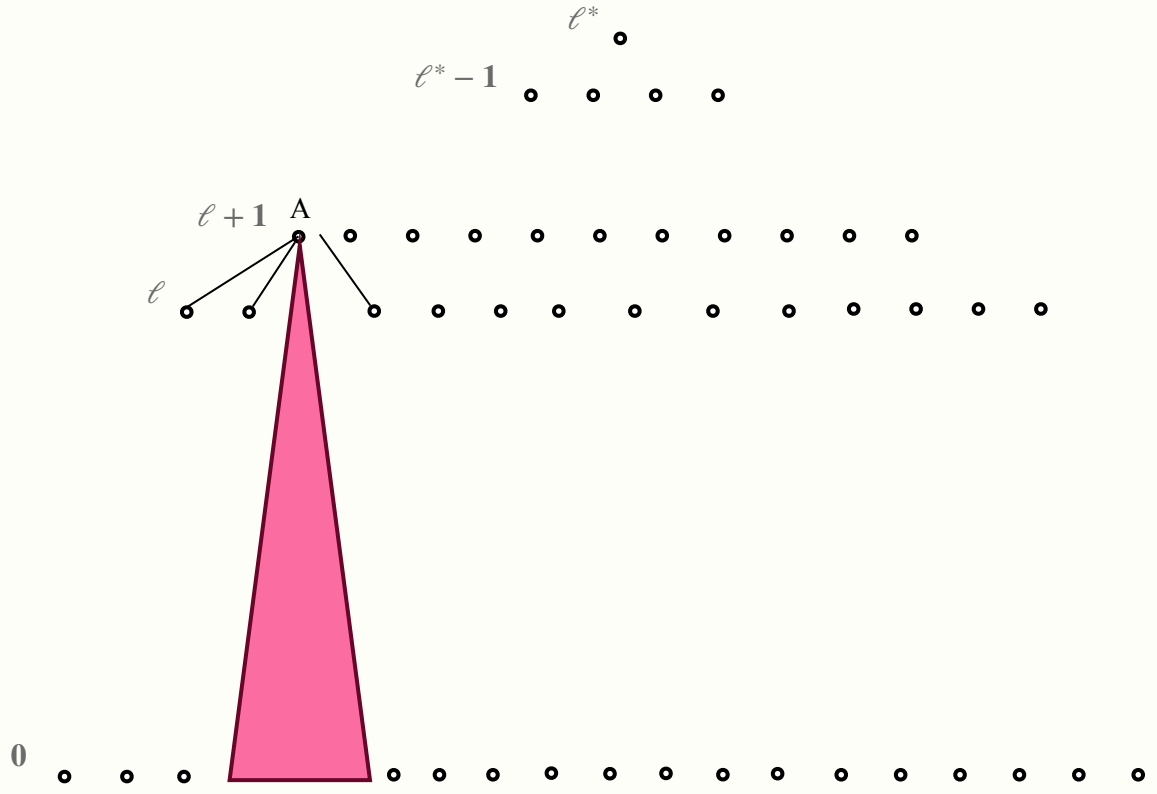


$0 < \delta < 1$

At most δ fraction of the nodes in R have more than $f + \delta$ fraction of pink neighbors

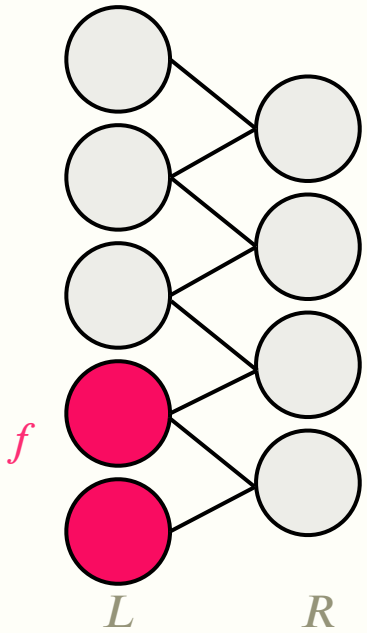
(up to δ nodes on each layer may be “bad”)

communication tree



Details

samplers

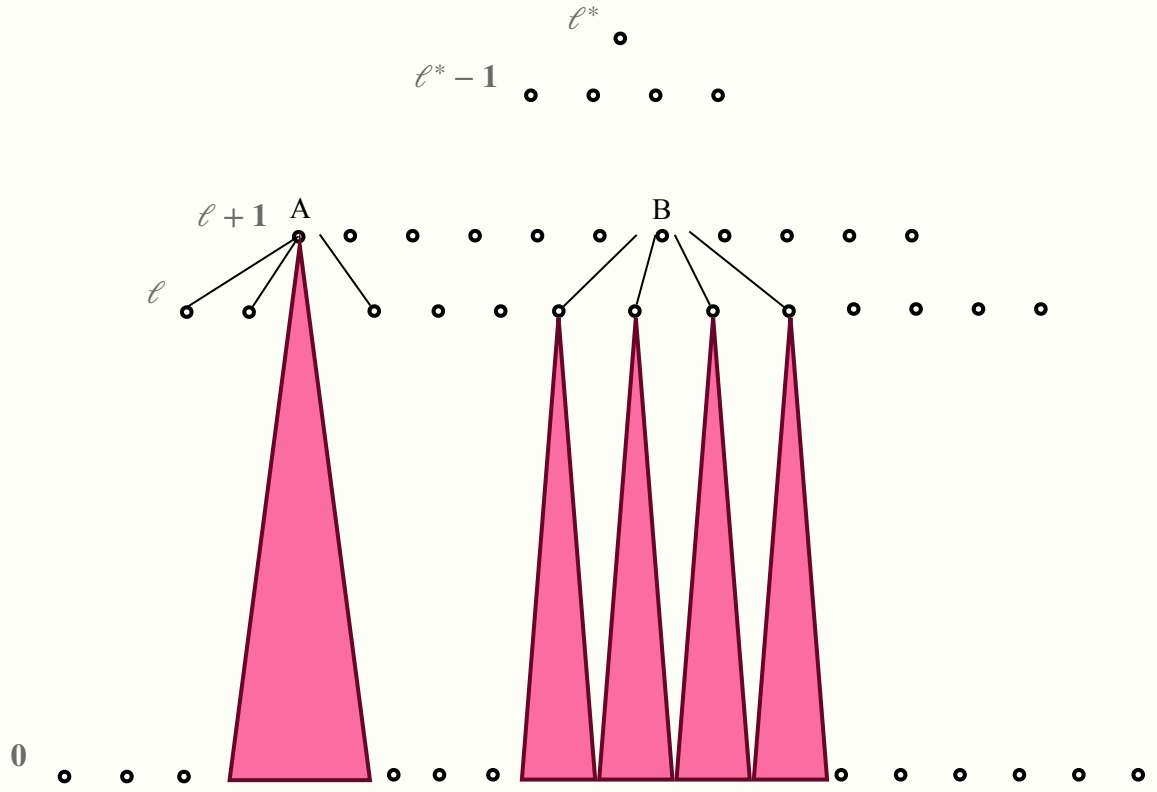


$0 < \delta < 1$

At most δ fraction of the nodes in R have more than $f + \delta$ fraction of pink neighbors

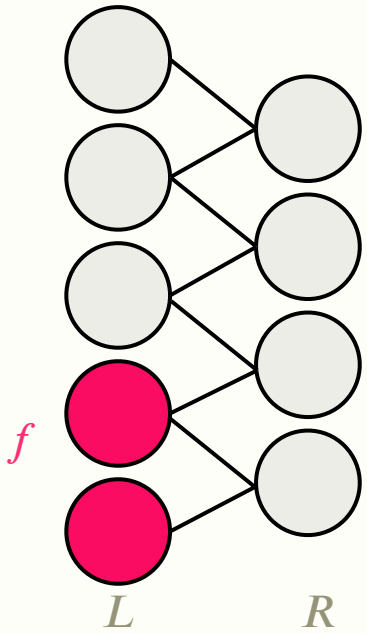
(up to δ nodes on each layer may be “bad”)

communication tree



Details

samplers

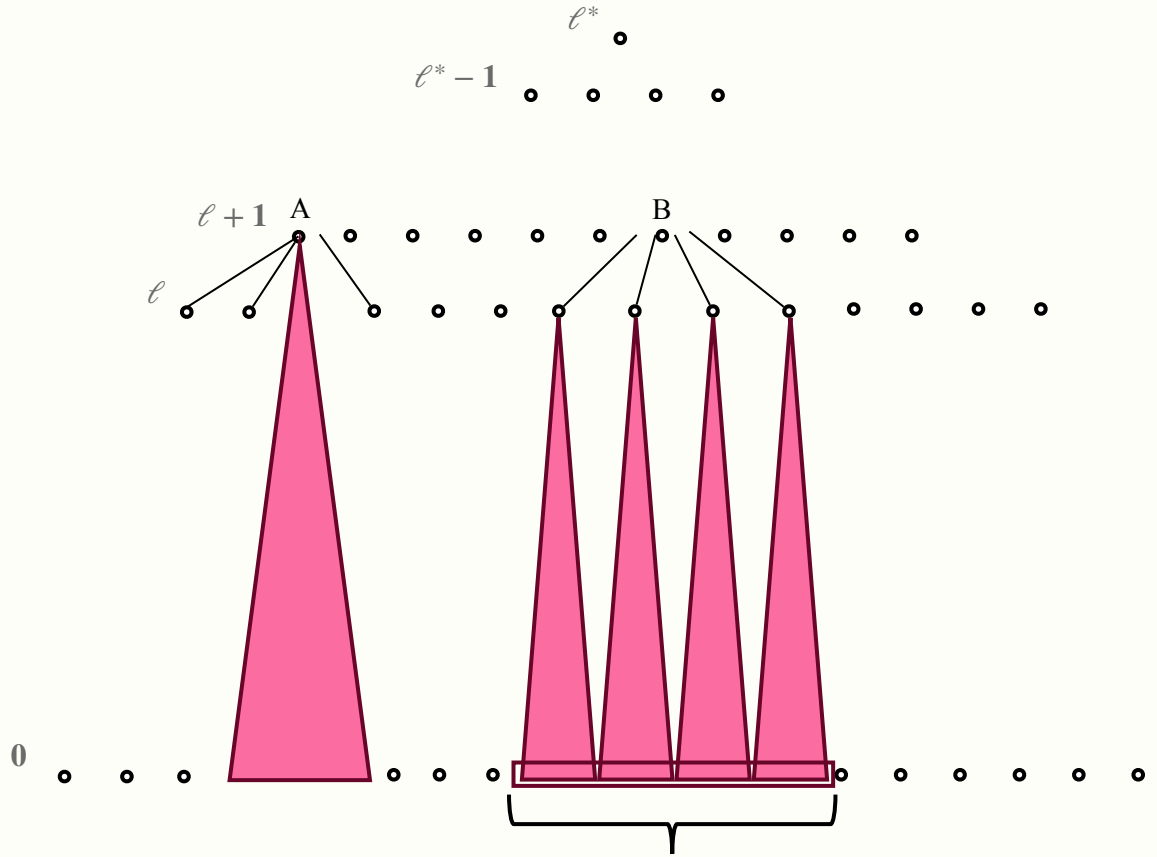


$0 < \delta < 1$

At most δ fraction of the nodes in R have more than $f + \delta$ fraction of pink neighbors

(up to δ nodes on each layer may be “bad”)

communication tree



High-Level Protocol

for layer $\ell = 1$ to ℓ^*:

for each node A on layer ℓ:

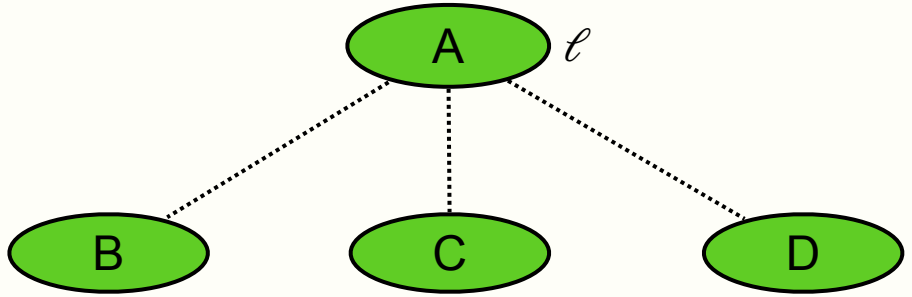
High-Level Protocol

for layer $\ell = 1$ to ℓ^*:

for each node A on layer ℓ:

1. **Parties in nodes in $\Gamma_{-1}(A)$ learn which parties are in other nodes in $\Gamma_{-1}(A)$ (via monitoring sets)**

neighbors of A on layer $\ell - 1$



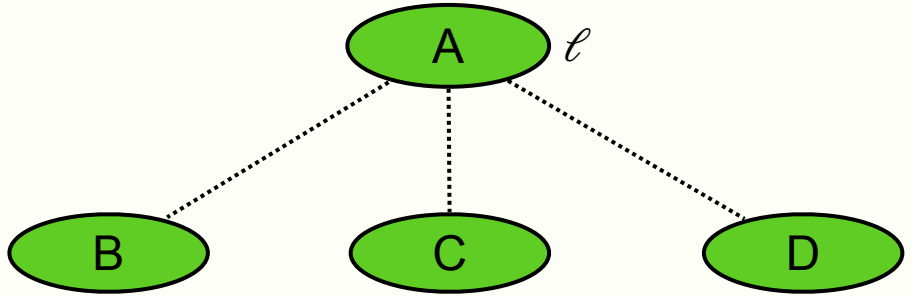
High-Level Protocol

for layer $\ell = 1$ to ℓ^*:

for each node A on layer ℓ:

neighbors of A on layer $\ell - 1$

1. **Parties in nodes in $\Gamma_{-1}(A)$ learn which parties are in other nodes in $\Gamma_{-1}(A)$ (via monitoring sets)**
2. **Parties in nodes in $\Gamma_{-1}(A)$ run election protocol to elect polylog parties to A uniformly at random**



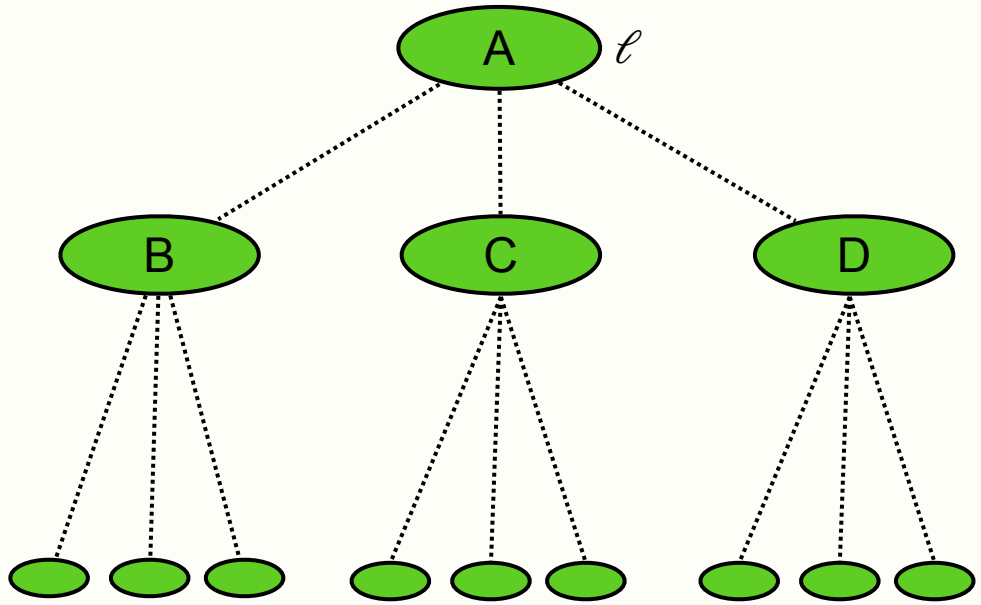
High-Level Protocol

for layer $\ell = 1$ to ℓ^*:

for each node A on layer ℓ:

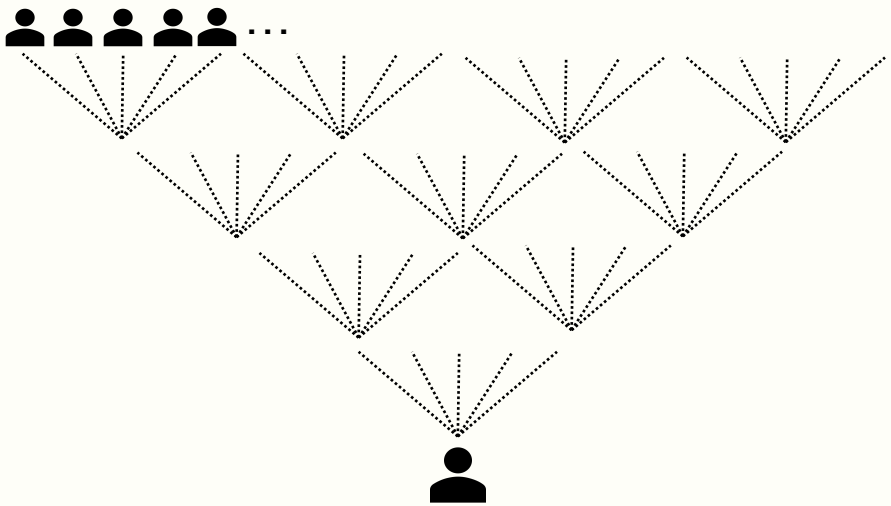
neighbors of A on layer $\ell - 1$

1. **Parties in nodes in $\Gamma_{-1}(A)$ learn which parties are in other nodes in $\Gamma_{-1}(A)$ (via monitoring sets)**
2. **Parties in nodes in $\Gamma_{-1}(A)$ run election protocol to elect polylog parties to A uniformly at random**
3. **Winners of election to A passed down to A's monitoring set via communication tree**

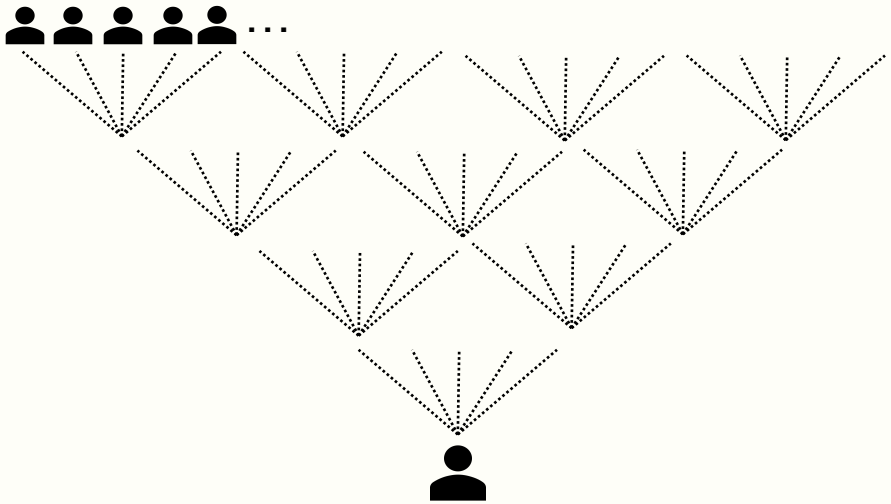


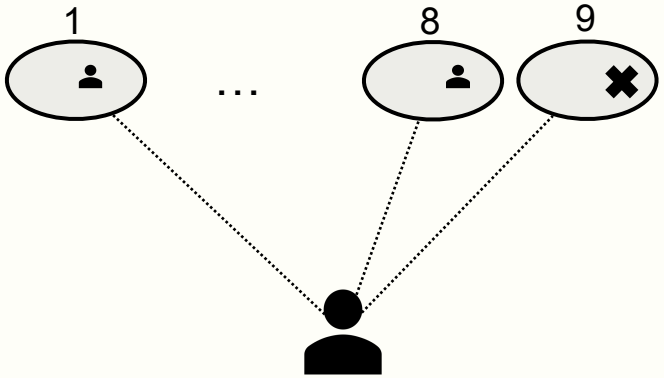
Load Balancing

Load Balancing



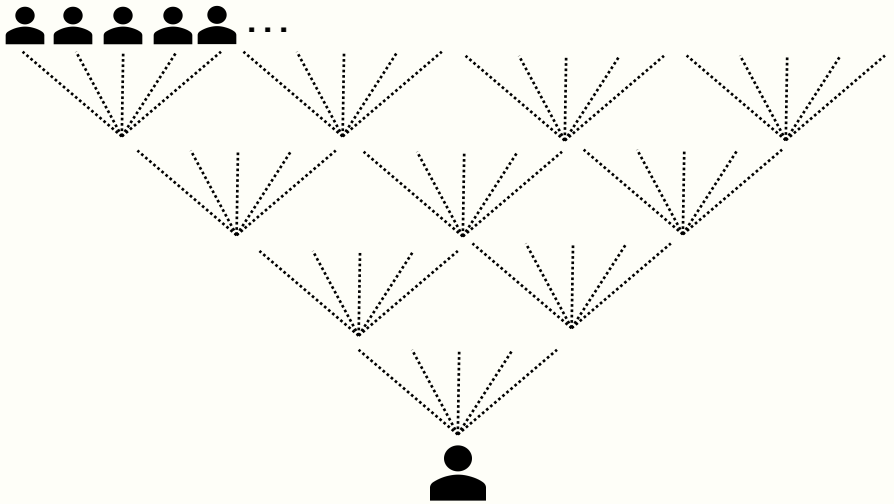
Load Balancing

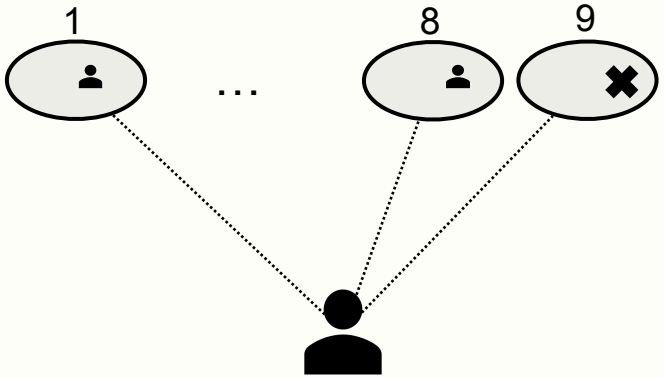




Load Balancing

- **parties go silent after winning 8 elections on a layer**





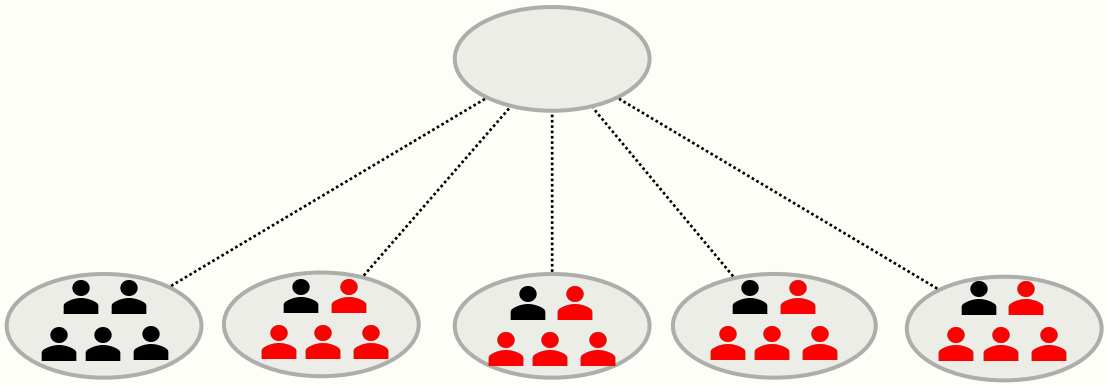
Bad Elections

Bad Elections

Recall: up to δ bad nodes per layer

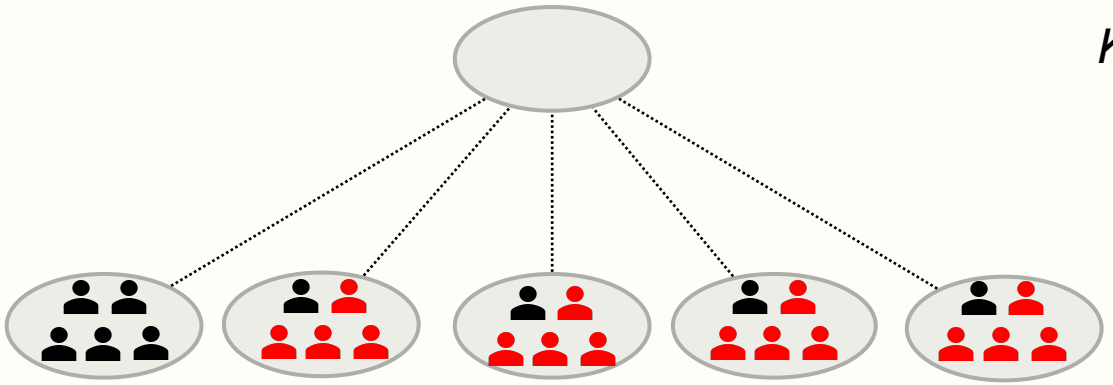
Bad Elections

Recall: up to δ bad nodes per layer



Bad Elections

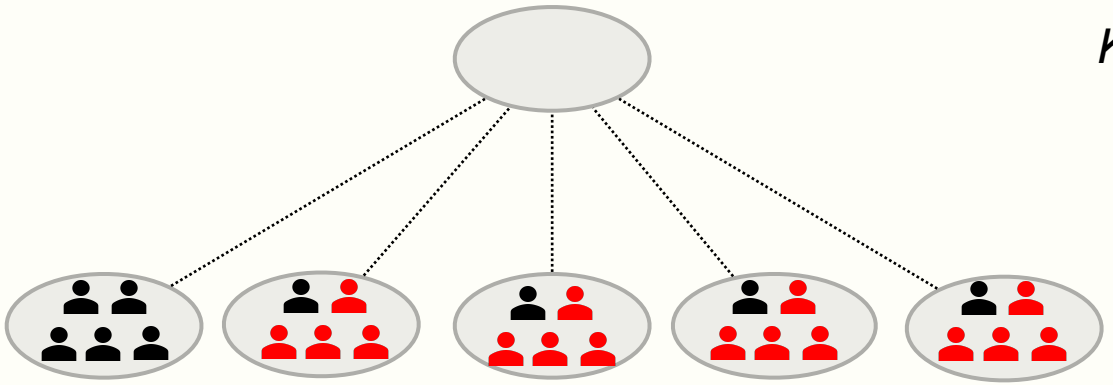
Recall: up to δ bad nodes per layer



KSSV: adversary can choose which polylog(n) parties win

Bad Elections

Recall: up to δ bad nodes per layer



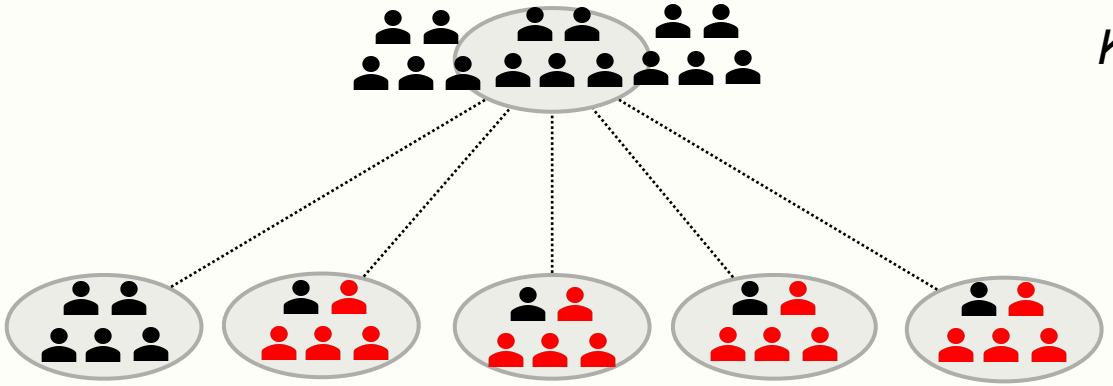
KSSV: adversary can choose which polylog(n) parties win

if $> \frac{1}{3}$ ***fraction of parties are Byzantine,***

no guarantees on election outcome

Bad Elections

Recall: up to δ bad nodes per layer



KSSV: adversary can choose which polylog(n) parties win

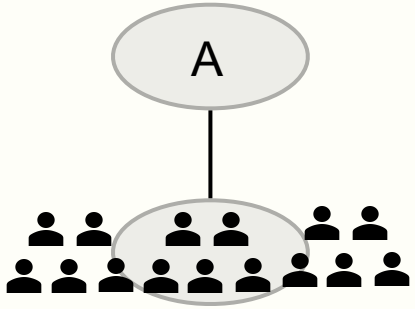
Worst case: every participating party thinks they are elected

if $> \frac{1}{3}$ ***fraction of parties are Byzantine,***
no guarantees on election outcome

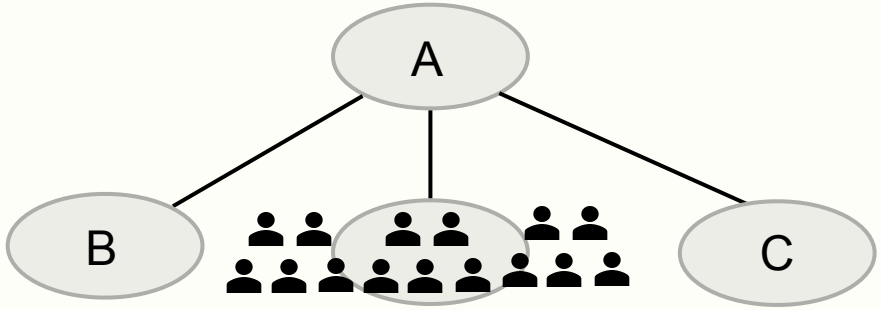
Shadow Elections

Shadow Elections

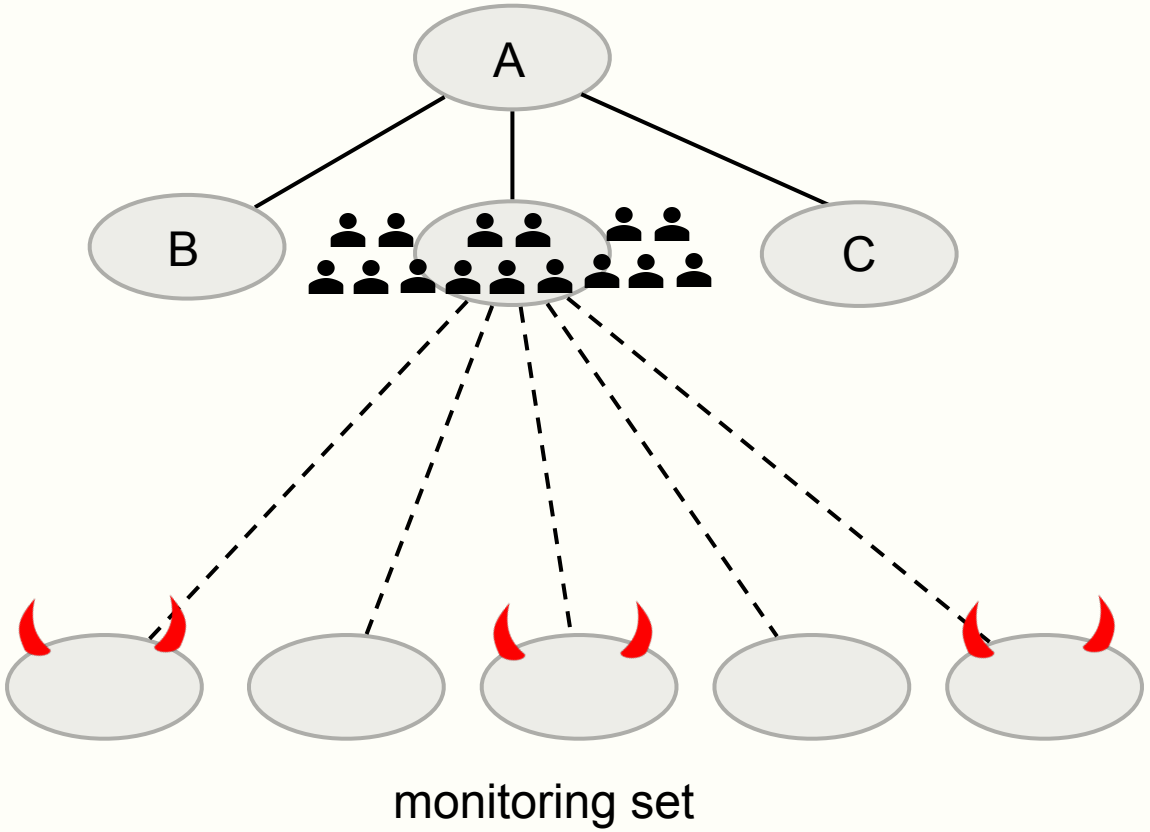
Shadow Elections



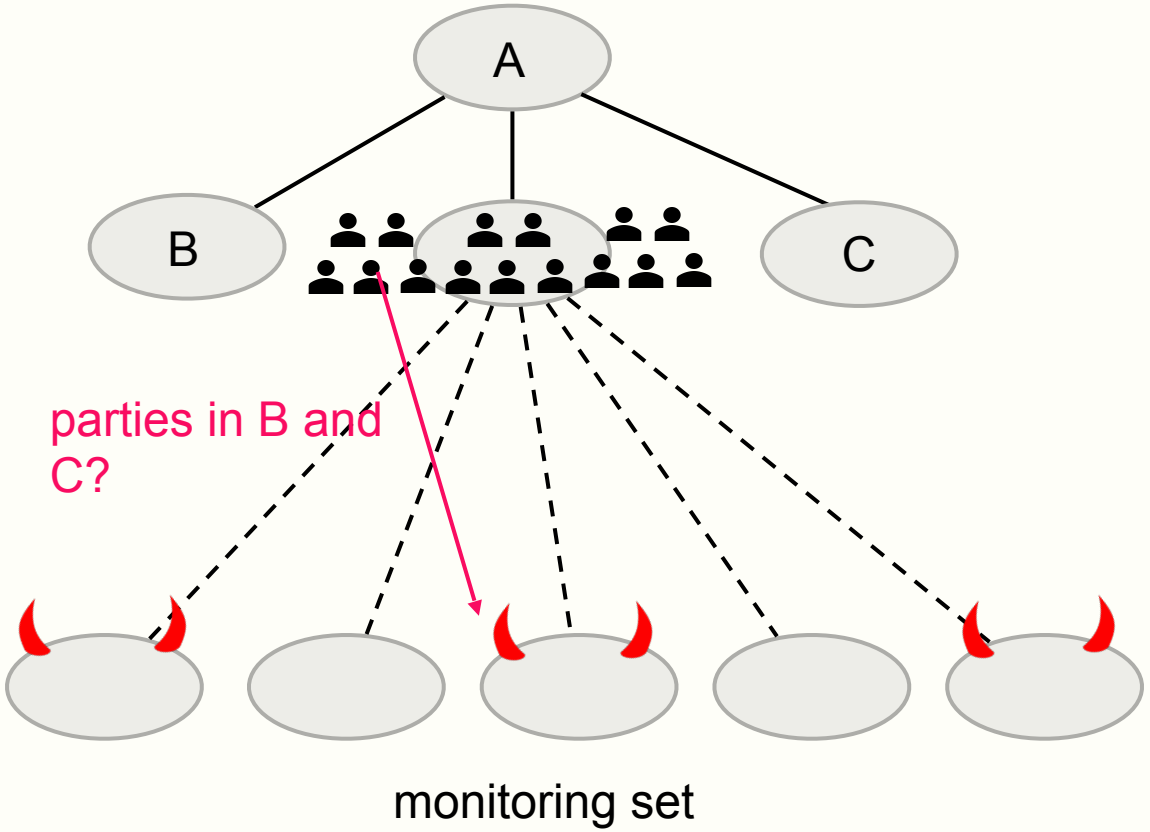
Shadow Elections



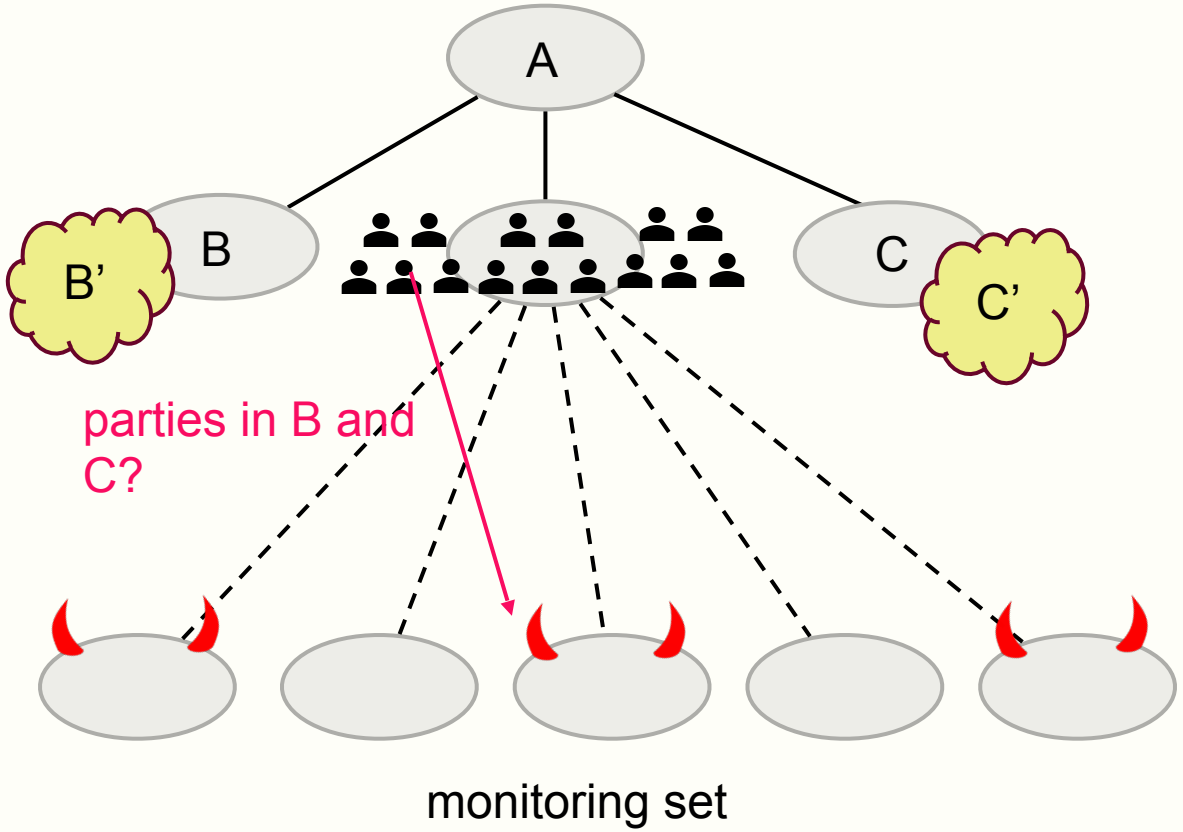
Shadow Elections



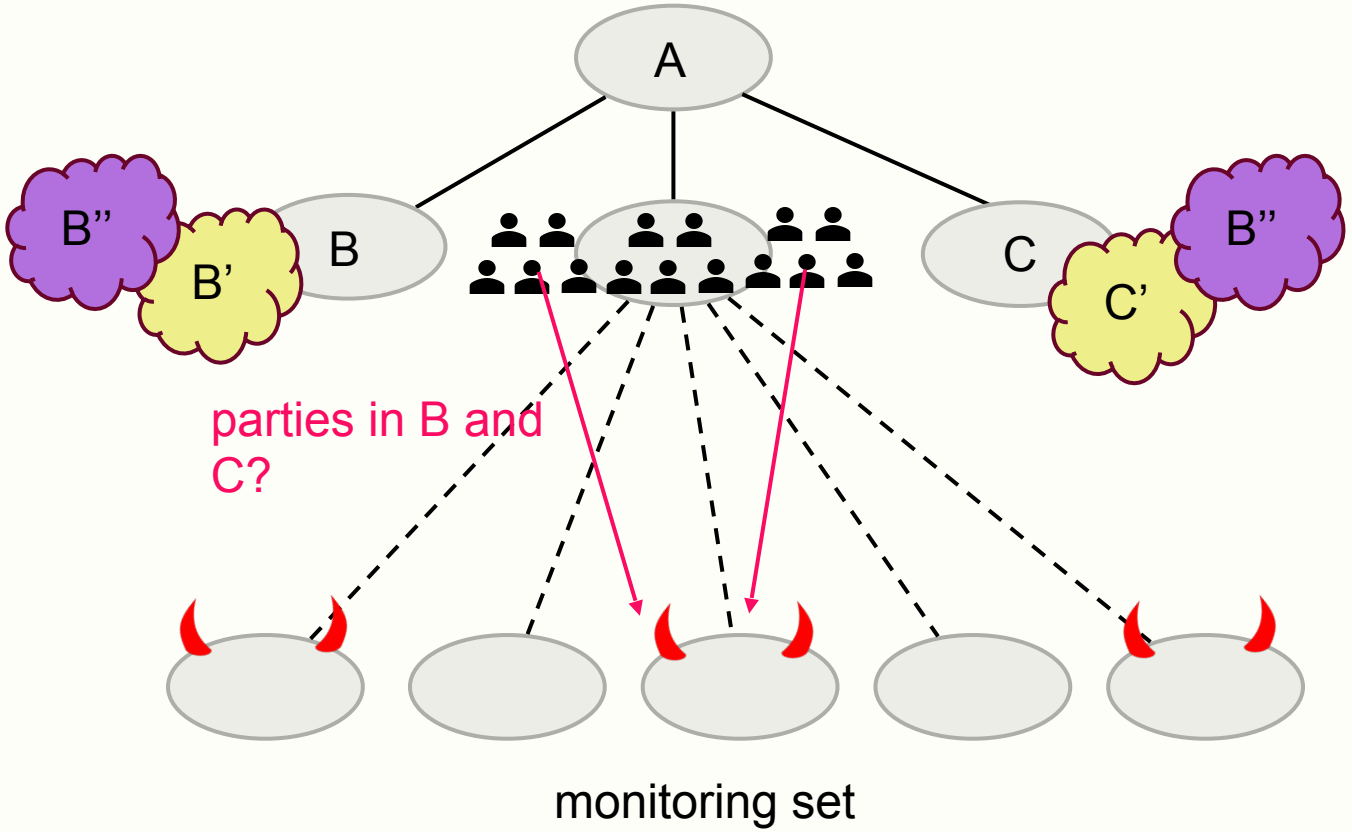
Shadow Elections



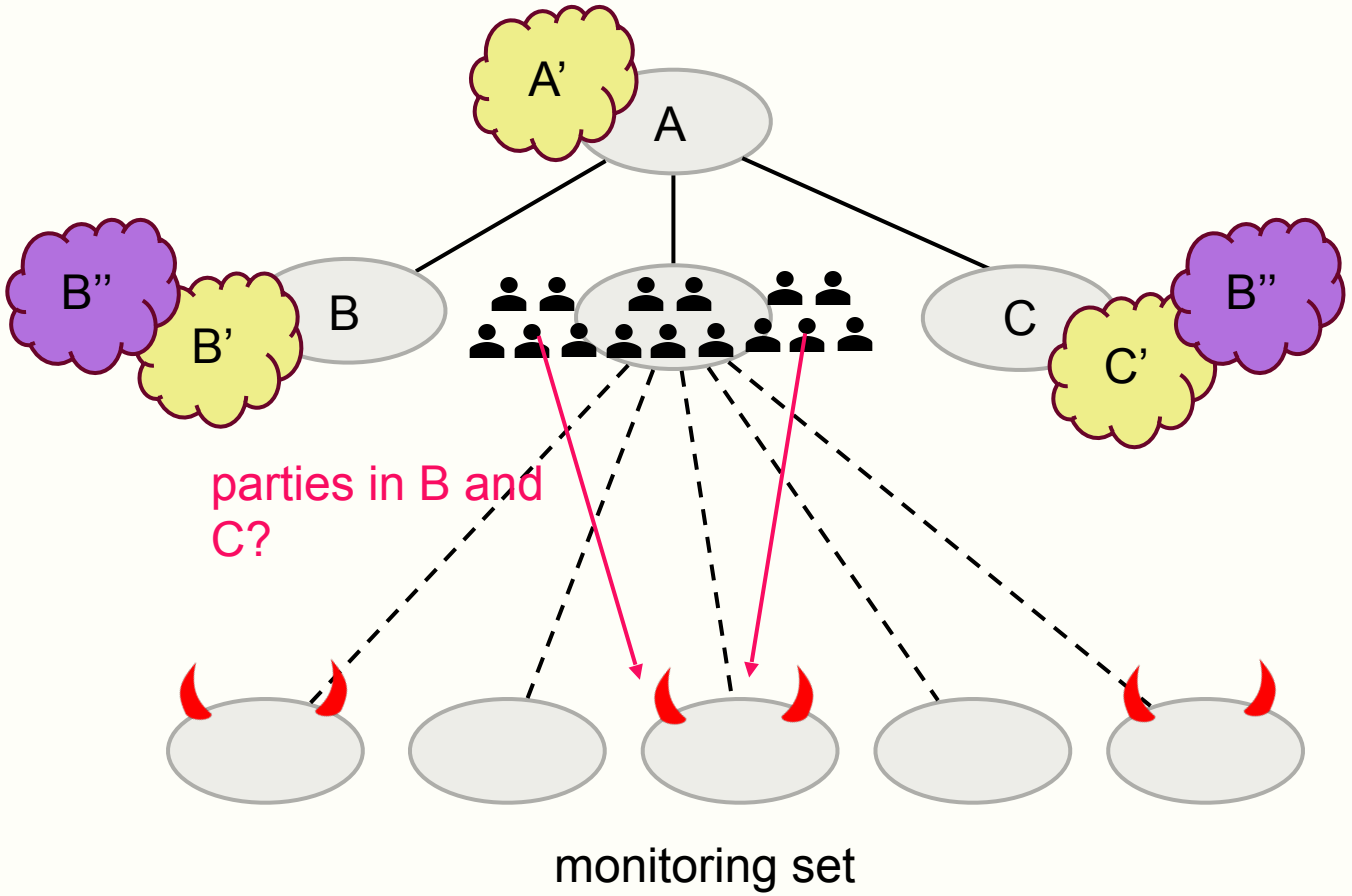
Shadow Elections



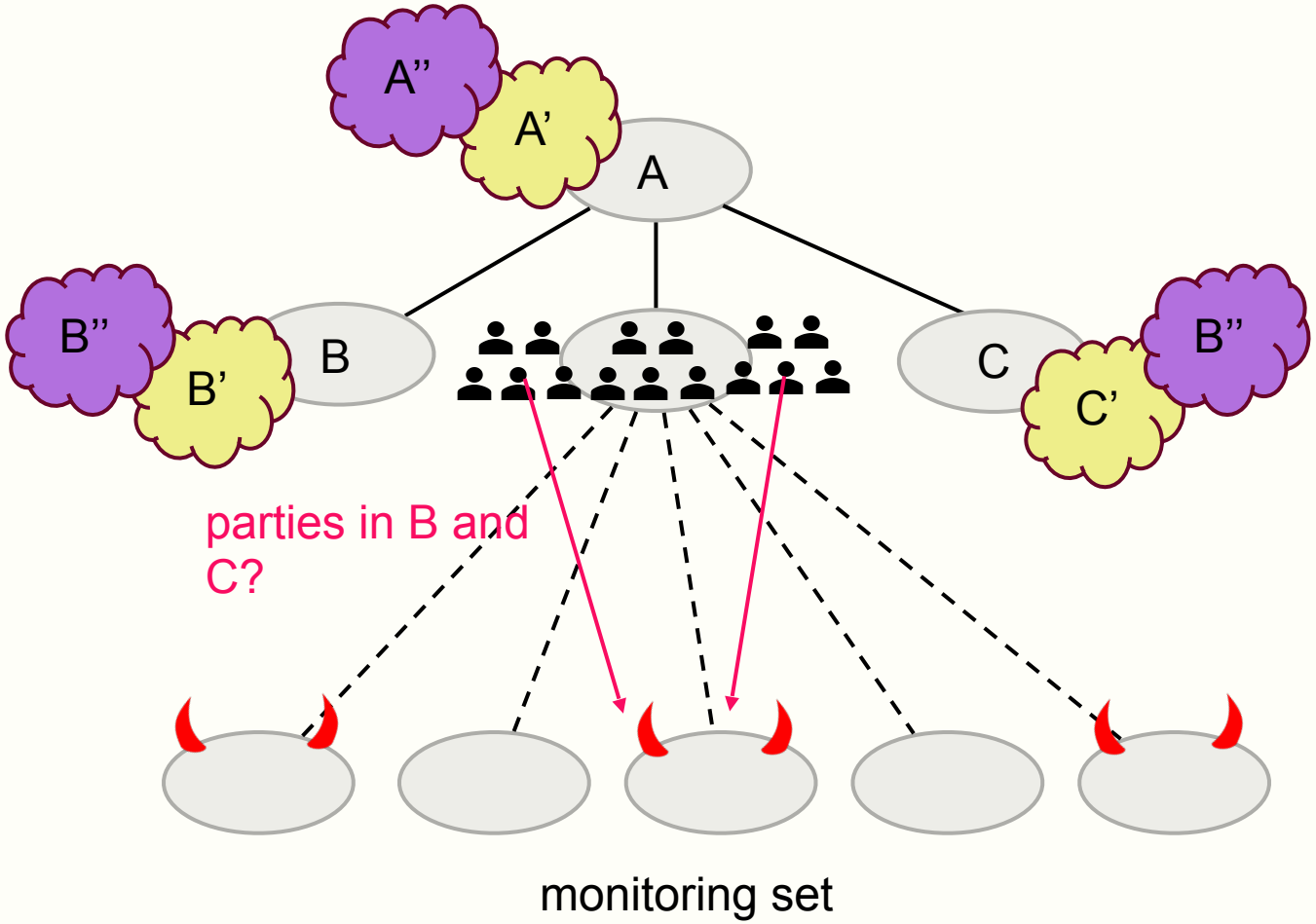
Shadow Elections



Shadow Elections

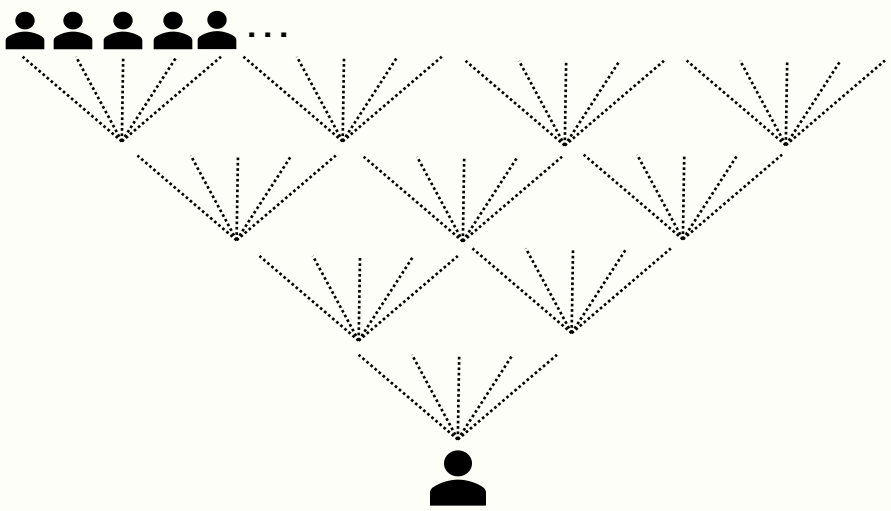


Shadow Elections

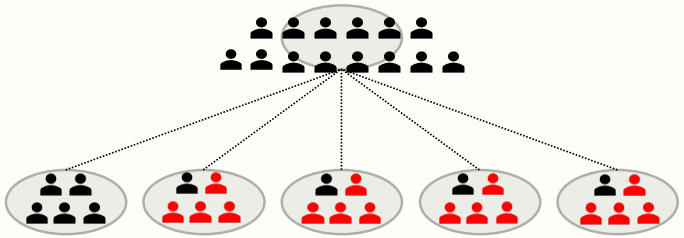


The Flaw in KSSV

Load balancing: parties go silent after winning 8 elections on a layer



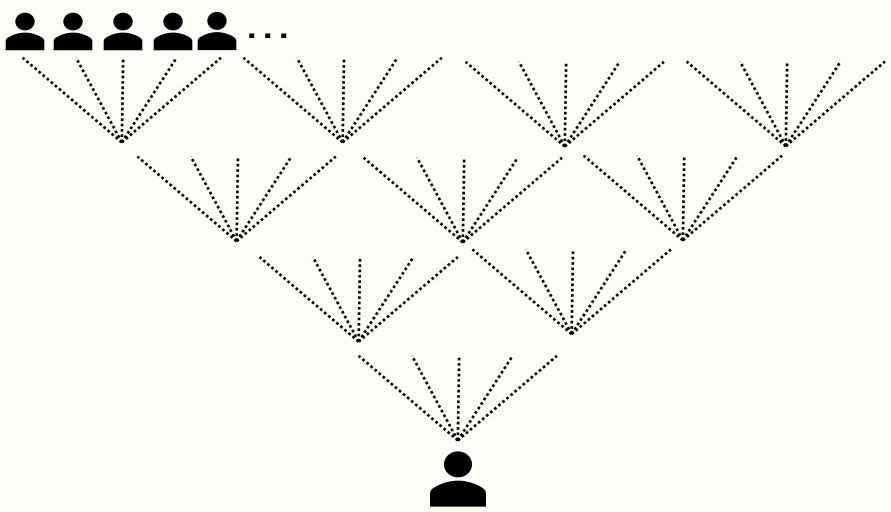
Bad elections: if >1/3 fraction of parties are Byzantine, no guarantees on election outcome



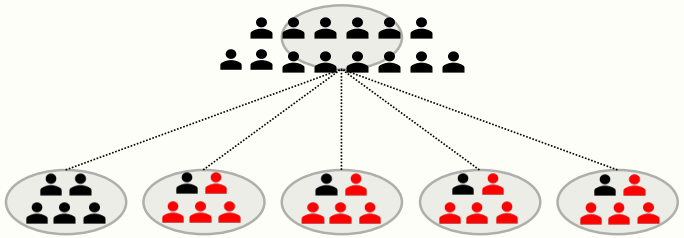
= *every participating party thinks they are elected*

The Flaw in KSSV

Load balancing: parties go silent after winning 8 elections on a layer



Bad elections: if >1/3 fraction of parties are Byzantine, no guarantees on election outcome



= *every participating party thinks they are elected*

The protocol may not terminate!

Strawman Fixes

Strawman Fixes

1. **Parties should realize something went wrong from monitoring sets?**

   - Up to δ nodes with “bad” monitoring sets per layer

Strawman Fixes

1. **Parties should realize something went wrong from monitoring sets?**

   - Up to δ nodes with “bad” monitoring sets per layer

2. **Reducing the # of bad elections per layer (δ) via sampler properties?**

Strawman Fixes

1. **Parties should realize something went wrong from monitoring sets?**

   - Up to δ nodes with "bad" monitoring sets per layer

2. **Reducing the # of bad elections per layer (δ) via sampler properties?**
3. **Stop silencing or increase silencing threshold?**

Strawman Fixes

1. **Parties should realize something went wrong from monitoring sets?**

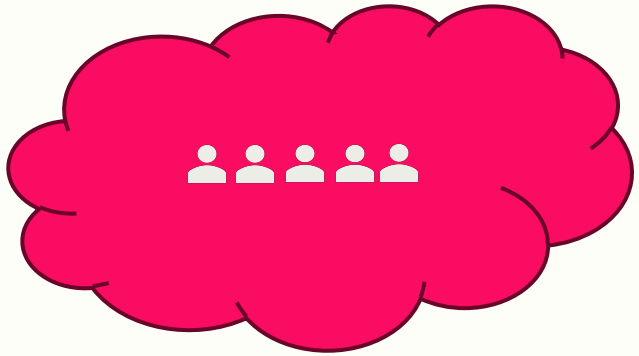
   - Up to δ nodes with "bad" monitoring sets per layer

2. **Reducing the # of bad elections per layer (δ) via sampler properties?**
3. **Stop silencing or increase silencing threshold?**

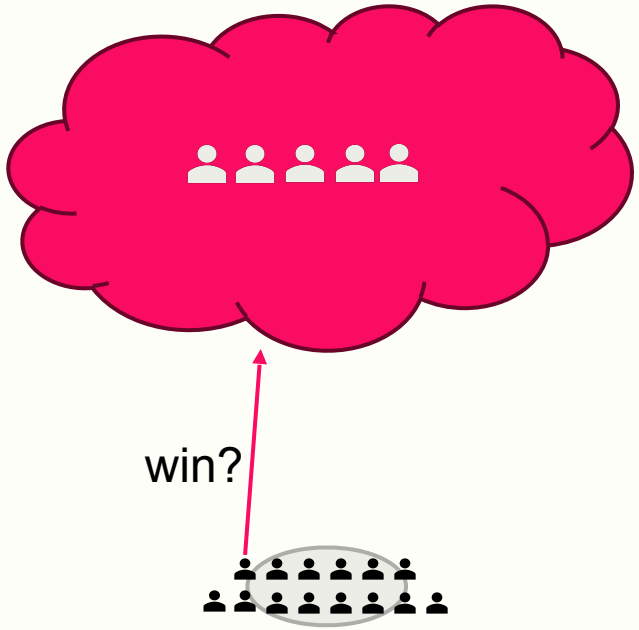
Our Fix: Intuition

Our Fix: Intuition

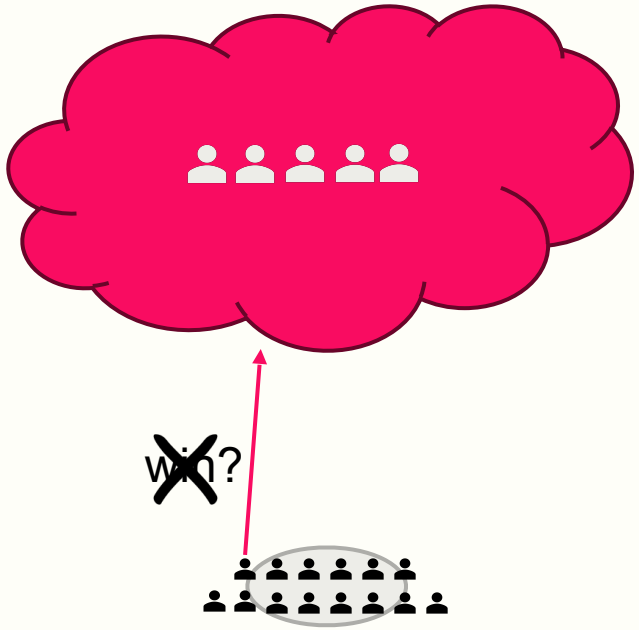
Our Fix: Intuition



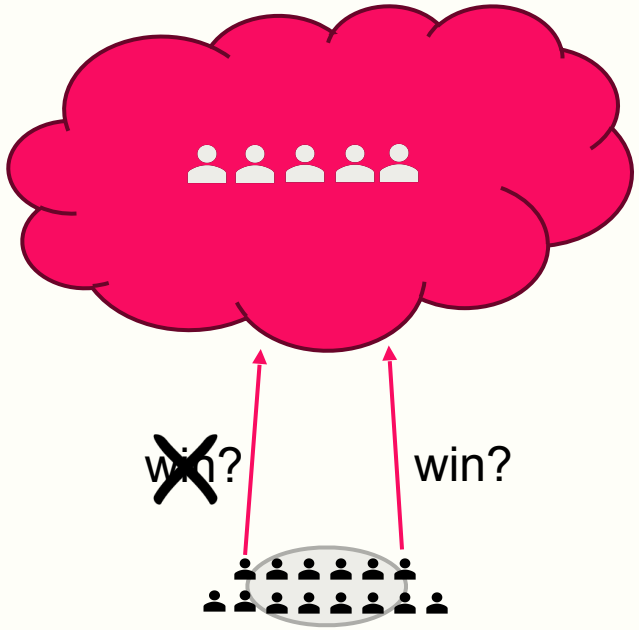
Our Fix: Intuition



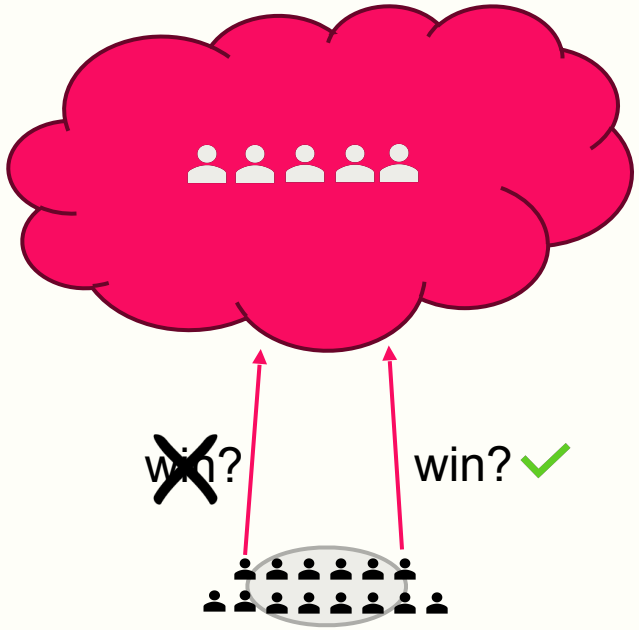
Our Fix: Intuition



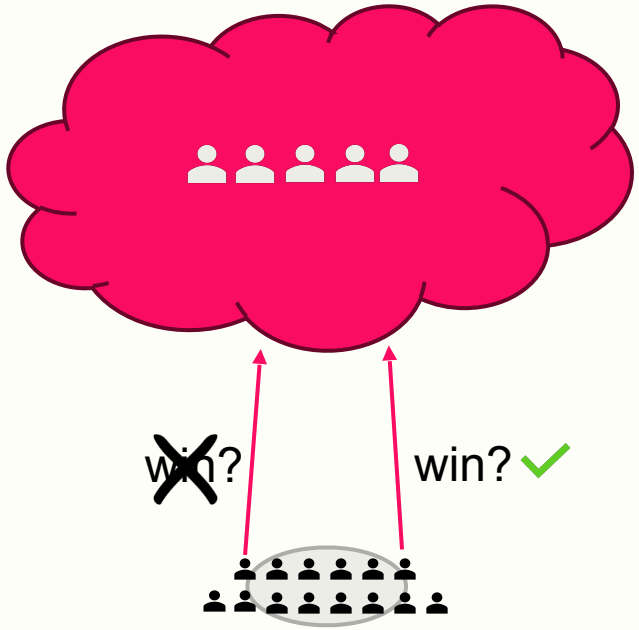
Our Fix: Intuition



Our Fix: Intuition

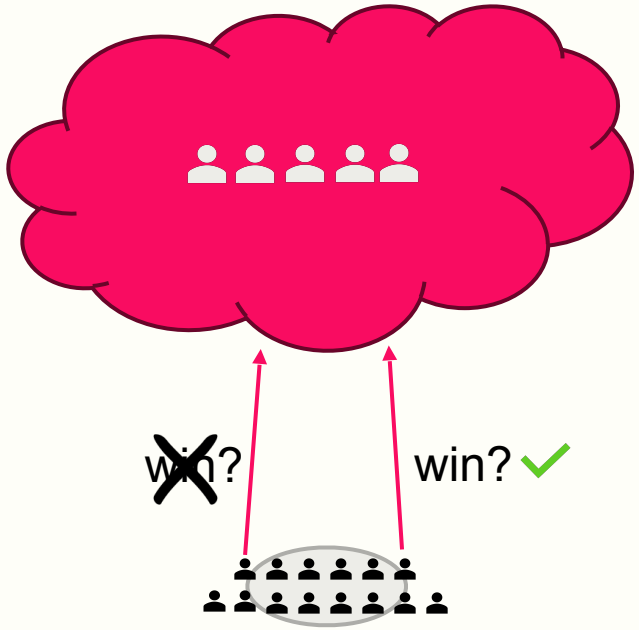


Our Fix: Intuition



monitoring sets?

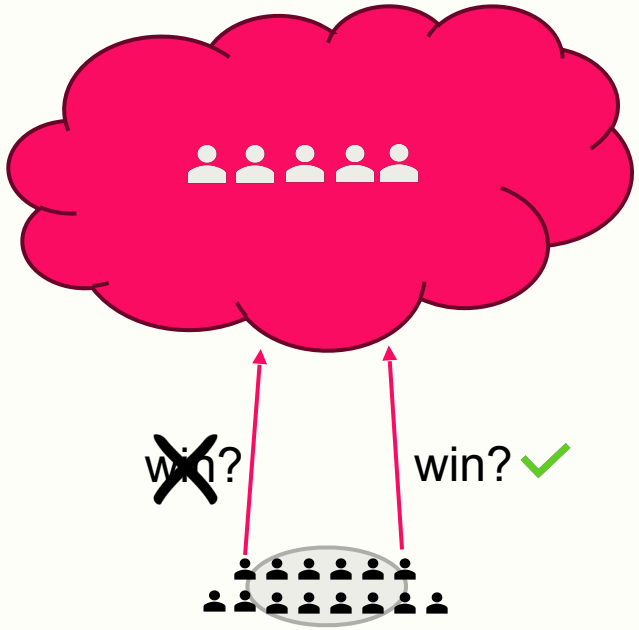
Our Fix: Intuition



monitoring sets?

- δ too large – many nodes have bad monitoring sets

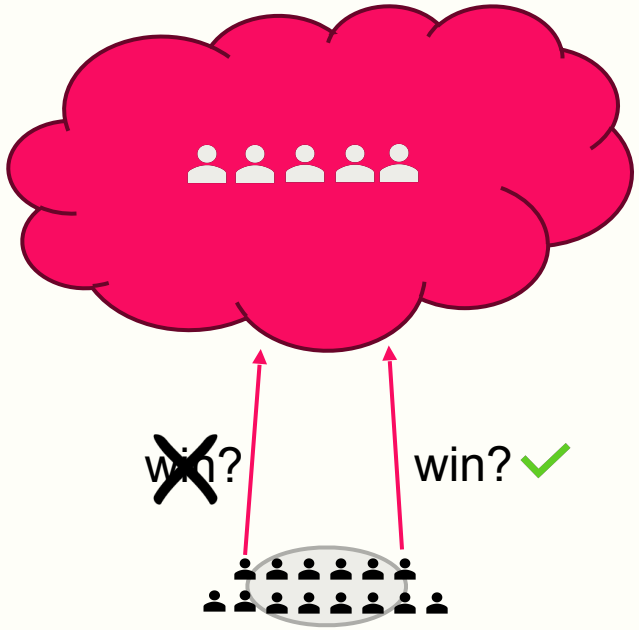
Our Fix: Intuition



~~monitoring sets?~~

- δ too large – many nodes have bad monitoring sets

Our Fix: Intuition

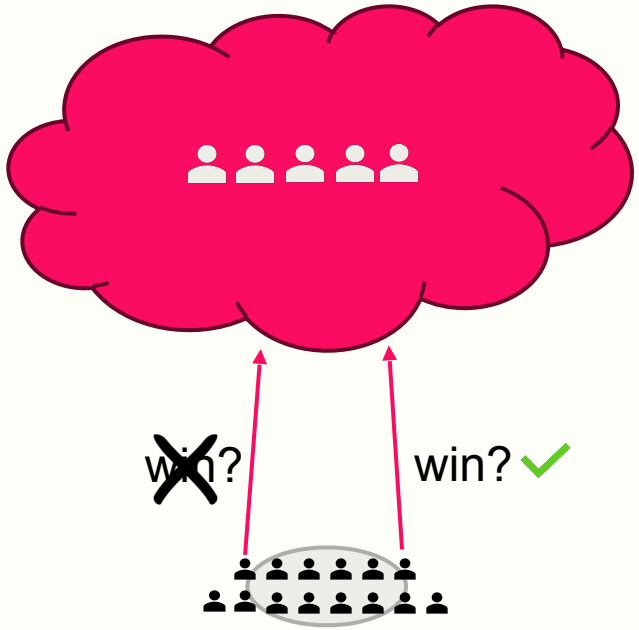


~~monitoring sets?~~

- δ too large – many nodes have bad monitoring sets

groups of monitoring sets

Our Fix: Intuition



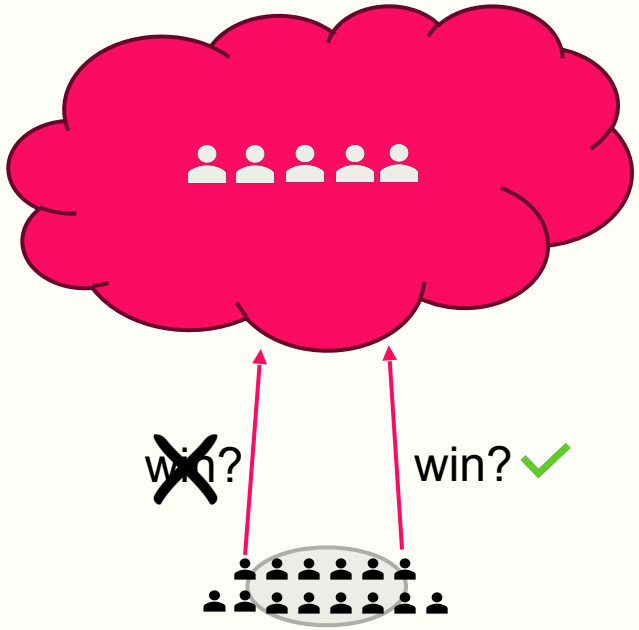
~~monitoring sets?~~

- δ too large – many nodes have bad monitoring sets

groups of monitoring sets

- **<<**δ nodes per layer have bad *groups* of monitoring sets

Our Fix: Intuition



~~monitoring sets?~~

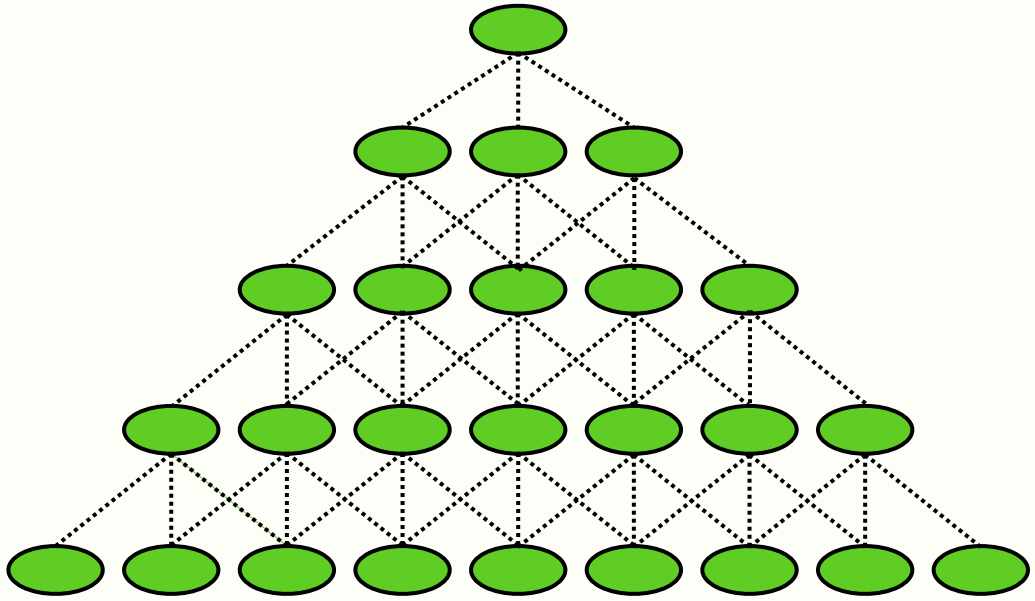
- δ too large – many nodes have bad monitoring sets

groups of monitoring sets

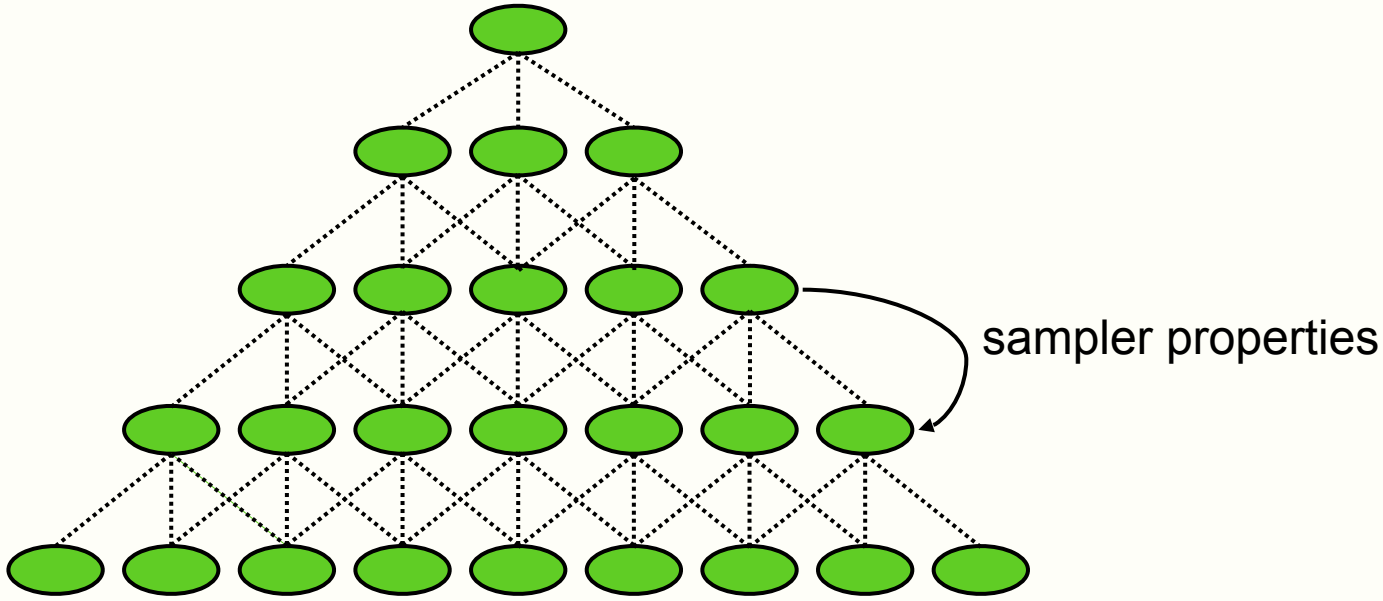
- **<<**δ nodes per layer have bad *groups* of monitoring sets
- decrease problematic nodes per layer *without* increasing degree of network

Our Fix: Good Expanders

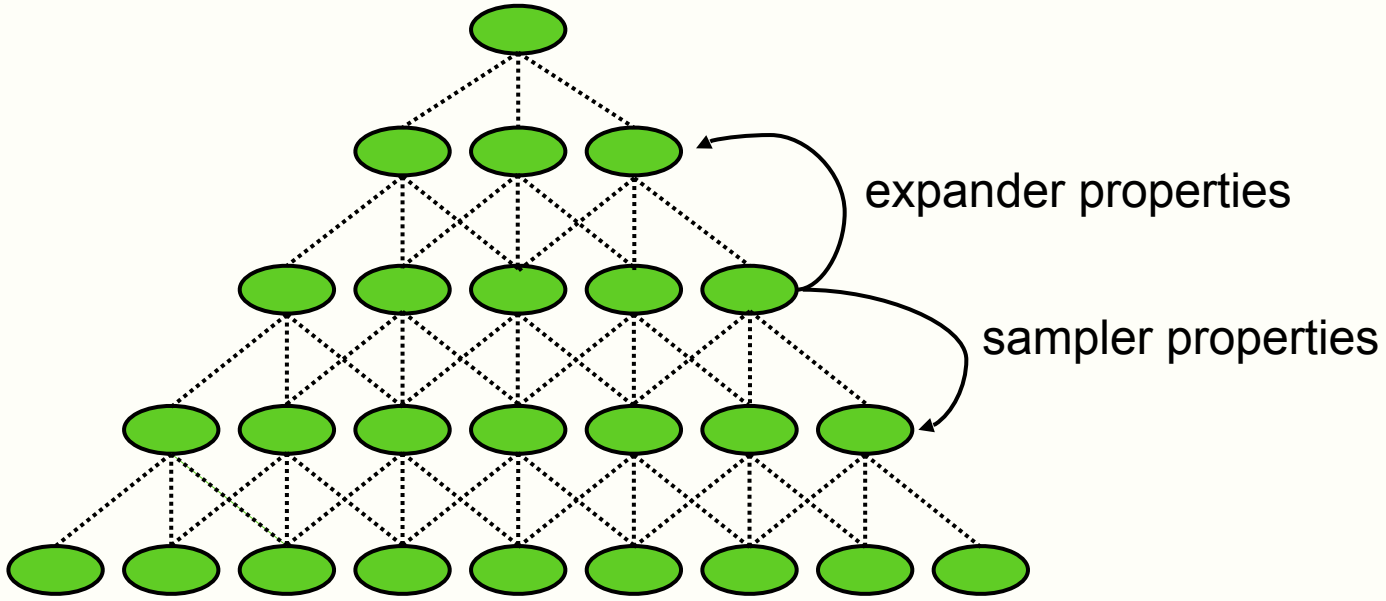
Our Fix: Good Expanders



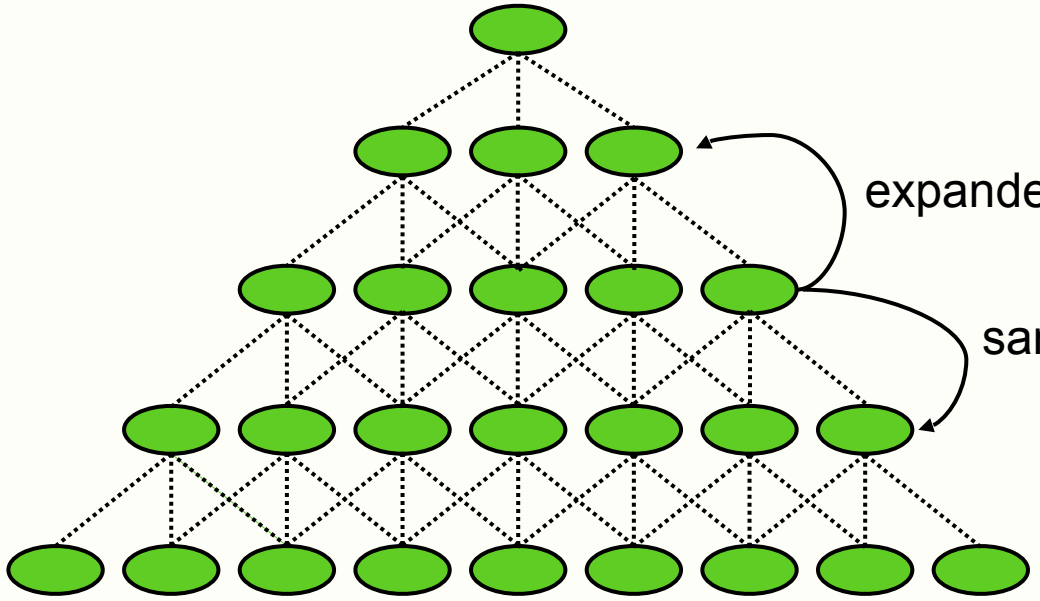
Our Fix: Good Expanders



Our Fix: Good Expanders



Our Fix: Good Expanders

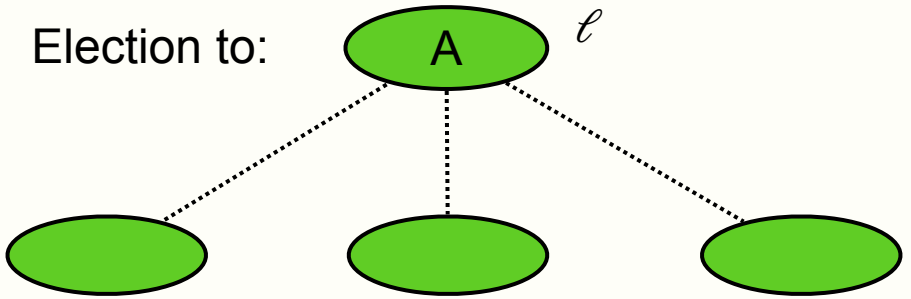


use monitoring sets of nodes 3 layers above for confirming election winners

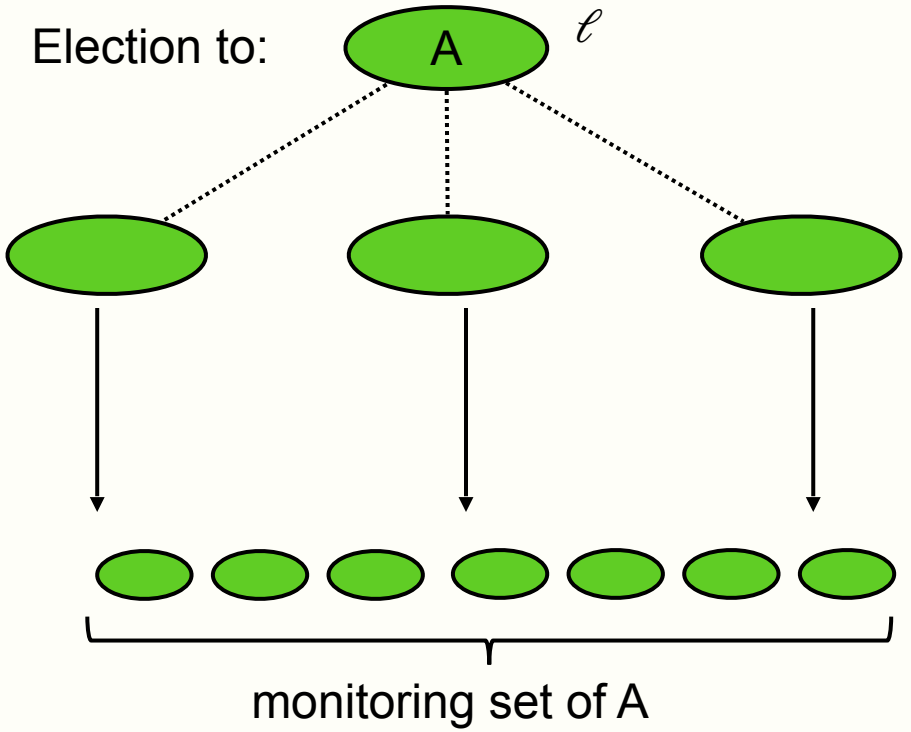
Our Fix: Election Winner Confirmation

Our Fix: Election Winner Confirmation

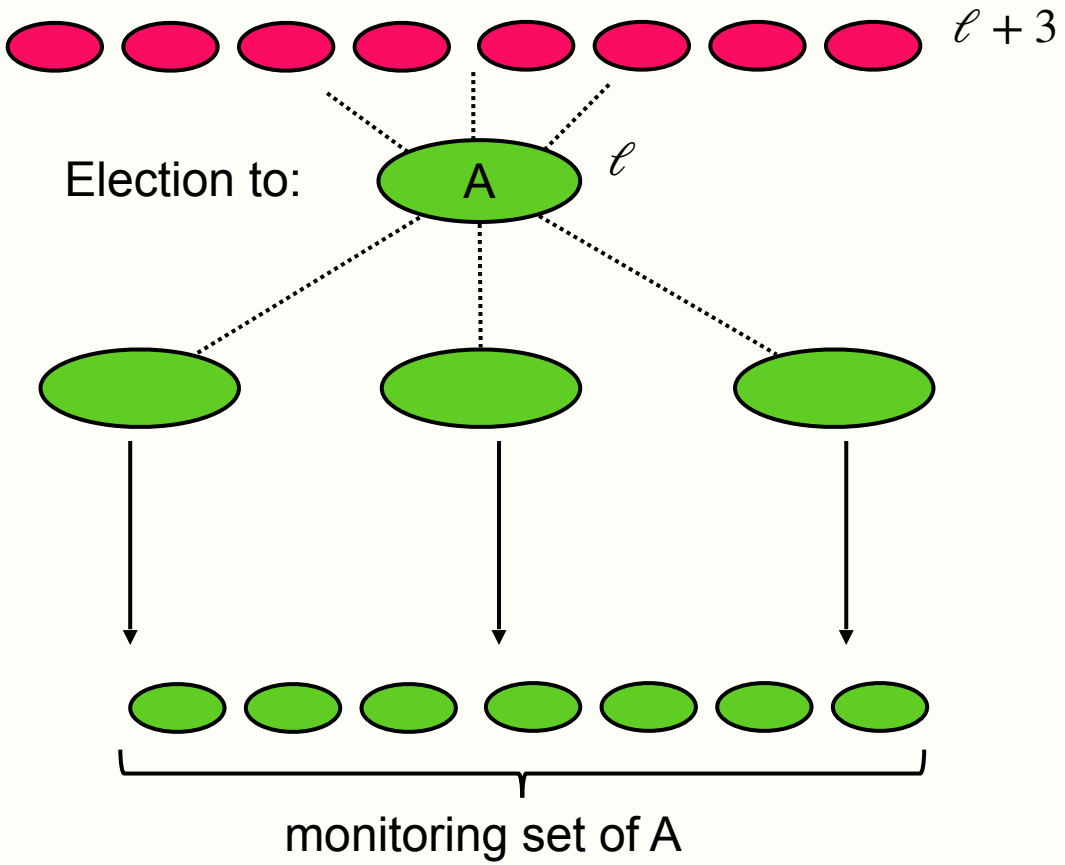
Our Fix: Election Winner Confirmation



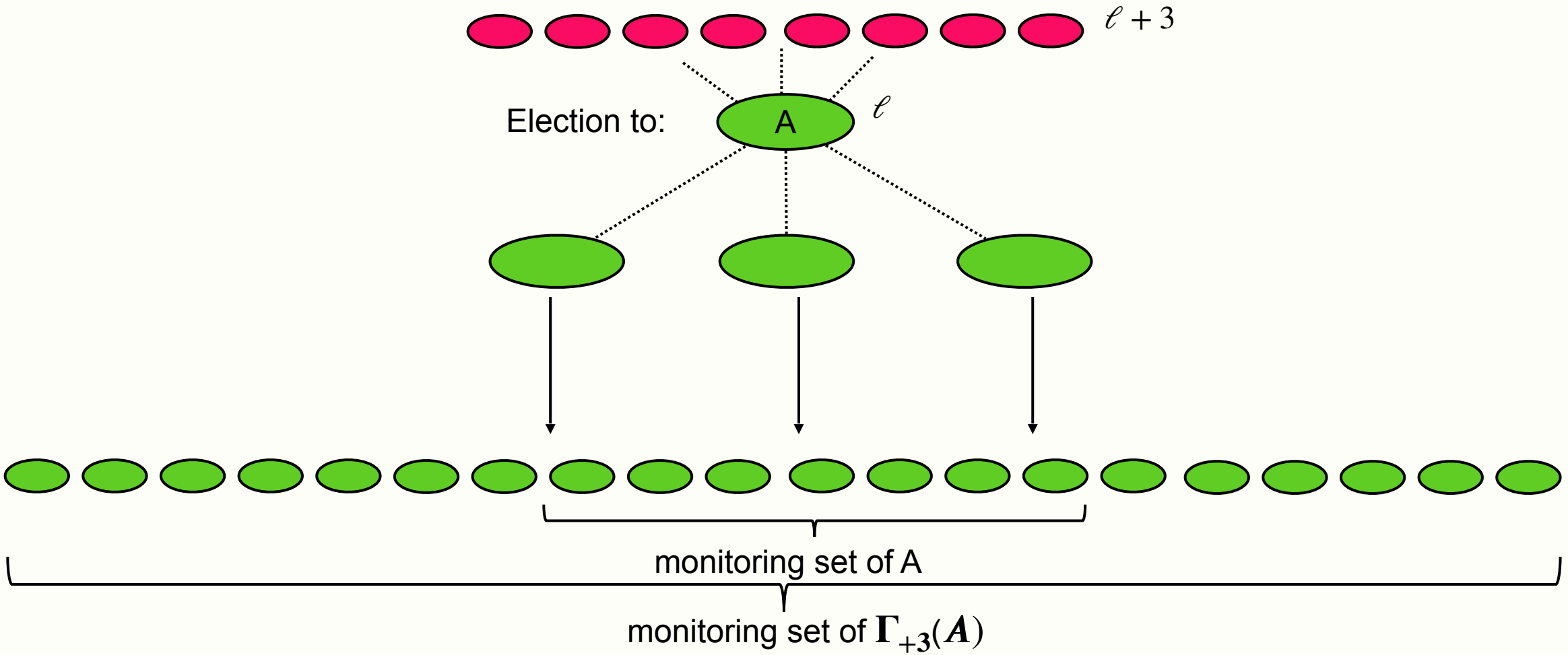
Our Fix: Election Winner Confirmation



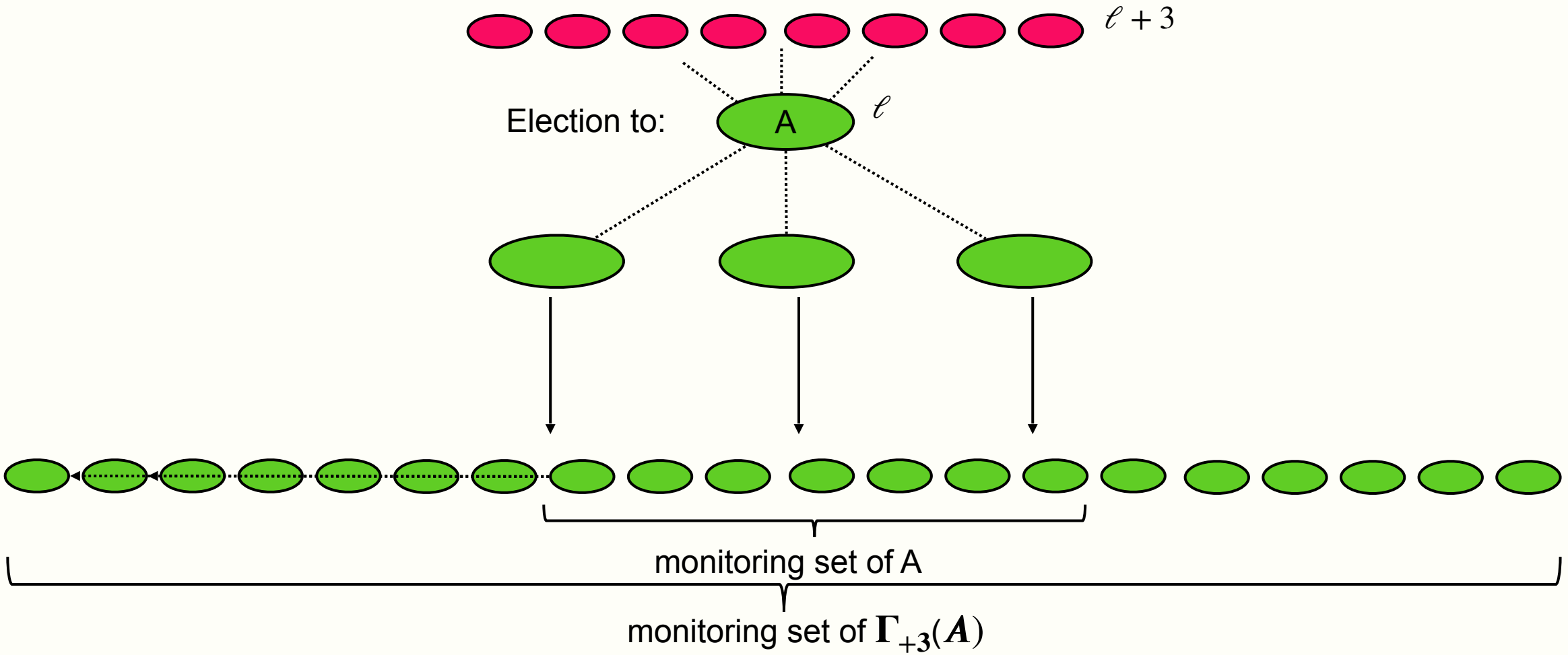
Our Fix: Election Winner Confirmation



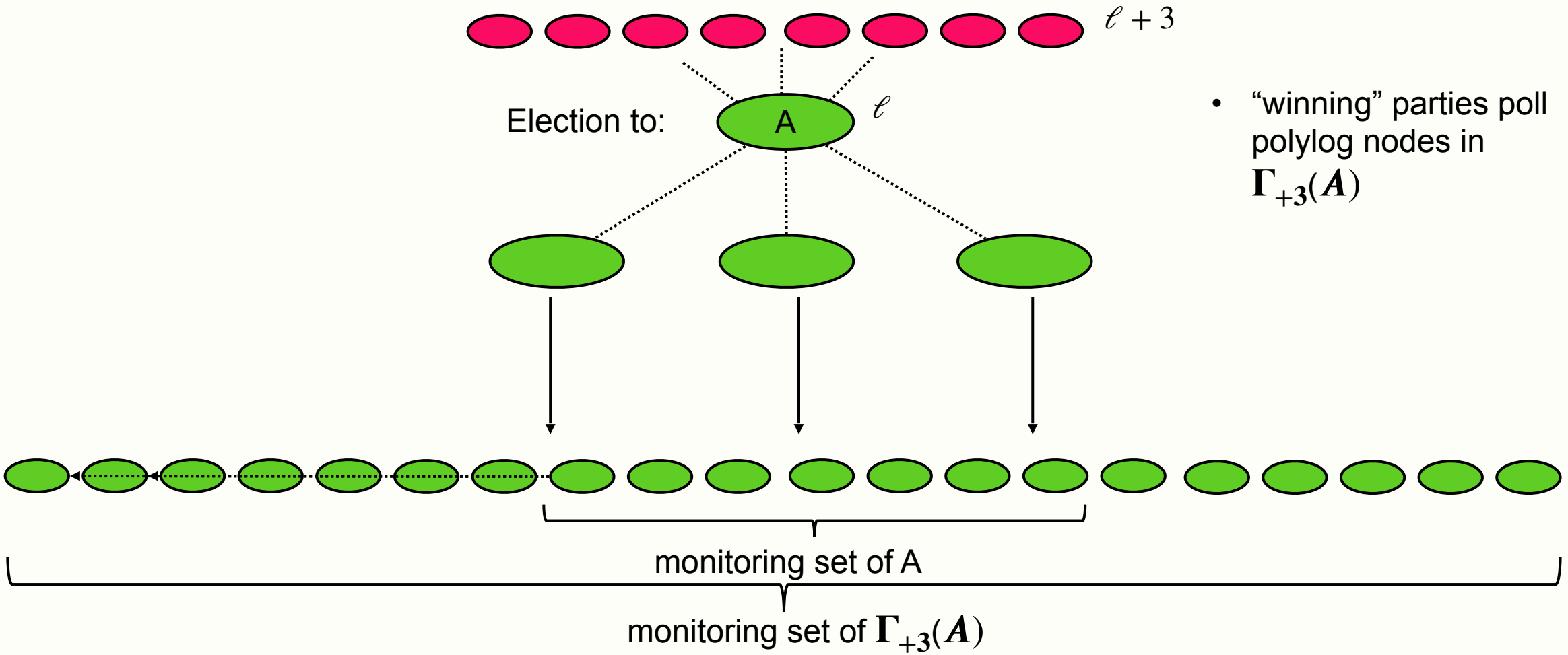
Our Fix: Election Winner Confirmation



Our Fix: Election Winner Confirmation

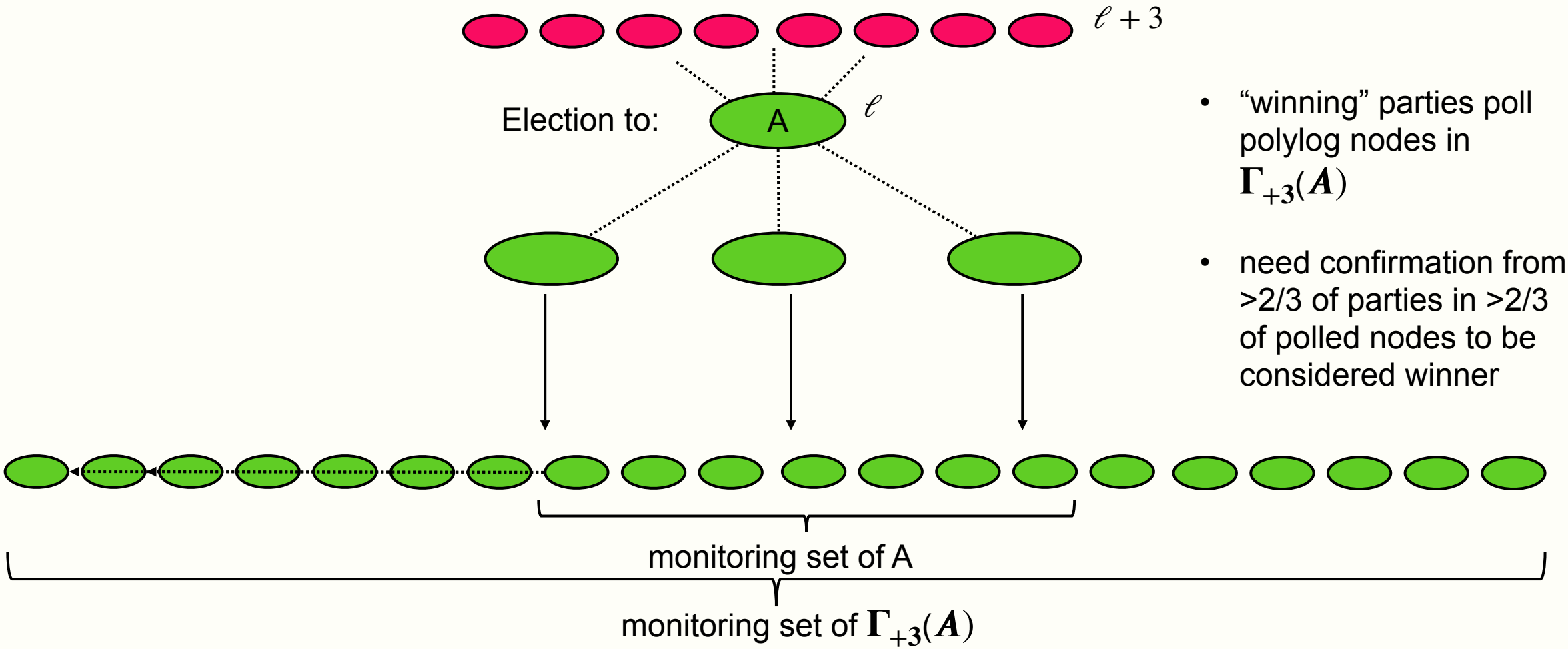


Our Fix: Election Winner Confirmation



- "winning" parties poll polylog nodes in $\Gamma_{+3}(A)$

Our Fix: Election Winner Confirmation



- "winning" parties poll polylog nodes in $\Gamma_{+3}(A)$
- need confirmation from >2/3 of parties in >2/3 of polled nodes to be considered winner

Analysis

Analysis

1. Reduce the number of nodes to which all parties can believe they were elected *without increasing degree of network*

Analysis

1. Reduce the number of nodes to which all parties can believe they were elected *without increasing degree of network*
   - *expander properties, election winner confirmation*

Analysis

1. Reduce the number of nodes to which all parties can believe they were elected *without increasing degree of network*
   - *expander properties, election winner confirmation*
2. Total number of honest parties that go silent erroneously (+ genuinely) sufficiently small on each layer

Analysis

1. Reduce the number of nodes to which all parties can believe they were elected *without increasing degree of network*
   - *expander properties, election winner confirmation*
2. Total number of honest parties that go silent erroneously (+ genuinely) sufficiently small on each layer
3. Protocol terminates with high probability

Analysis

1. Reduce the number of nodes to which all parties can believe they were elected *without increasing degree of network*
   - *expander properties, election winner confirmation*
2. Total number of honest parties that go silent erroneously (+ genuinely) sufficiently small on each layer
3. Protocol terminates with high probability

Sravya Yandamuri

sravya@commonprefix.com

Analysis

1. Reduce the number of nodes to which all parties can believe they were elected *without increasing degree of network*
   - *expander properties, election winner confirmation*
2. Total number of honest parties that go silent erroneously (+ genuinely) sufficiently small on each layer
3. Protocol terminates with high probability

Sravya Yandamuri

sravya@commonprefix.com

Analysis

1. Reduce the number of nodes to which all parties can believe they were elected *without increasing degree of network*
   - *expander properties, election winner confirmation*
2. Total number of honest parties that go silent erroneously (+ genuinely) sufficiently small on each layer
3. Protocol terminates with high probability

Sravya Yandamuri

sravya@commonprefix.com