Optimal Communication Complexity of Authenticated Byzantine Agreement

Atsuki Momose * †, Ling Ren‡

Nagoya University

Intelligent Systems Laboratory, SECOM CO., LTD.†

University of Illinois at Urbana-Champaign‡

Byzantine Agreement (BA)

A set of parties $\{r_1, .., r_n\}$ have input values $\{x_1, .., x_n\}$, and agree on a single output .

- At most f parties are faulty and behave arbitrarily — Byzantine fault.
- Consistency. Honest parties do not output different values.
- Termination. Every honest party eventually outputs a value.
- Validity. If every honest party has the same input value ,
  every honest party outputs the value $y = b$ — Unanimity.

Byzantine Agreement (BA)

Unauthenticated model.

- No cryptography, i.e., information-theoretic security.
- $f < n/3$ is the best possible.

Authenticated model.

- Assume cryptography, e.g., digital signature with PKI.
- $f < n/2$ is the best possible.

Communication Complexity

The maximum amount of bits transferred by all honest parties combined across all executions–worst-case communication cost.

- All parties multicast $O(1)$ messages, i.e., all-to-all communication

  $\rightarrow O(n^2)$ communication

- All-to-all communication with $O(n)$ messages (e.g., a quorum of votes)

  $\rightarrow O(n^3)$ communication

Communication Complexity of BA

Model	Fault-tolerace	Lower Bound	Upper Bound
unauthenticated	$f < n/3$	$\Omega(n^2)$ [Dolev-Resichuk]	$O(n^2)$ [Berman et al.]
authenticated (PKI)	$f < n/2$		$O(n^3)$ [Dolev-Strong]

Communication Complexity of BA

> 0 : any constant

Model	Fault-tolerace	Lower Bound	Upper Bound
unauthenticated	$f < n/3$	$\Omega(n^2)$ [Dolev-Resichuk]	$O(n^2)$ [Berman et al.]
authenticated (PKI)	$f < n/2$		$O(n^3)$ [Dolev-Strong]
authenticated (trusted setup)	$f < n/2$		$O(n^2)$ this work
authenticated (PKI)	$f < (1/2 - \varepsilon)n$		$O(n^2)$ this work

Other Assumptions

Lockstep synchrony model.

- Every party runs at the same clock speed
  → a clock step is called round

- All message sent by honest parties are delivered by the next round

Adaptive corruption.

- An adversary can corrupt parties anytime in the protocol execution

Outline

1. Achieving BA from Graded Agreement (GA)

- Berman et al's protocol is a problem reduction from BA to GA.

2. Solving GA for $f \geq n/3$

- Solution 1: GA with $f < n/2$ and trusted setup.
- Solution 2: GA with $f \leq (1/2 - \varepsilon)n$ and PKI.

Outline

1. Achieving BA from Graded Agreement (GA)

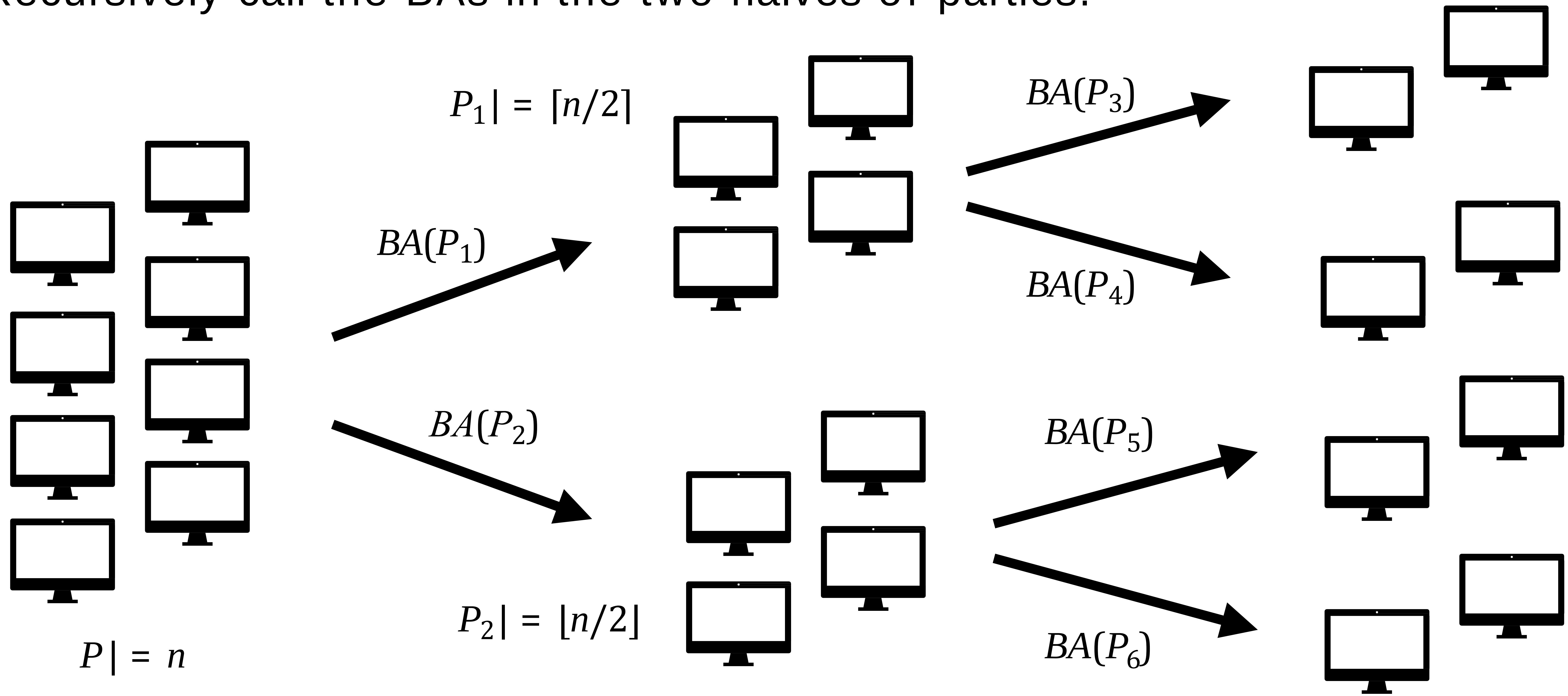
- Berman et al's protocol is a problem reduction from BA to GA.

2. Solving GA for $f \geq n/3$

- Solution 1: GA with $f < n/2$ and trusted setup.
- Solution 2: GA with $f \leq (1/2 - \varepsilon)n$ and PKI.

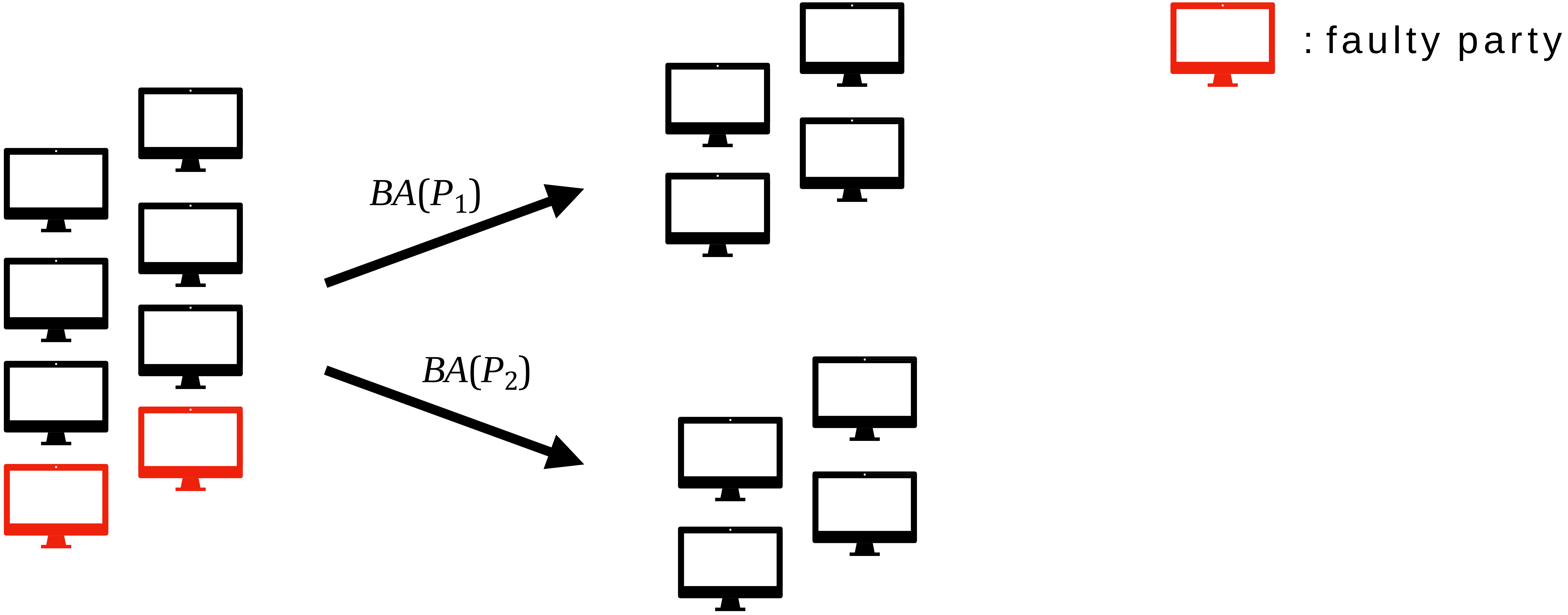
Berman et al.

Recursively call the BAs in the two halves of parties.



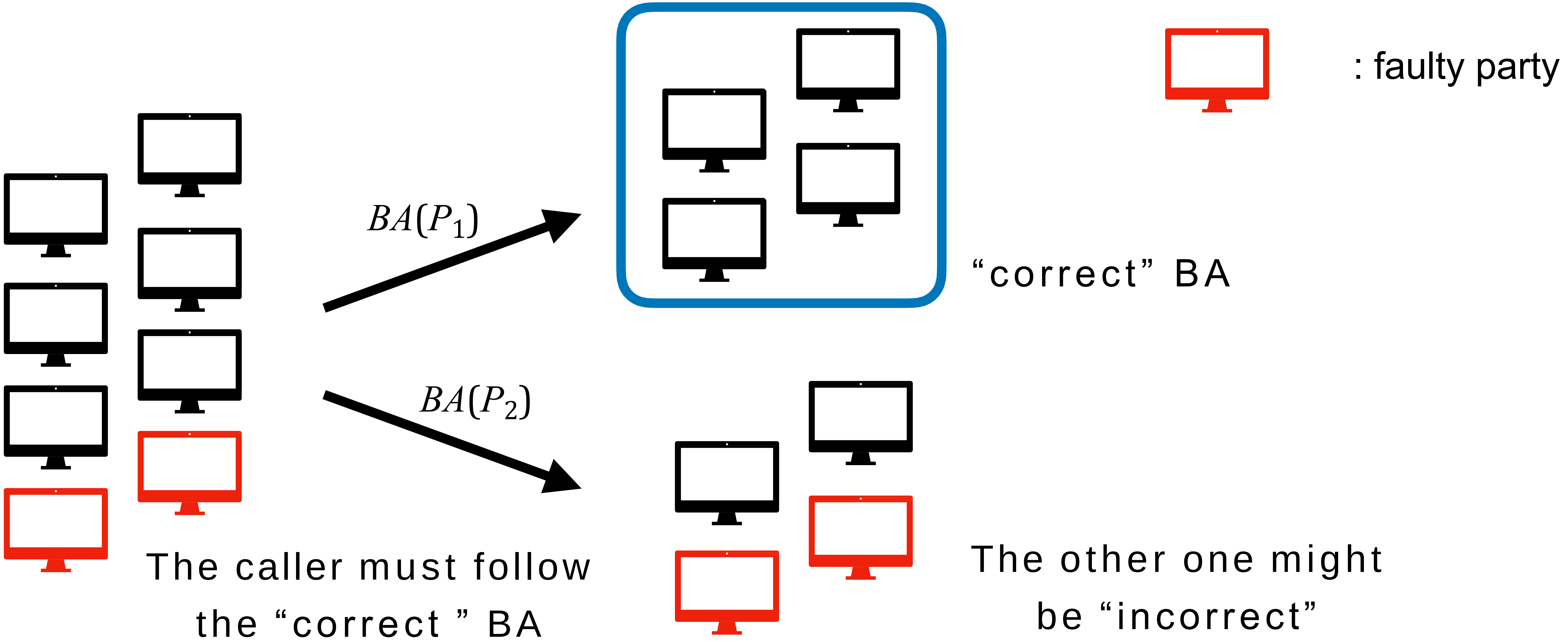
Berman et al.

One of two halves preserves the 1/3 fault fraction → “correct” BA exec.



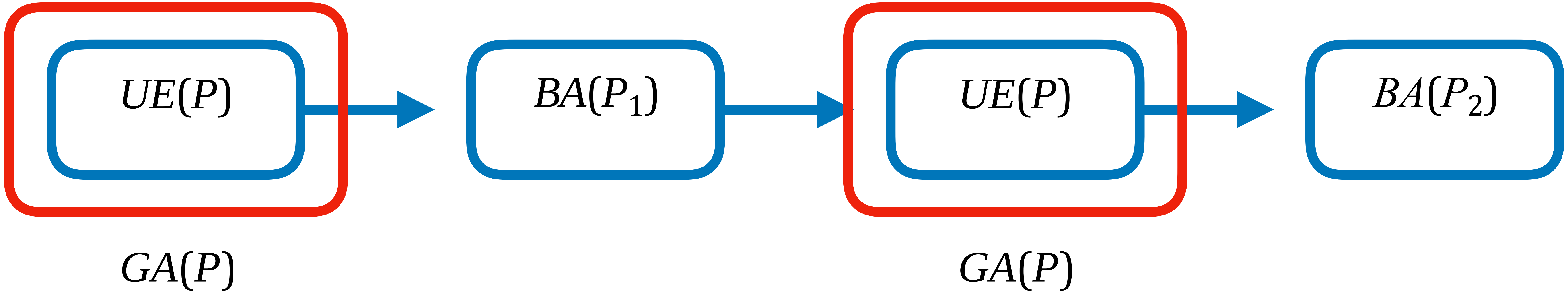
Berman et al.

One of two halves preserves the $1/3$ fault fraction → “correct” BA exec.



Berman et al.

The tailored Universal Exchange pre-process (for $f < n/3$) helps parties ignore the incorrect BA and follow the correct BA.



What the Universal Exchange achieves is the well-known problem called Graded Agreement.

Graded Agreement (GA)

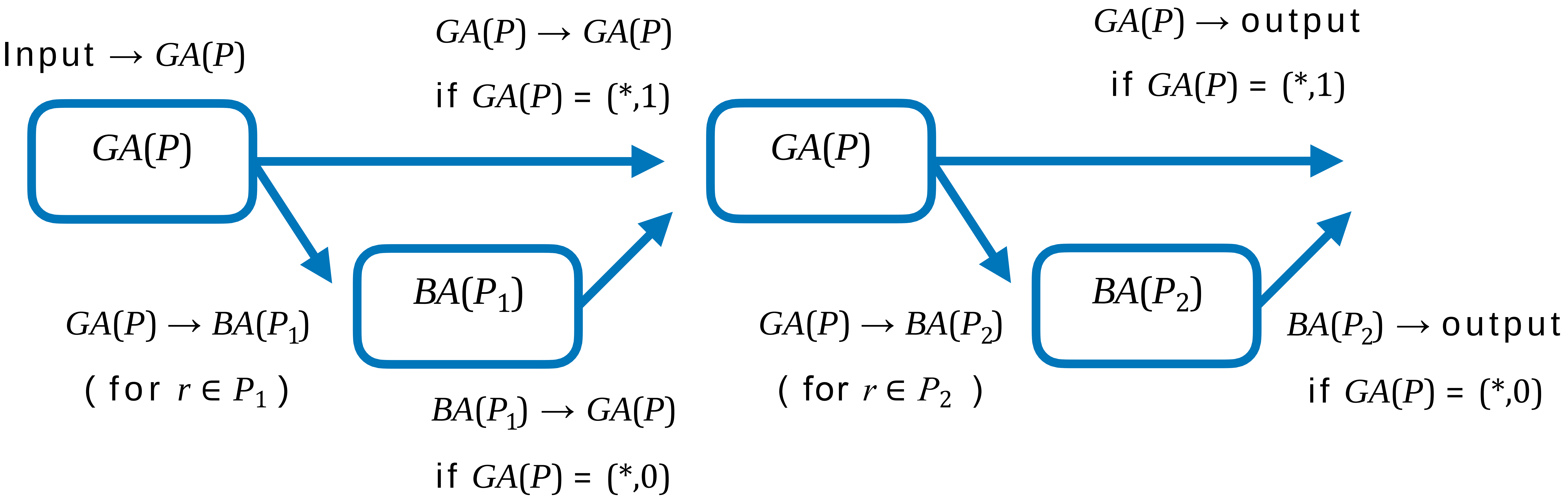
A set of parties $\{r_1, .., r_n\}$ have input values $\{x_1, .., x_n\}$, and each party outputs a pair (y, g) of value and a grade bit $g \in \{0,1\}$

- Consistency. If an honest party outputs $(y, 1)$, every honest party outputs $(y, *)$
- Validity. If every honest party has the same input value $x_i, .. x_j = b$, every honest party outputs $(b, 1)$
- Termination. Every honest party eventually outputs a pair.

BA in partition (i.e., $BA(P)$)

1. GA determines the input of BA

2. If GA outputs grade 1, ignore BA's output.

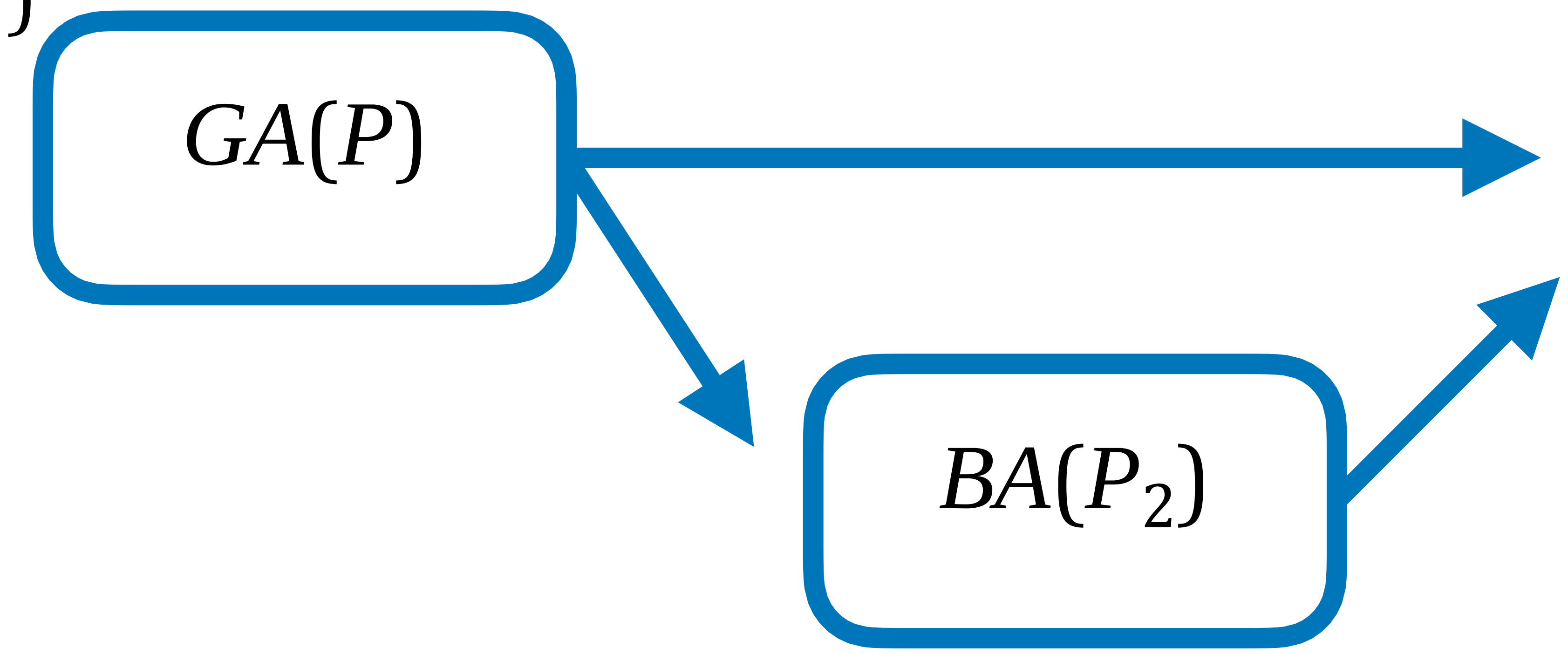


Idea 1: An agreed upon value will not be changed.

If all honest parties already agree on a value at the beginning of each step, they do not change the value in the step.

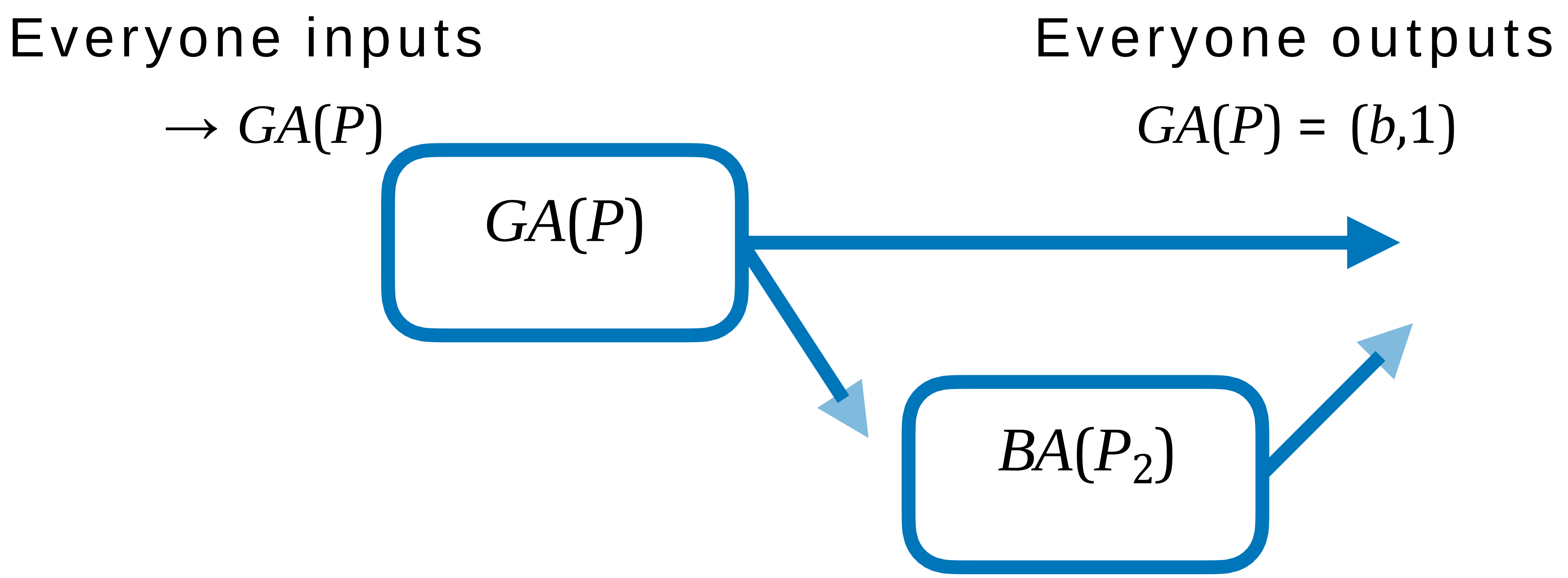
Everyone inputs

$\to GA(P)$



Idea 1: An agreed upon value will not be changed.

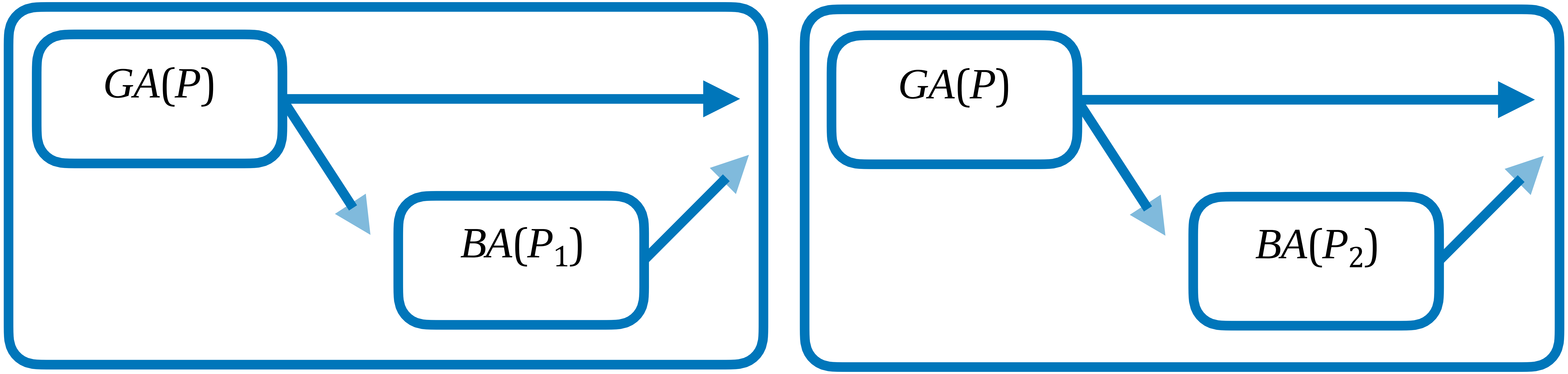
If all honest parties already agree on a value at the beginning of each step, they do not change the value in the step.



Validity

If all honest parties input the same value, they all output the value.

Input $\rightarrow GA(P)$ $\rightarrow$ Output



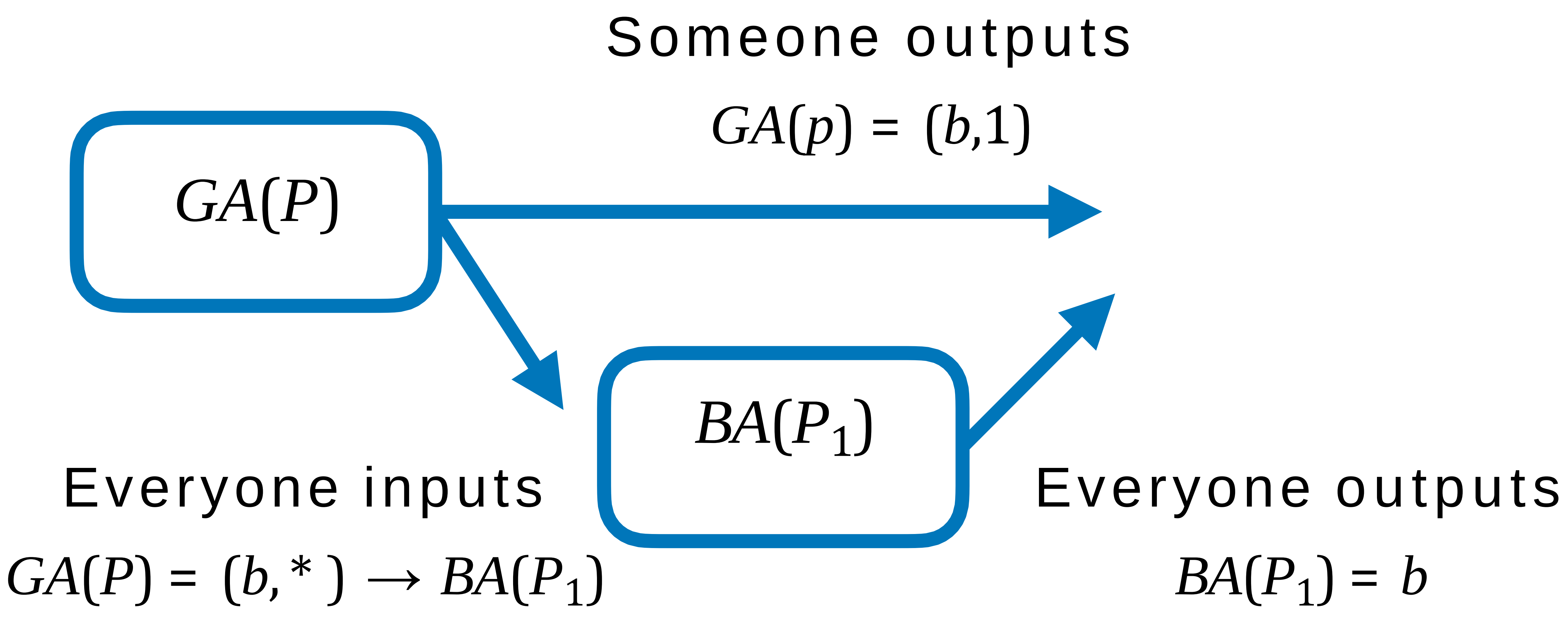
Step 1: Everyone outputs the agreed upon value

Step 2: Everyone outputs the agreed upon value

Idea 2: The "correct" step drives agreement

All honest parties agree on a value at the end of the "correct" step.

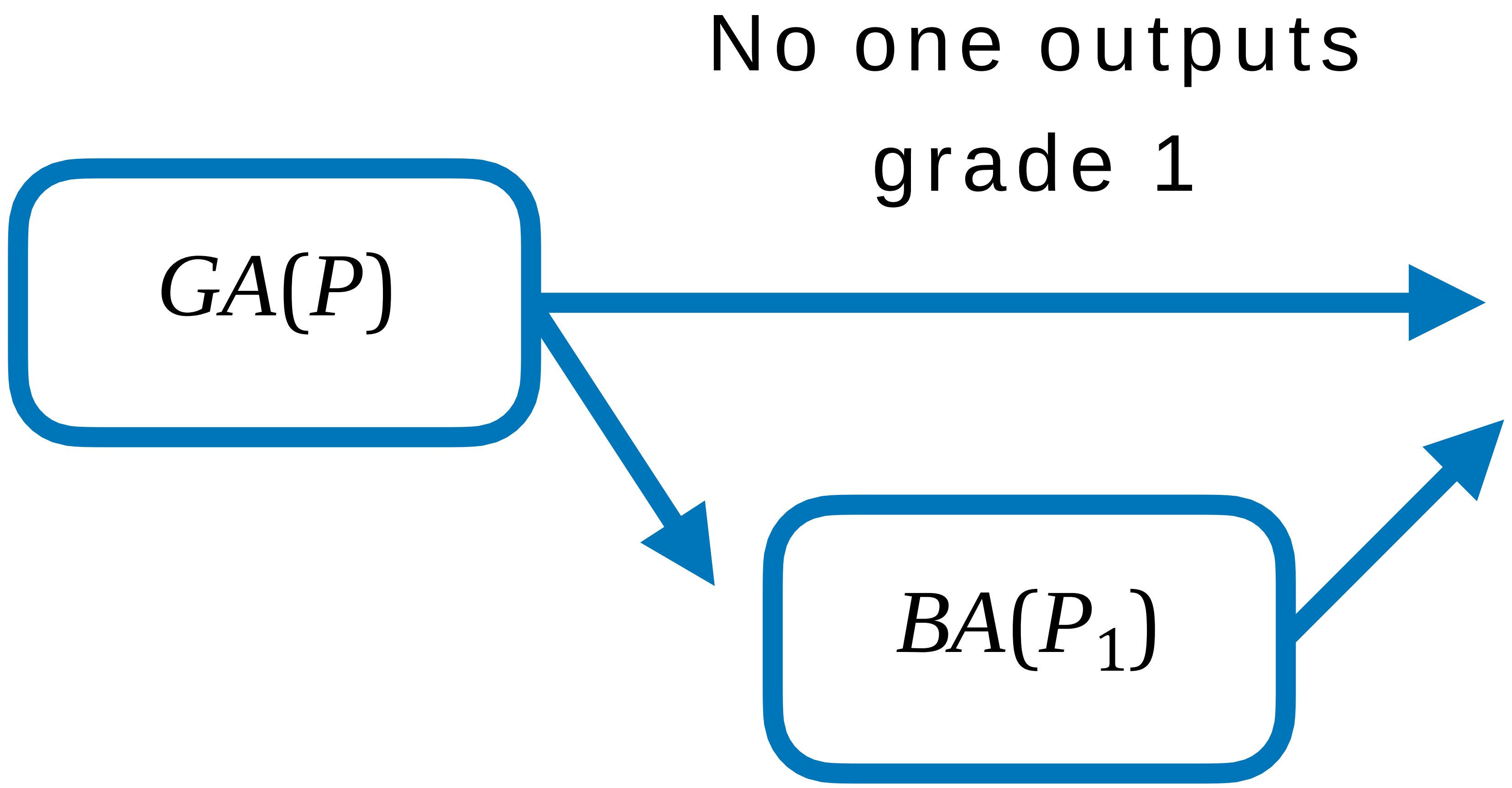
Case 1: Someone outputs a value with grade 1 in GA.



Idea 2: The “correct” step drives agreement

All honest parties agree on a value at the end of the “correct” step.

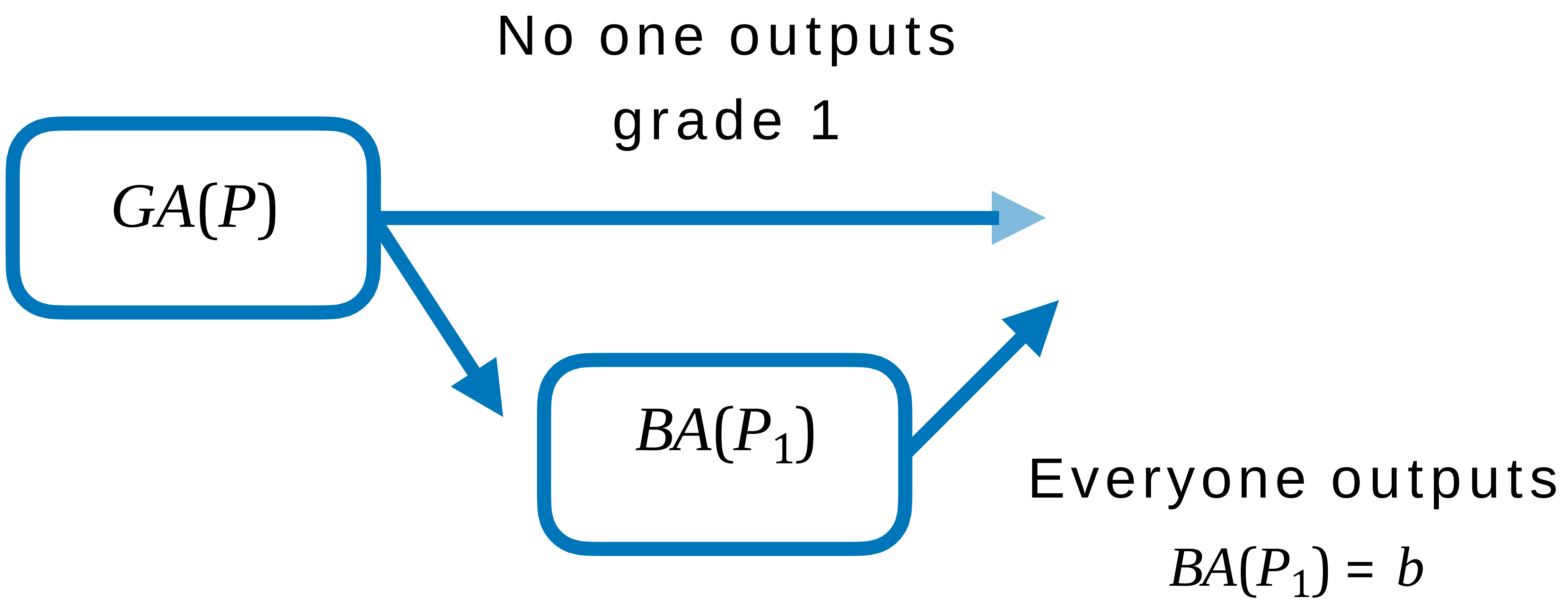
Case 2: No one outputs grade 1 in GA.



Idea 2: The “correct” step drives agreement

All honest parties agree on a value at the end of the “correct” step.

Case 2: No one outputs grade 1 in GA.

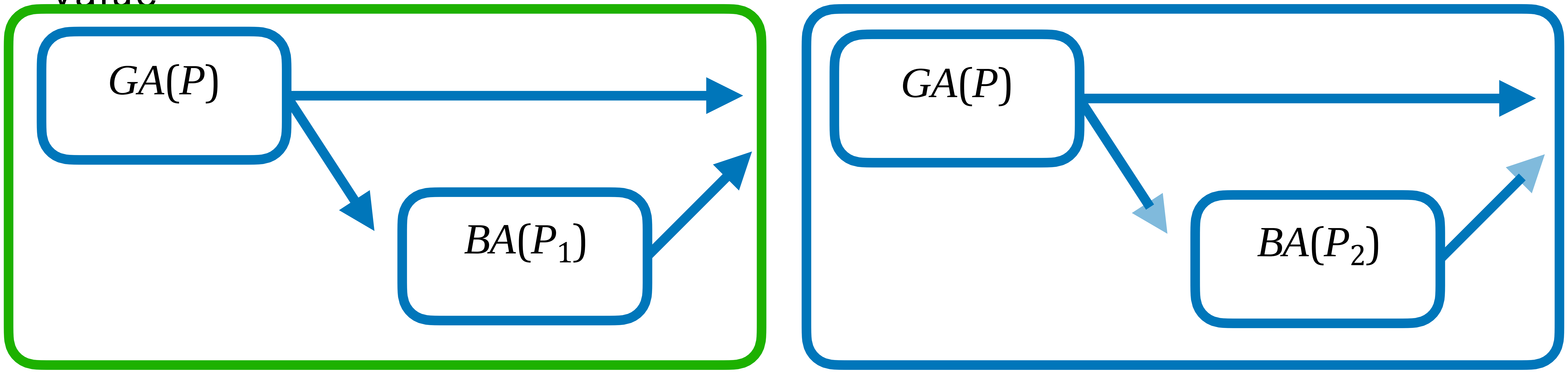


Consistency (case 1: step 1 is correct)

The "correct" first step drives agreement, and the second step does not change the agreed upon value.

Step 1: Everyone agrees on a value

Step 2: Everyone outputs the agreed upon value

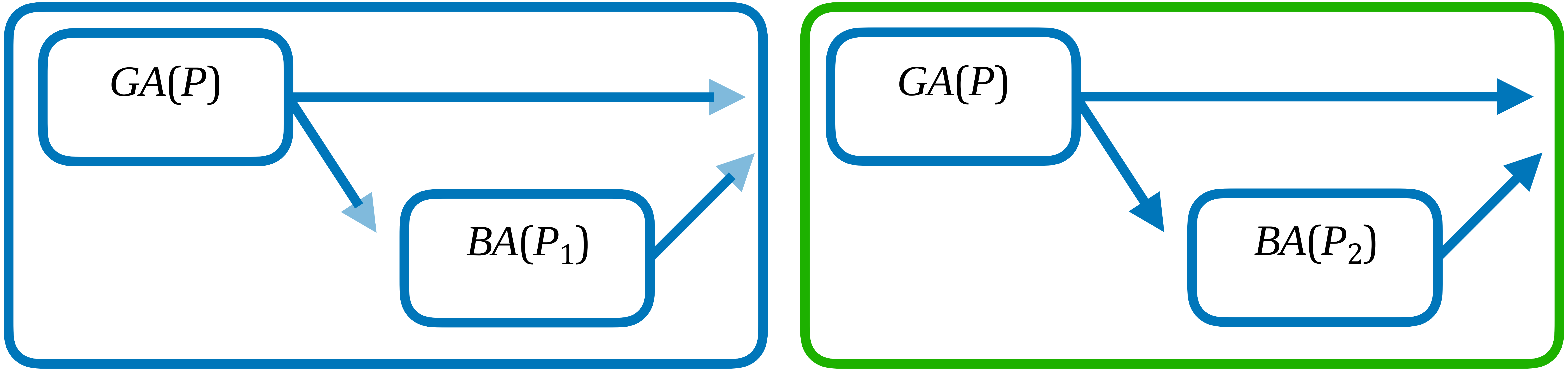


Step 1 is "correct"

Consistency (case 2: step 2 is correct)

All honest parties agree on a value following the “correct” second step.

Step 2: Everyone agrees on a value



Step 2 is “correct”

Communication Complexity

If the GA protocol costs $O(n^2)$ communication, the total communication of the BA protocol will be $O(n^2)$

k - the depth of the recursion

$$C(n) \leq O\left(2^k + \sum_{i=0}^{k} 2^i \cdot \left(\frac{n}{2^i}\right)^2\right) = O(n^2)$$

The end of the recursion has 2^k partitions of $O(1)$ parties

Depth recursion has 2^i GAs with $O((\frac{n}{2^i})^2)$ communication

Outline

1. Achieving BA from GA

- Berman et al's protocol is a problem reduction from BA to GA.

2. Solving GA for $f \geq n/3$

- Solution 1: GA with $f < n/2$ and trusted setup.
- Solution 2: GA with $f \leq (1/2 - \varepsilon)n$ and PKI.

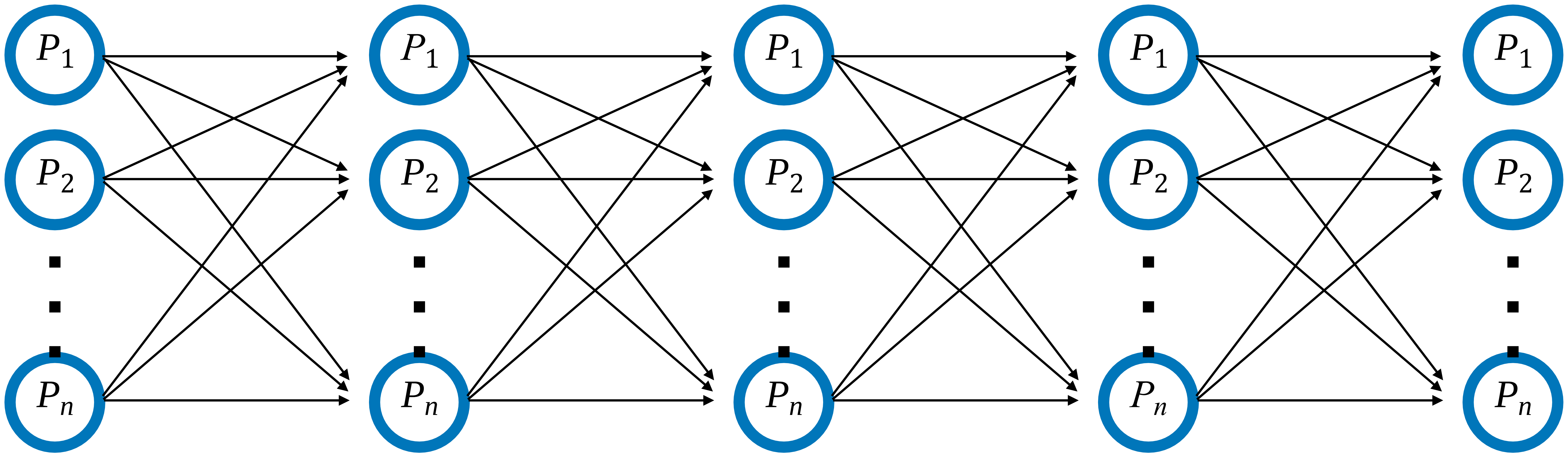
Warmup: GA with $f < n/2$

$C^1(b)$: $n - f$ (vote1, b)

$C^2(b)$: $n - f$ (vote2, b)

2. Forward $C^1(b)$

4. Forward $C^2(b)$



1. Send the input to all as (vote1, b)

3. If no $C^1(b')$ for $b' \neq b$, send (vote2, b) to all

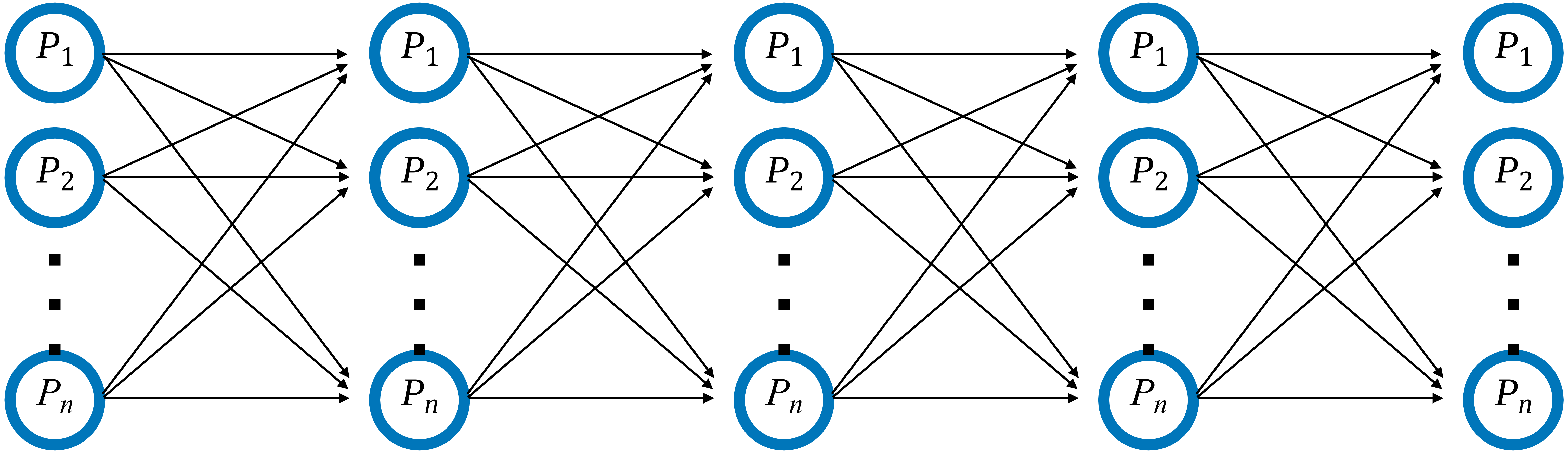
If it receives $C^2(b)$ output

(1) in round 3 $\Rightarrow g \leftarrow 1$

(2) in round 4 $\Rightarrow g \leftarrow 0$

Core idea: Eliminate conflicting majority vote2

Two different majority vote2 $C^2(b)$ and $C^2(b')$ cannot be collected.



2. Forward $C^1(b)$

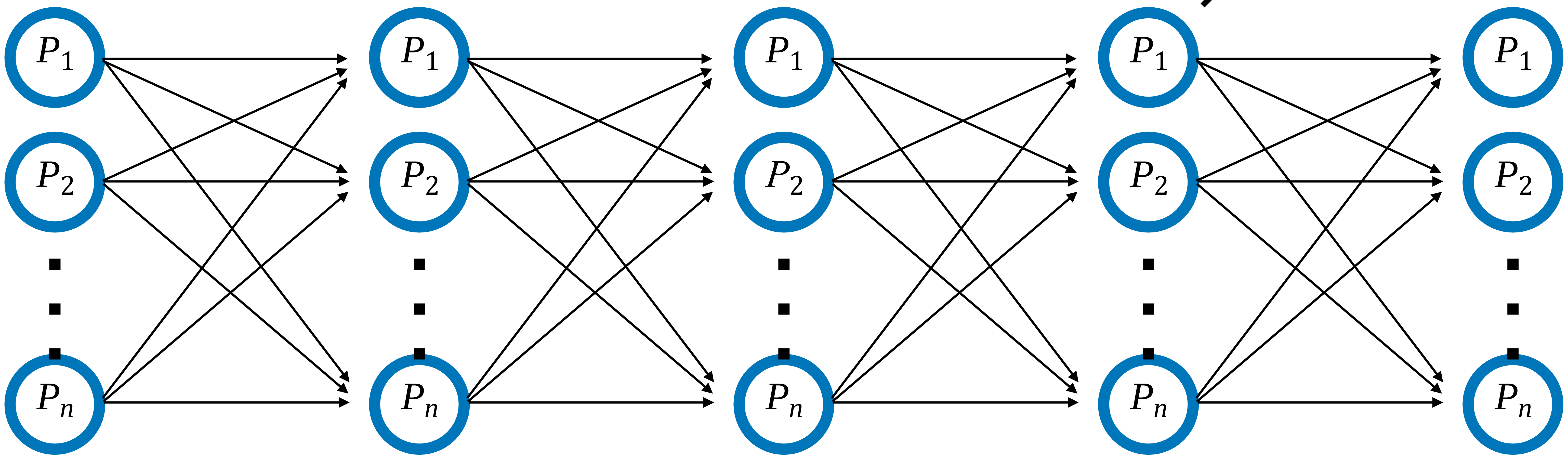
3. If no $C^1(b')$ for $b' \neq b$,

send $(\text{vote2}, b)$ to all

Consistency

If someone outputs $(b,1)$, everyone outputs $(b,*)$.

Forward
$C^2(b)$

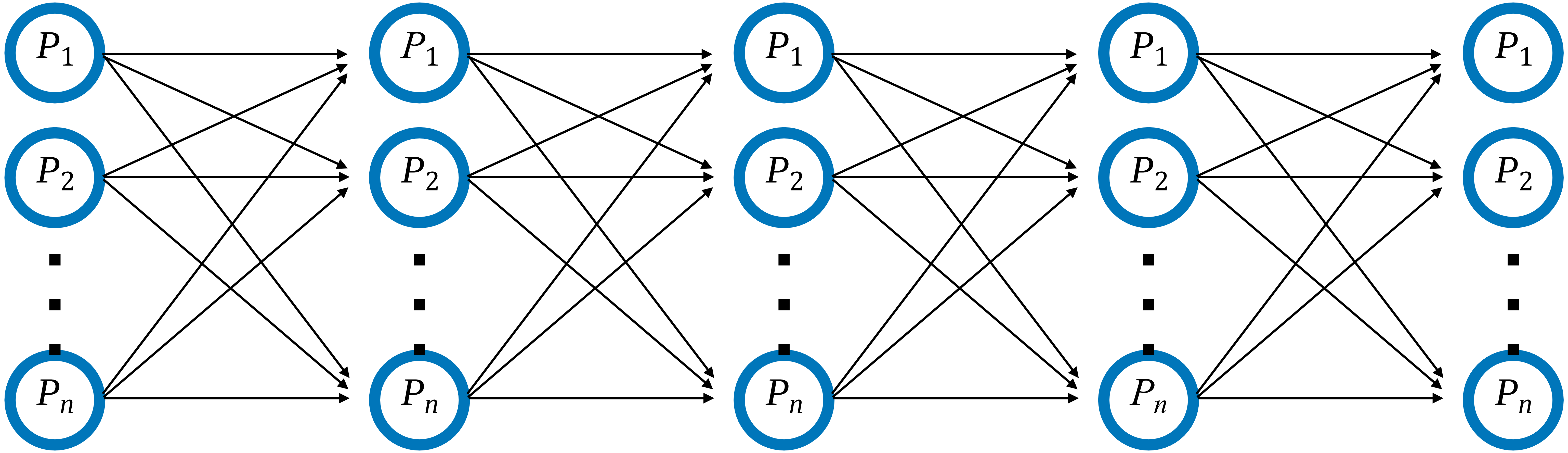


Everyone receives $C^2(b)$

$\Rightarrow$ everyone outputs $(b,*)$

The communication complexity is $\Omega(n^3)$

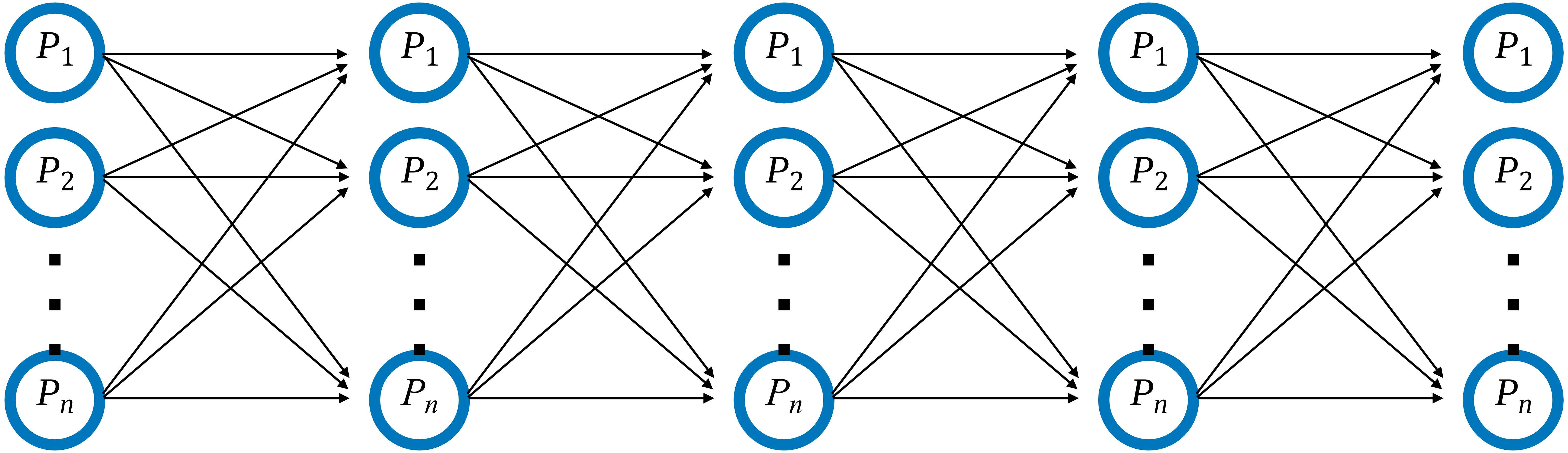
Everyone forwarding $n - f = \Omega(n)$ vote2 costs $\Omega(n^3)$ communication



2. Forward $C^1(b)$

Solution 1: Combining a set of votes

Combining $C^1(b)$ into a single signature using $(n - f, n)$-threshold signature



2. Combine $C^1(b)$ into a single signature, and forward it

Threshold signatures require strong trusted key setup assumption

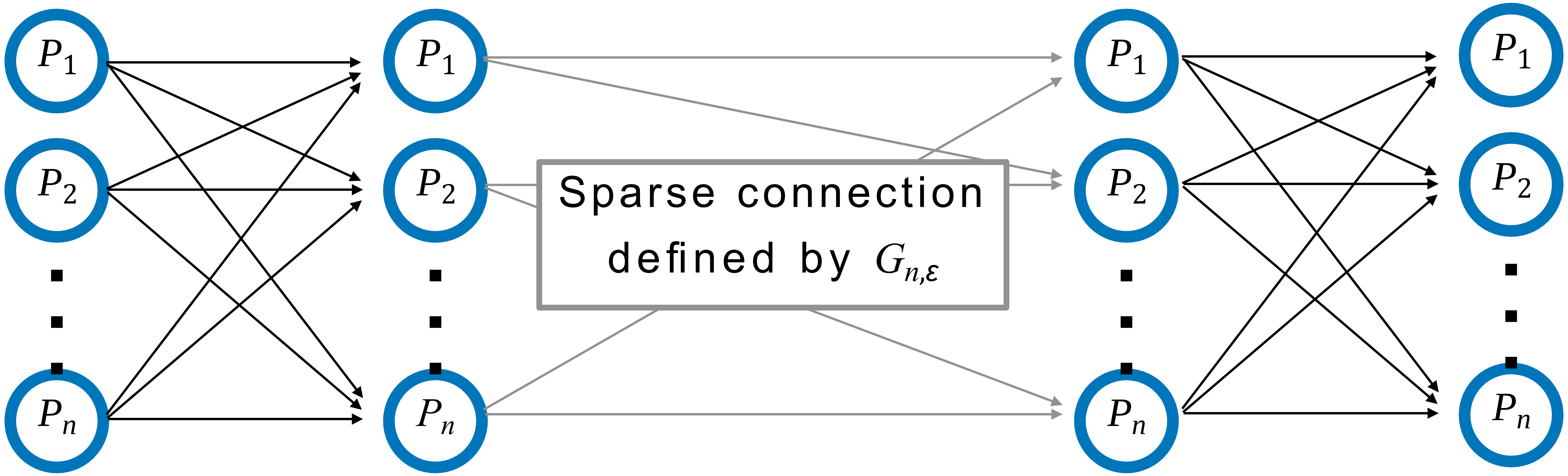
Solution 2: Expander

(n, α, β)-expander. $(0 < \alpha, \beta < 1)$

- A graph of nodes with good connectivety.
- Expansion property. For any subset of nodes, the neighbors $\Gamma(S)$ contains more than βn nodes.
- For any $0 < \alpha, \beta < 1$, a constant degree (n, α, β)-expander exists.
- We use $(n, 2\varepsilon, 1 - 2\varepsilon)$-expander denoted $G_{n,\varepsilon}$

Solution 2: GA with $f \le (1/2 - \varepsilon)n$

1. Send the input to all as (vote1, b)
2. Send $C^1(b)$ to the neighbors in $G_{n,\varepsilon}$
3. If no $C^1(b')$ for $b' \neq b$, send $(\text{vote2}, b)$ to all



The degree of $G_{n,\varepsilon}$ is $O(1)$

→ parties can forward $\Omega(n)$-sized $C^1(b)$ with $O(n^2)$ total communication

Solution 2: GA with $f \leq (1/2 - \varepsilon)n$

Suppose $C^2(b)$ is collected.

→ At least $n - f \geq f + 2\epsilon n$ parties, i.e., 2ε honest parties, must have sent vote2 on the value , who must have propagated $C^1(b)$ to the neighbors in $G_{n,\varepsilon}$

→ More than $(1 - 2\varepsilon)n \geq 2f$ parties, i.e., $> f$ honest parties, must have received $C^1(b)$, who could not have sent vote2 on $b' \neq b$

→ $C^2(b')$ cannot be collected.

Summary

- Solution 1 achieves $f < n/2$, but requires trusted key setup for threshold signatures.
- Solution 2 tolerate $f \le (1/2 - \varepsilon)n$, but requires only PKI.

<table>
<tr><td>authenticated
(trusted setup)</td><td>$f < n/2$</td><td rowspan="2">$\Omega(n^2)$
[Dolev-Resichuk]</td><td>$O(n^2)$
this work</td></tr>
<tr><td>authenticated
(PKI)</td><td>$f < (1/2 - \varepsilon)n$</td><td>$O(n^2)$
this work</td></tr>
</table>