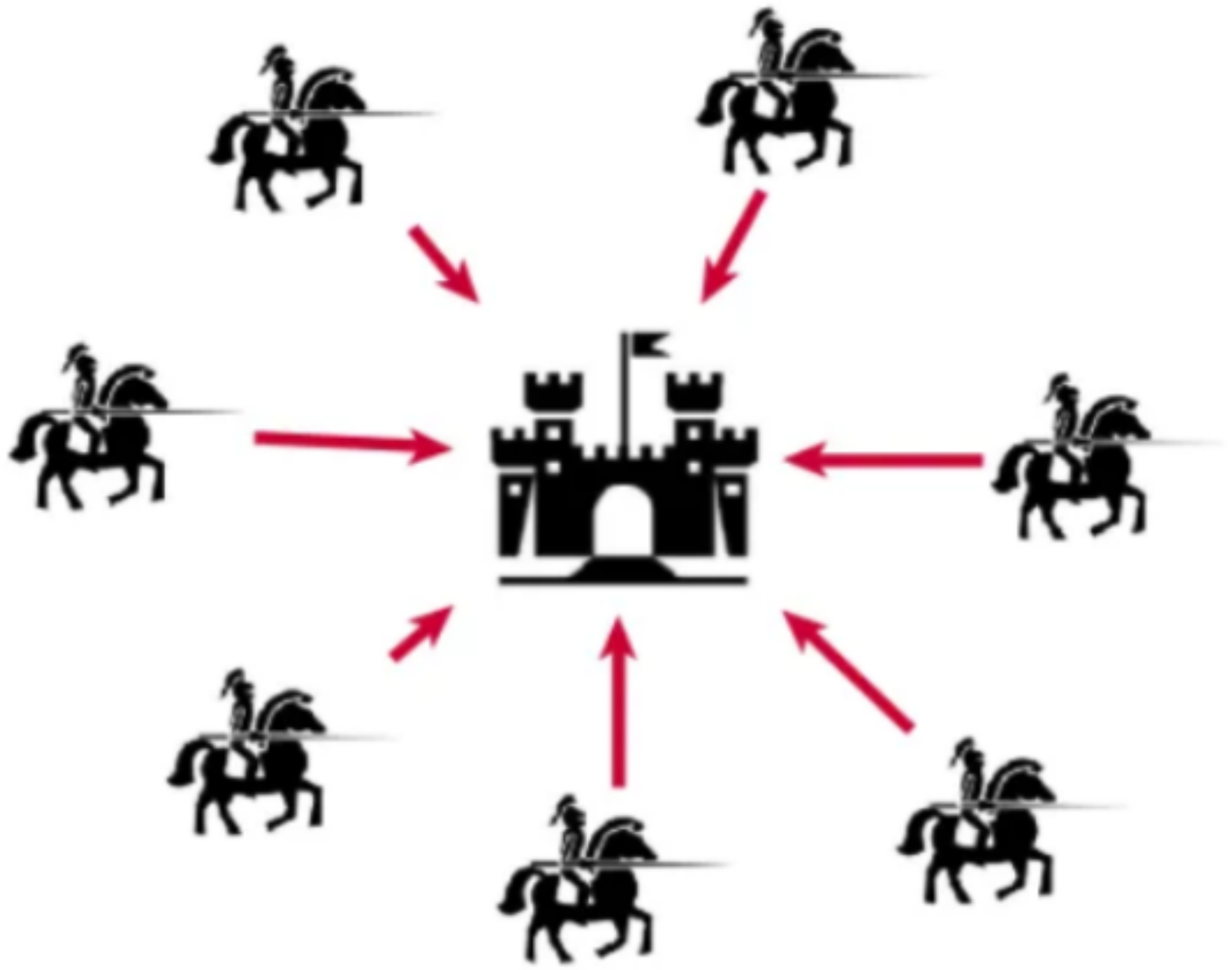
Early Stopping Byzantine Agreement in $(1 + \epsilon) \cdot f$ Rounds

Fatima Elsheimy[1], Julian Loss[2], Charalampos Papamanthou[1]

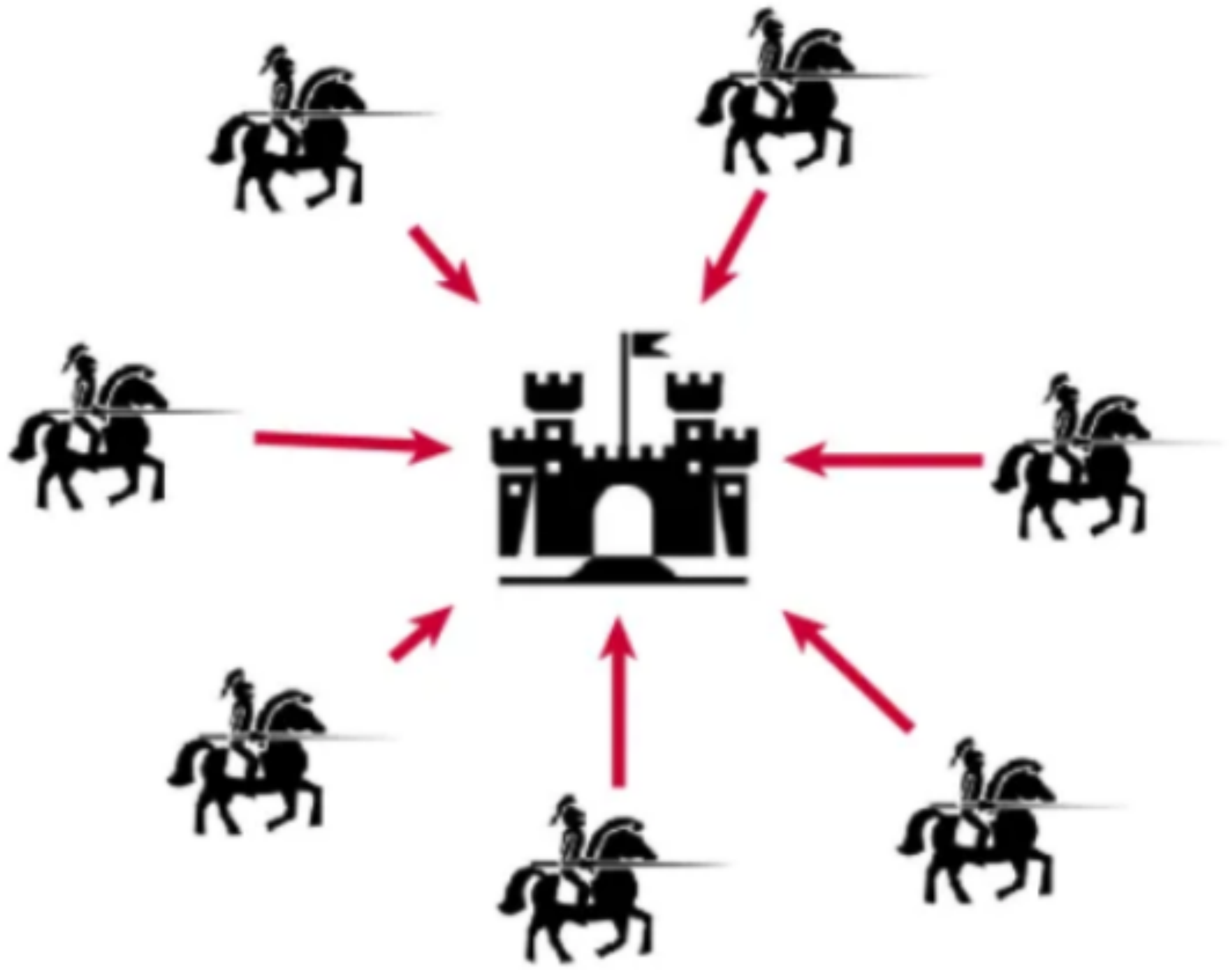
1. Yale University

2. CISPA Helmholtz Center for Information Security

Byzantine Agreement

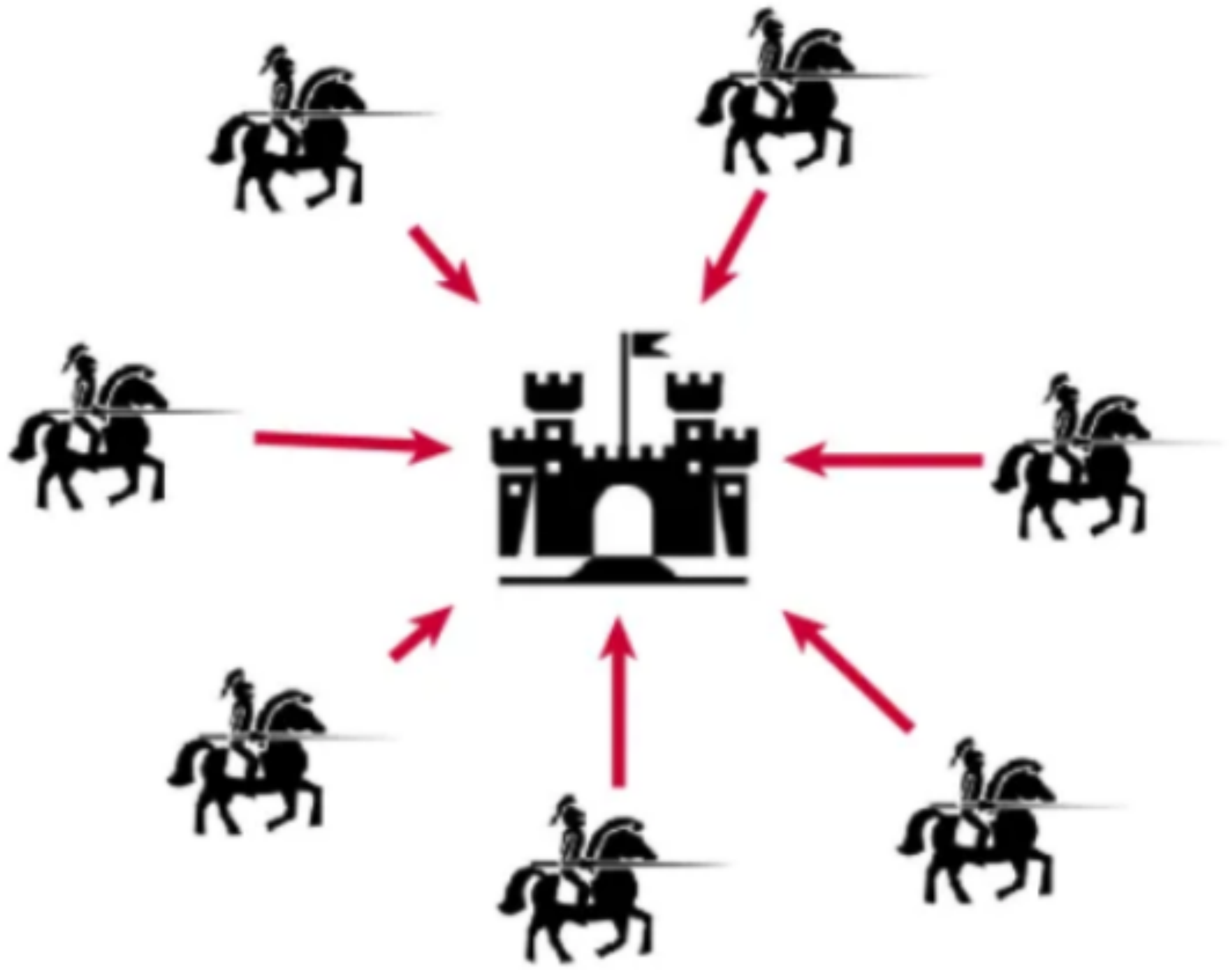


Byzantine Agreement



Validity: If every honest party P_i inputs $v_i = v,$ then all honest parties output $y_i = v$.

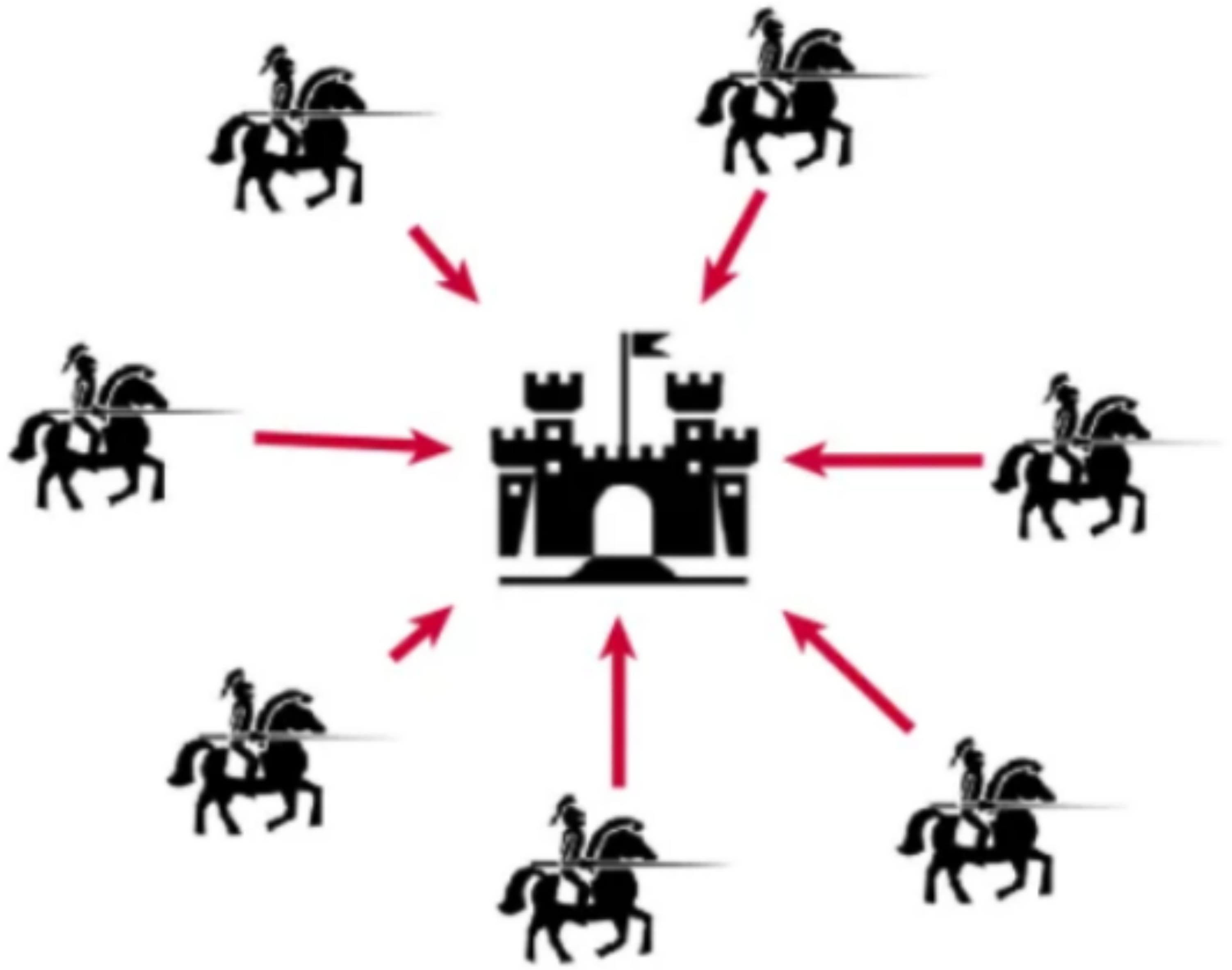
Byzantine Agreement



Validity: If every honest party P_i inputs $v_i = v,$ then all honest parties output $y_i = v$.

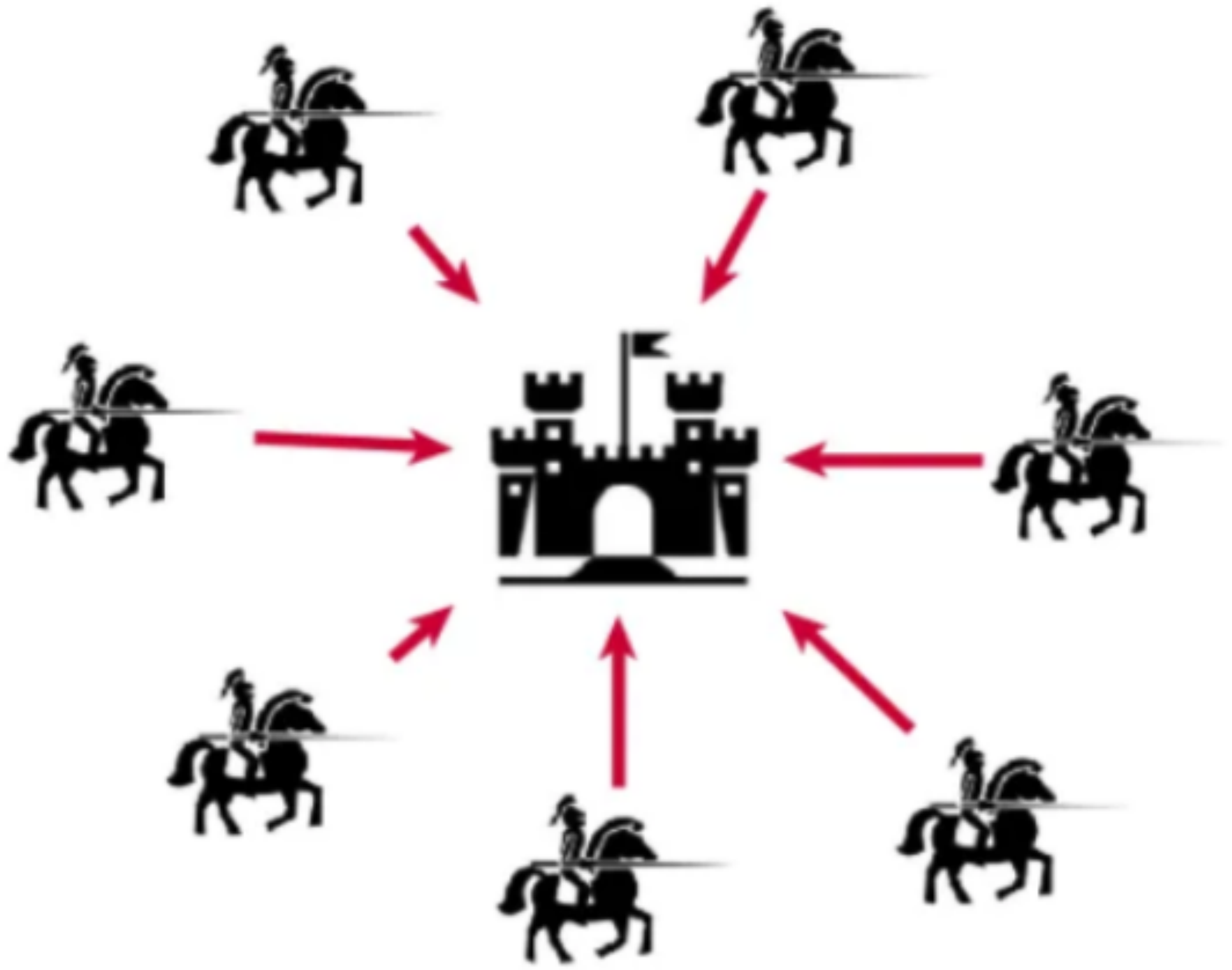
Agreement: All honest parties output the same value v.

Byzantine Agreement



System Model:

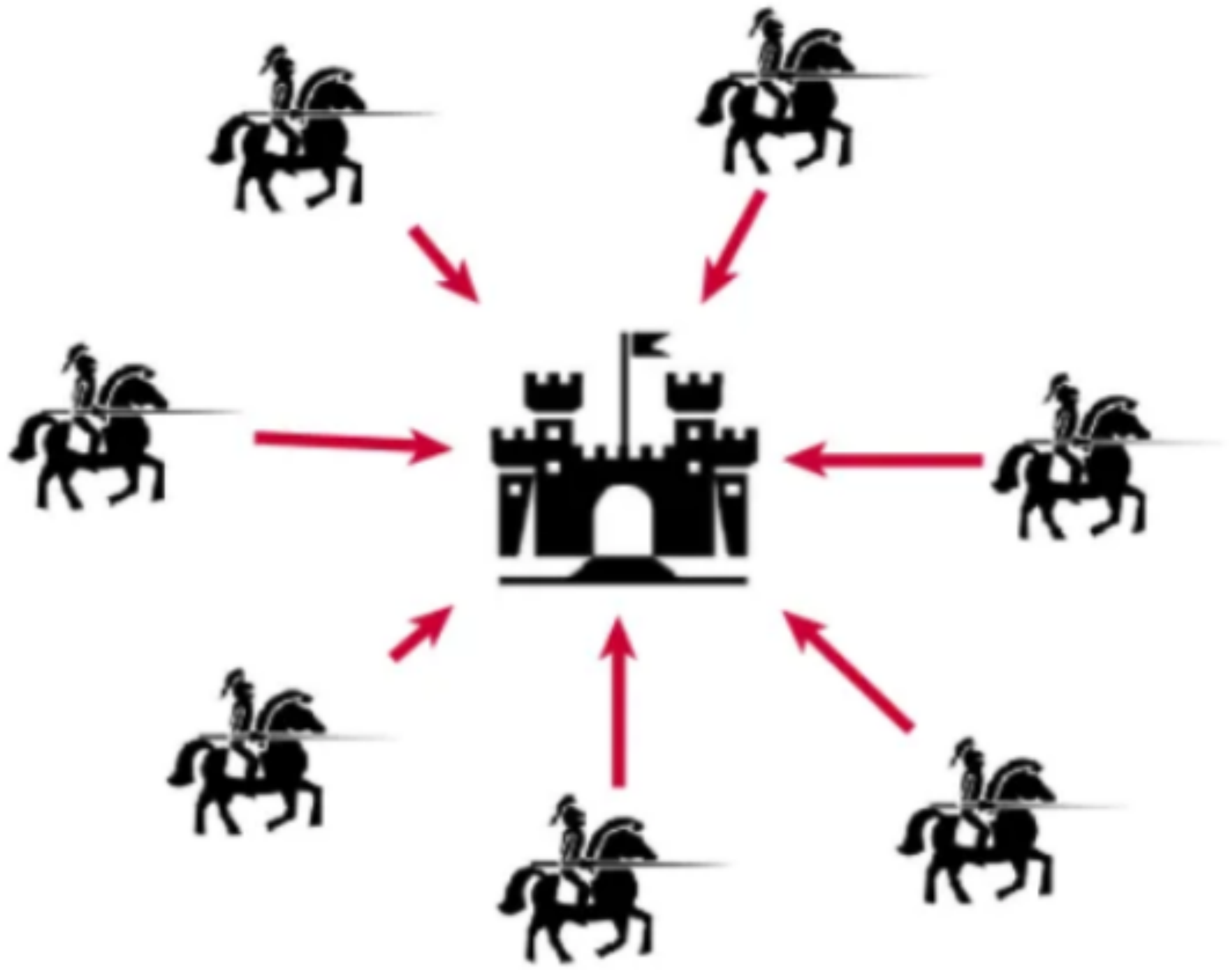
Byzantine Agreement



System Model:

- Synchronous Setting

Byzantine Agreement



System Model:

- Synchronous Setting
- Adversary can corrupt up to $t < n/2$ parties.

Round Efficiency

Round Complexity Lower Bound [$DRS90$]: Byzantine Agreement terminates in $min(f+2, t+1)$

Round Efficiency

Exact number of corruption $f < t$

Round Complexity Lower Bound [$DRS90$]: Byzantine Agreement terminates in $min(f + 2, t + 1)$

Round Efficiency

Exact number of corruption $f < t$

Round Complexity Lower Bound [$DRS90$]: Byzantine Agreement terminates in $min(f + 2, t + 1)$

Related Work:

	Resilience	Round Complexity
[NL24]	$t < n$	$O(min(f^2, t))$
[AD15]	$t < n/3$	$min(f + 2, t + 1)$
[PT88]	$t < n/2$	$min(2f + 4, 2t + 2)$
This Work: Π_{BA^d}	$t < n/2$	$(1 + \epsilon) \cdot f$

	Resilience	Expected Complexity	Worst-Case Complexity
[GP90]	$t < n/3$	$O(1)$	$t + log(t)$
This Work: Π_{BA^r}	$t < n/2$	$O(1)$	$(1 + \epsilon) \cdot f$

Round Efficiency

Exact number of corruption $f < t$

Round Complexity Lower Bound $[DRS90]$: Byzantine Agreement terminates in $min(f + 2, t + 1)$

Related Work:

	Resilience	**Round Complexity**
[NL24]	$t < n$	$O(min(f^2, t))$
[AD15]	$t < n/3$	$min(f + 2, t + 1)$
[PT88]	$t < n/2$	$min(2f + 4, 2t + 2)$
This Work: Π_{BA^d}	$t < n/2$	$(1 + \epsilon) \cdot f$

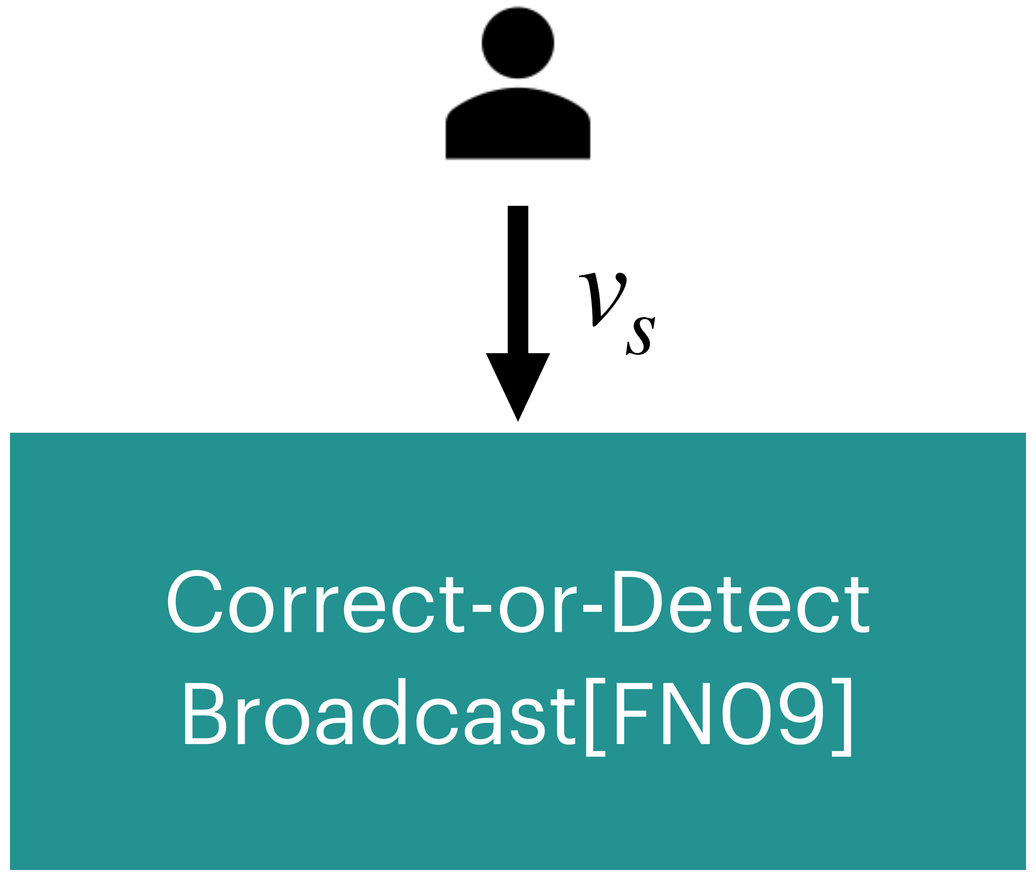
	Resilience	**Expected Complexity**	**Worst-Case Complexity**
[GP90]	$t < n/3$	$O(1)$	$t + log(t)$
This Work: Π_{BA^r}	$t < n/2$	$O(1)$	$(1 + \epsilon) \cdot f$

Construction of Π_{BA^d}

Building Block 1: Detect Malicious Parties

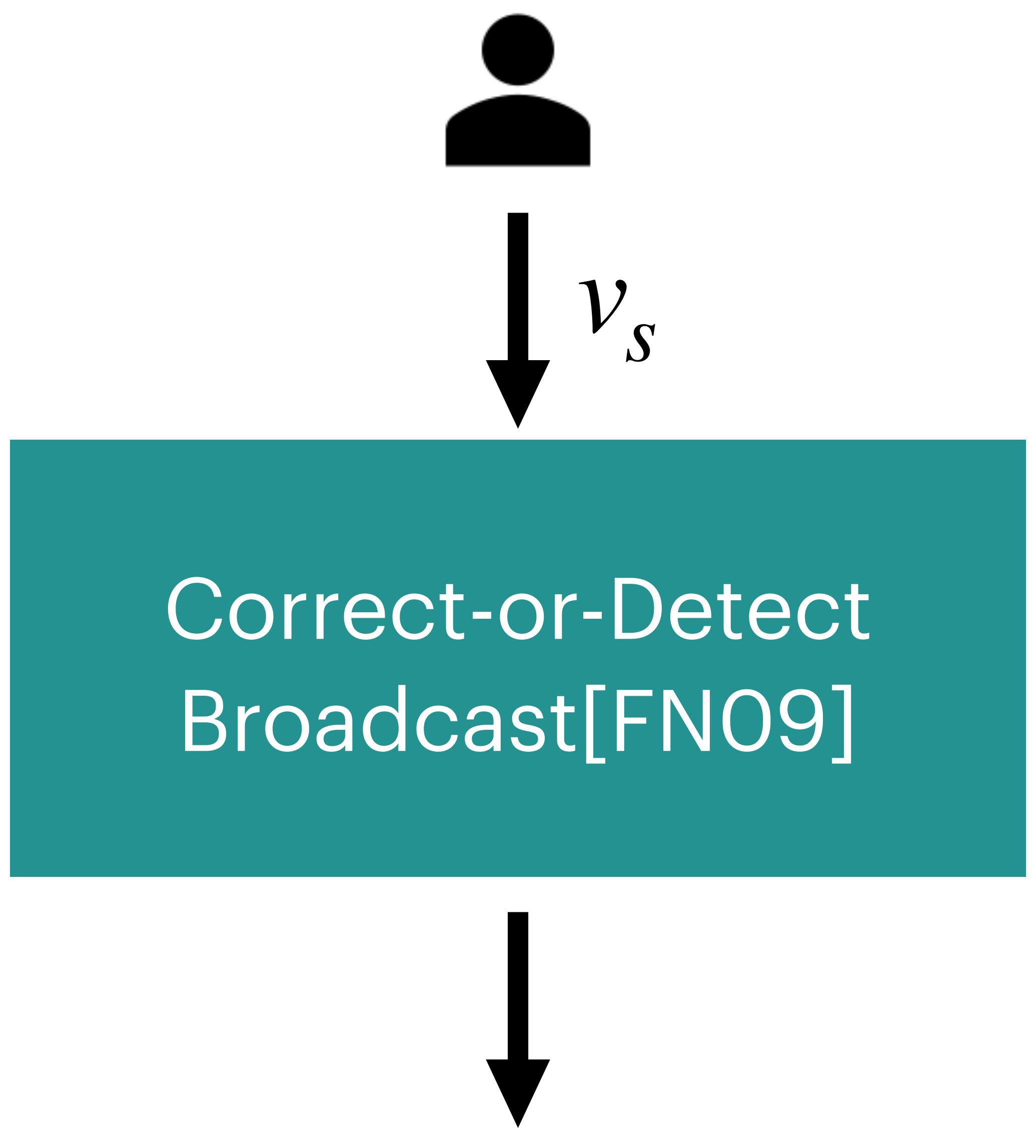
Correct-or-Detect Broadcast[FN09]

Building Block 1: Detect Malicious Parties



Correct-or-Detect Broadcast[FN09]

Building Block 1: Detect Malicious Parties

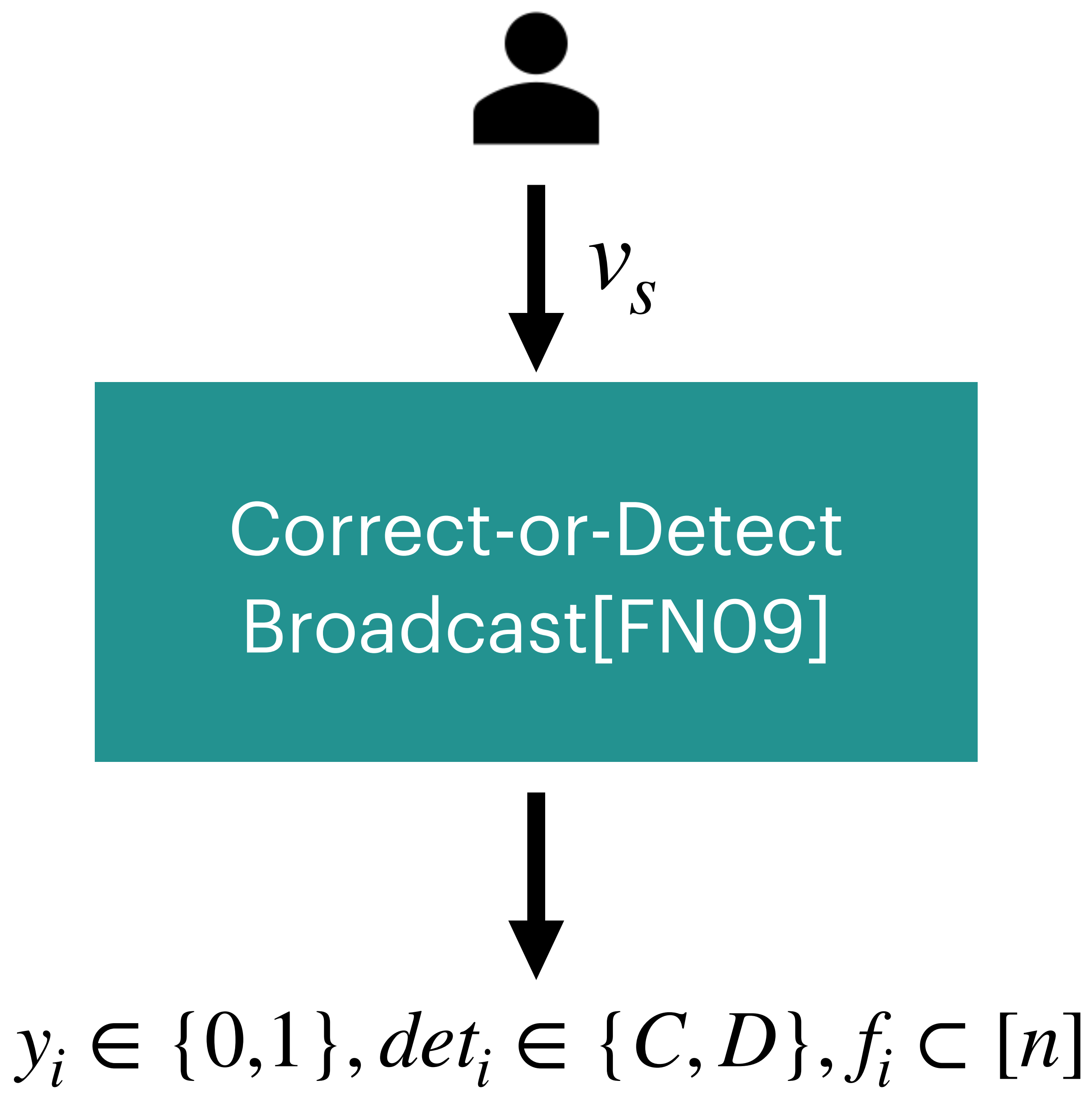


v_s

Correct-or-Detect Broadcast[FN09]

$y_i \in \{0,1\}, det_i \in \{C, D\}, f_i \subset [n]$

Building Block 1: Detect Malicious Parties



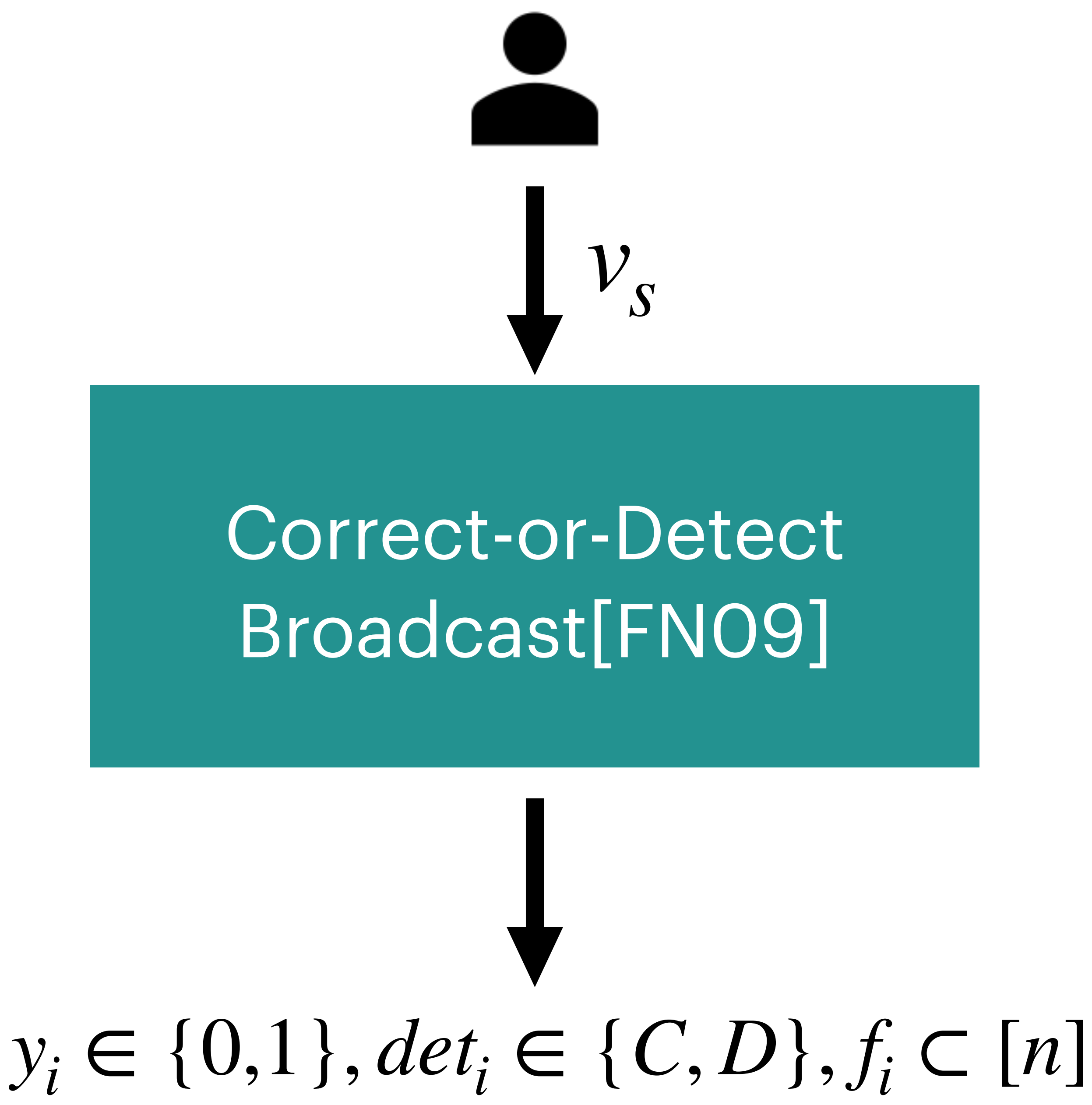
v_s

Correct-or-Detect Broadcast[FN09]

$y_i \in \{0,1\}, det_i \in \{C, D\}, f_i \subset [n]$

- Runs for $d + 4$ rounds

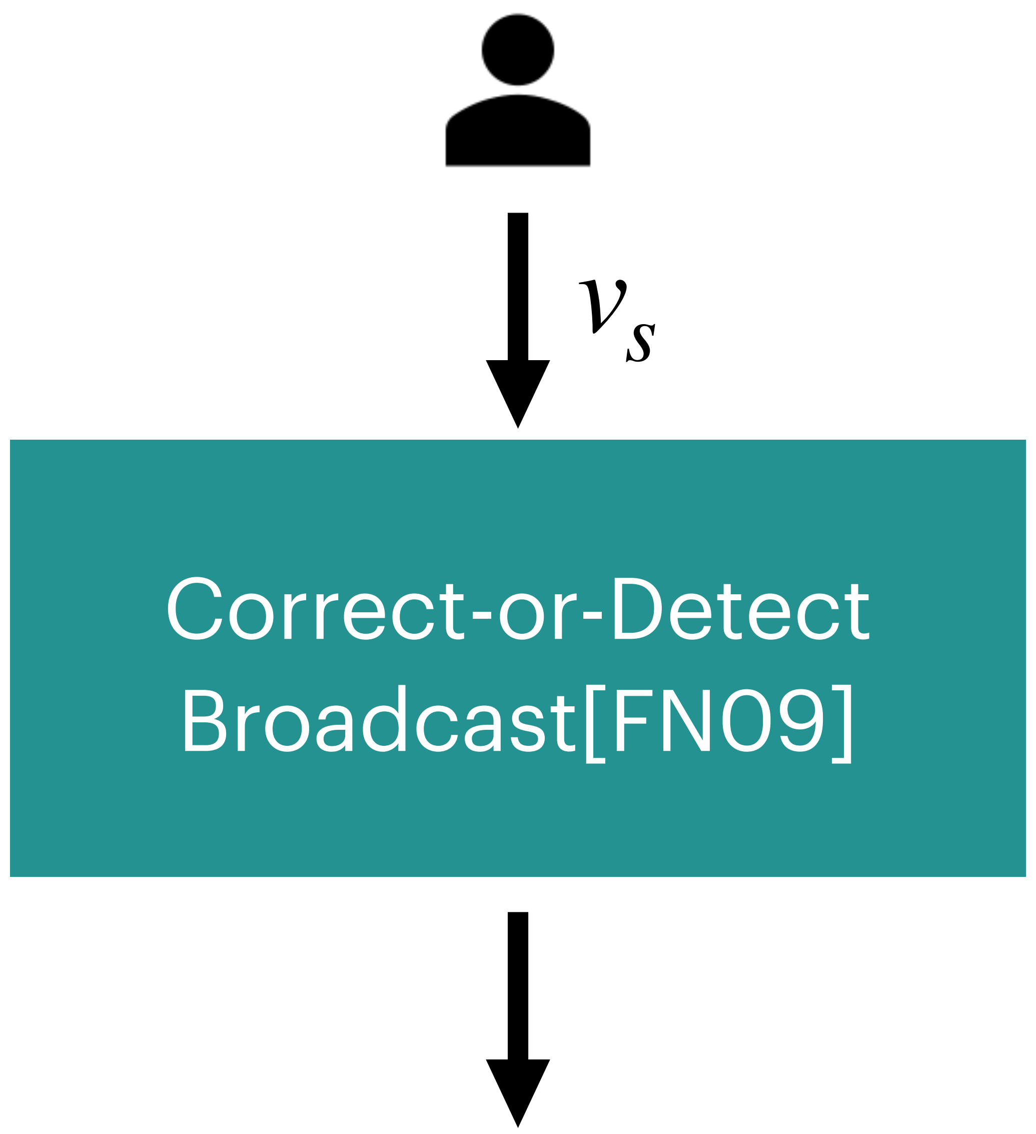
Building Block 1: Detect Malicious Parties


v_s
Correct-or-Detect Broadcast[FN09]
$y_i \in \{0,1\}, det_i \in \{C, D\}, f_i \subset [n]$

Constant

- Runs for $d+4$ rounds

Building Block 1: Detect Malicious Parties



Correct-or-Detect Broadcast[FN09]

$y_i \in \{0,1\}, det_i \in \{C, D\}, f_i \subset [n]$

Constant

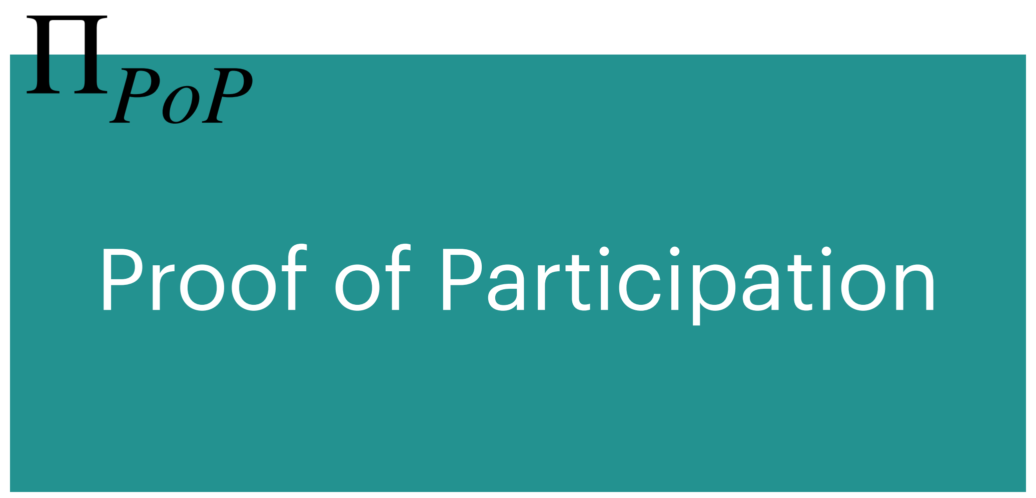
- Runs for $d + 4$ rounds
- Parties either have agreement (*Correct*) or identify d malicious parties (*Detect*)

Building Block 2: Restrict Detected Parties

Π_{PoP}

Proof of Participation

Building Block 2: Restrict Detected Parties

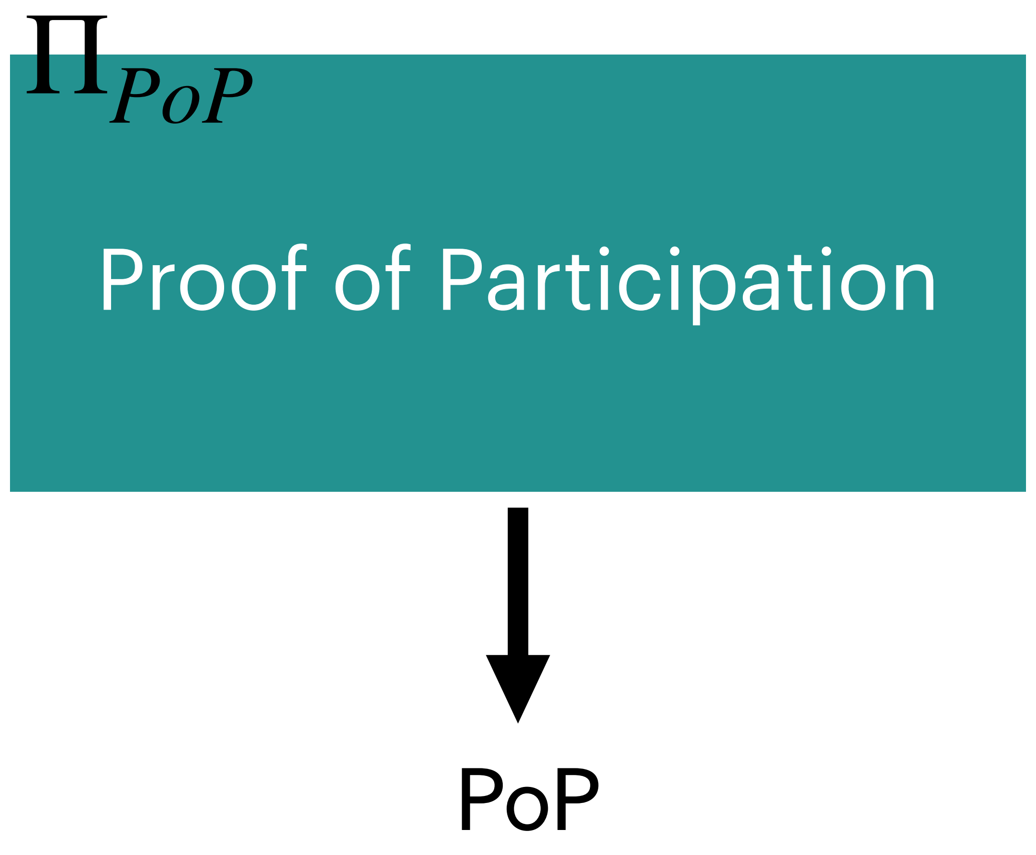


Π_{PoP}

Proof of Participation

- One round: Each party sends an honesty message to party P_j if it is not on its faulty list.

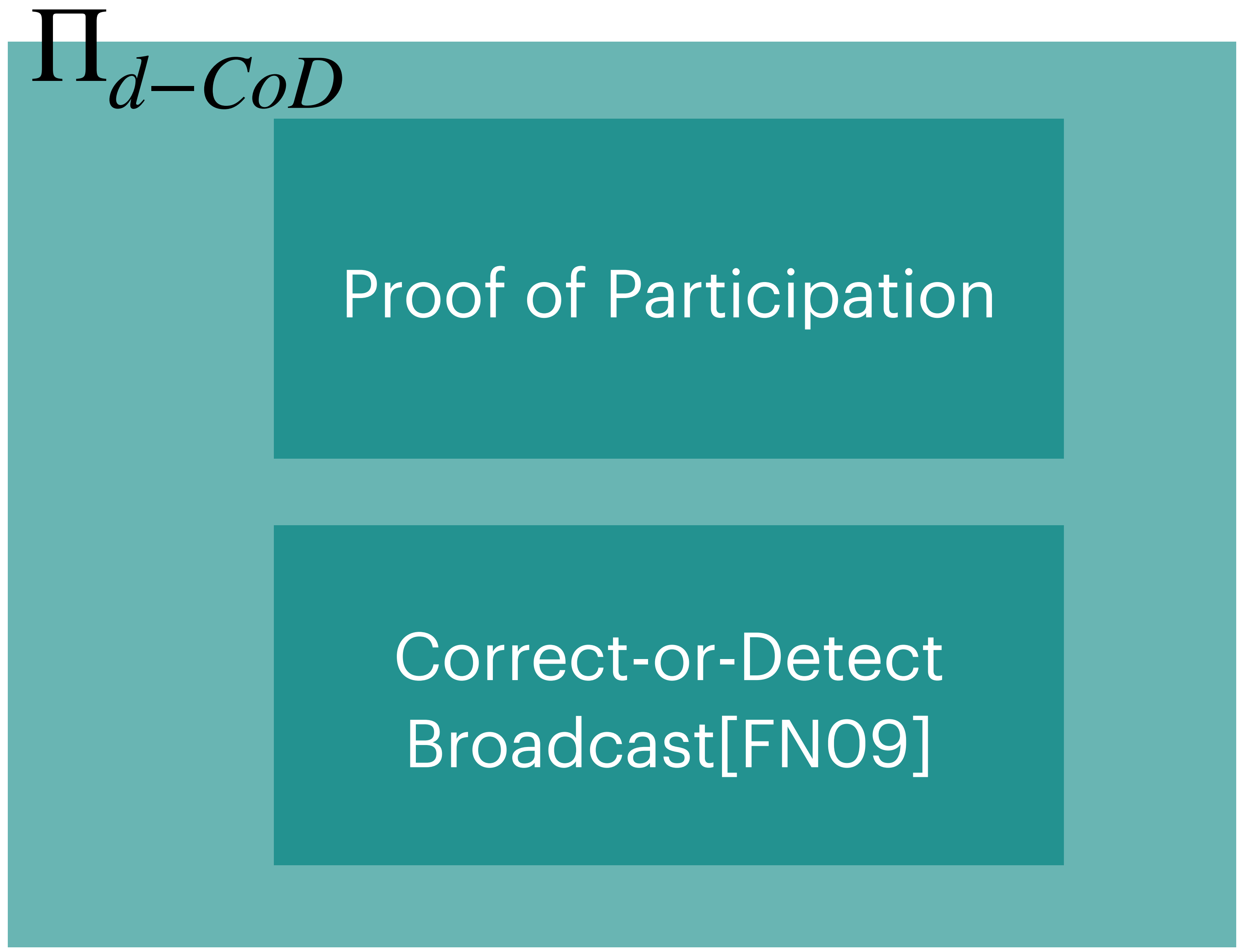
Building Block 2: Restrict Detected Parties



- One round: Each party sends an honesty message to party P_j if it is not on its faulty list.

- Output: Proof of participation(PoP) = accumulation of >n/2 honesty messages received.

Building Block 3: Π_{d-CoD}

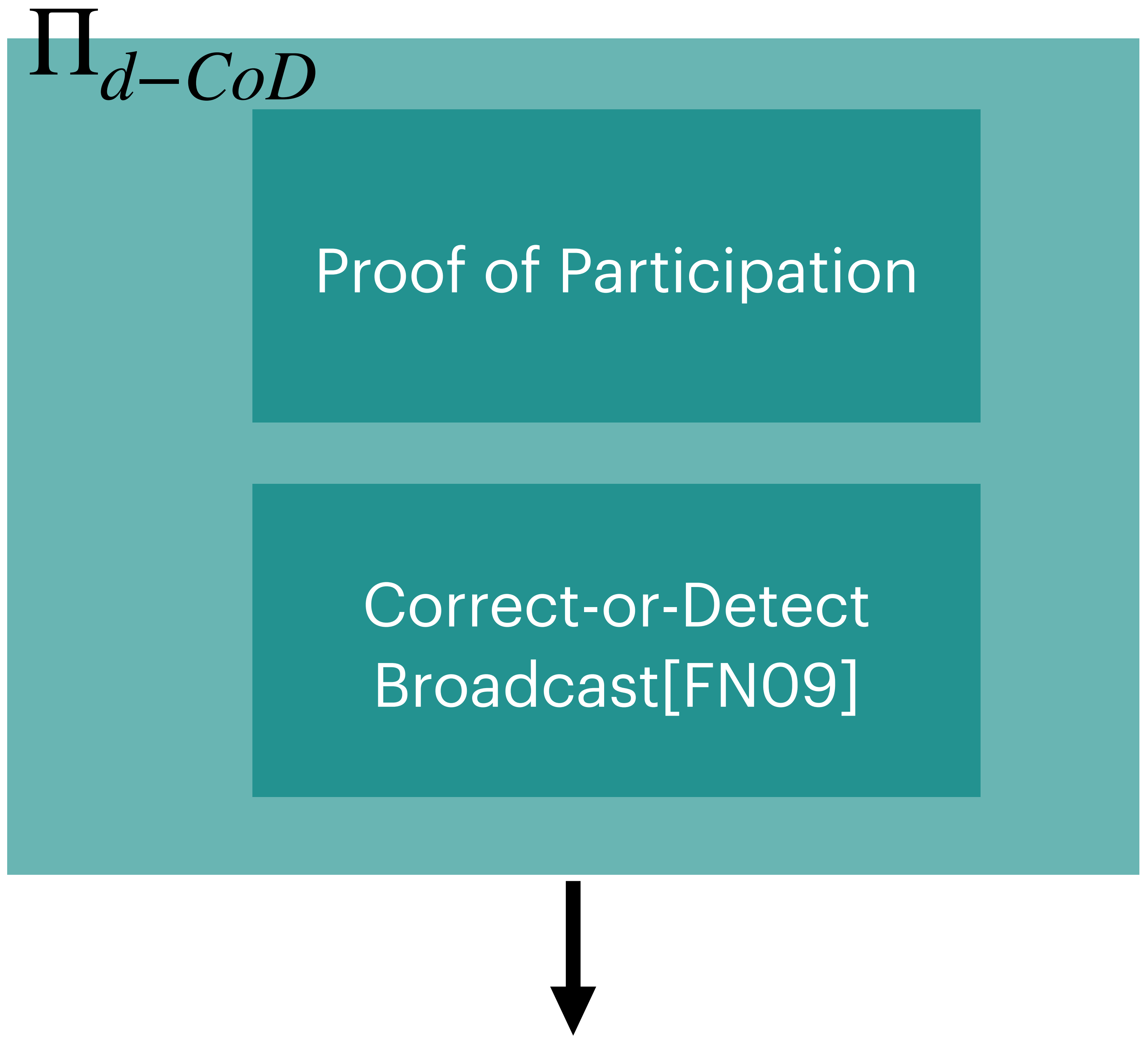


Π_{d-CoD}

Proof of Participation

Correct-or-Detect Broadcast[FN09]

Building Block 3: Π_{d-CoD}



Π_{d-CoD}

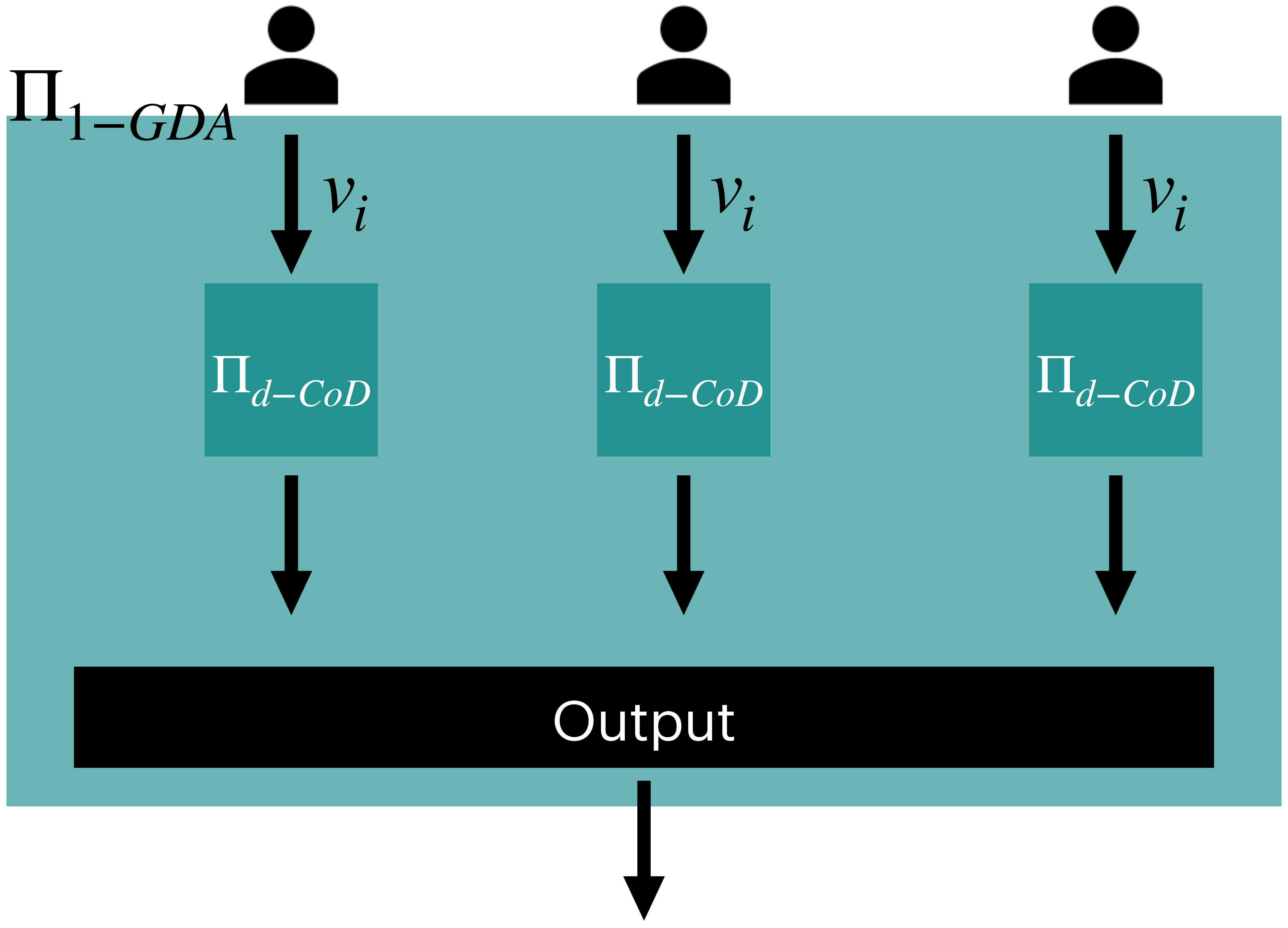
Proof of Participation

Correct-or-Detect Broadcast[FN09]

$y_i \in \{0,1\}, det_i \in \{C, D\}, f_i \subset [n]$

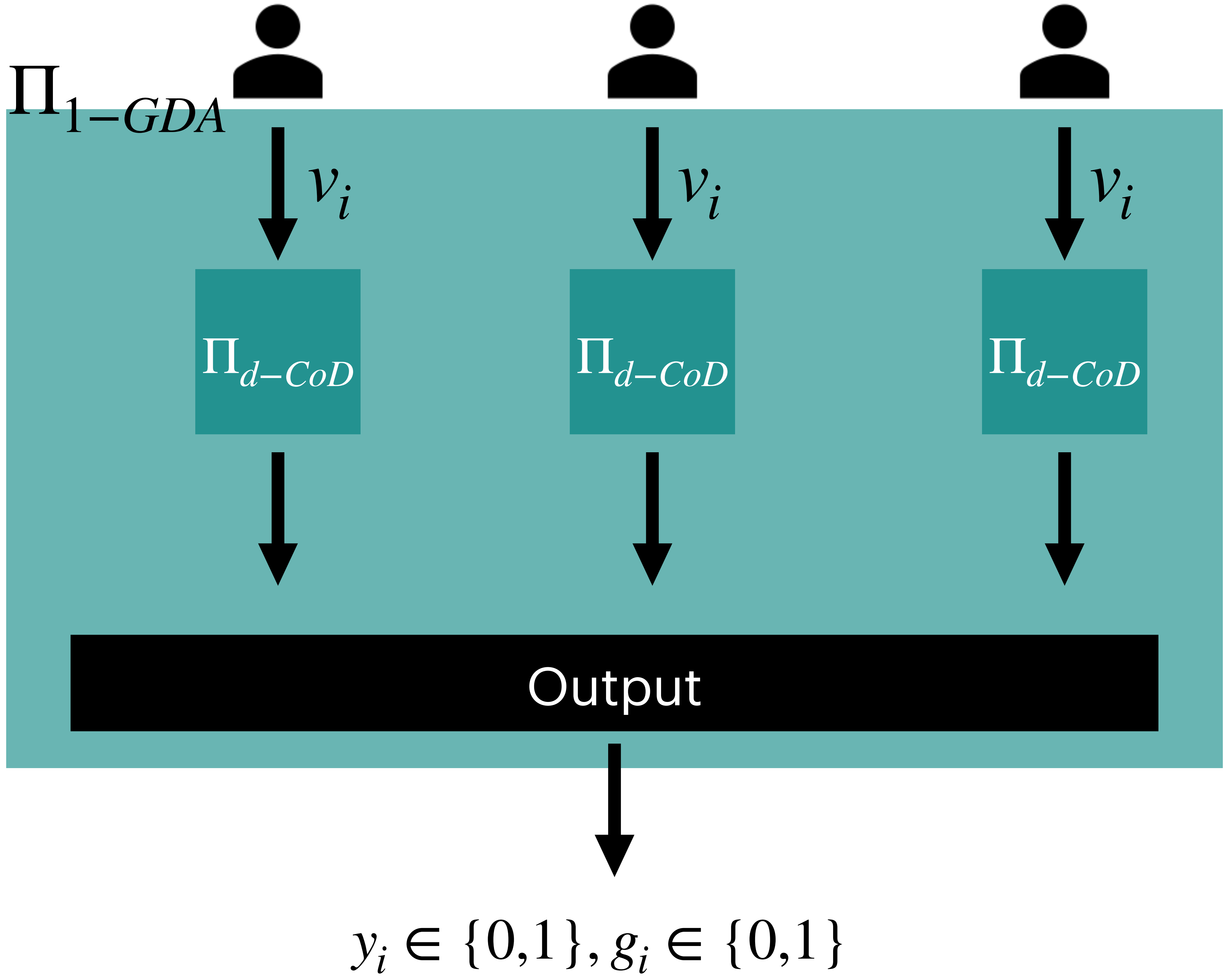
- Same security properties as Correct-or-Detect Broadcast
- Only Parties with valid PoP can participate
- d+5 rounds in total

Building Block 4: 1-Graded *d*-Detecting Agreement



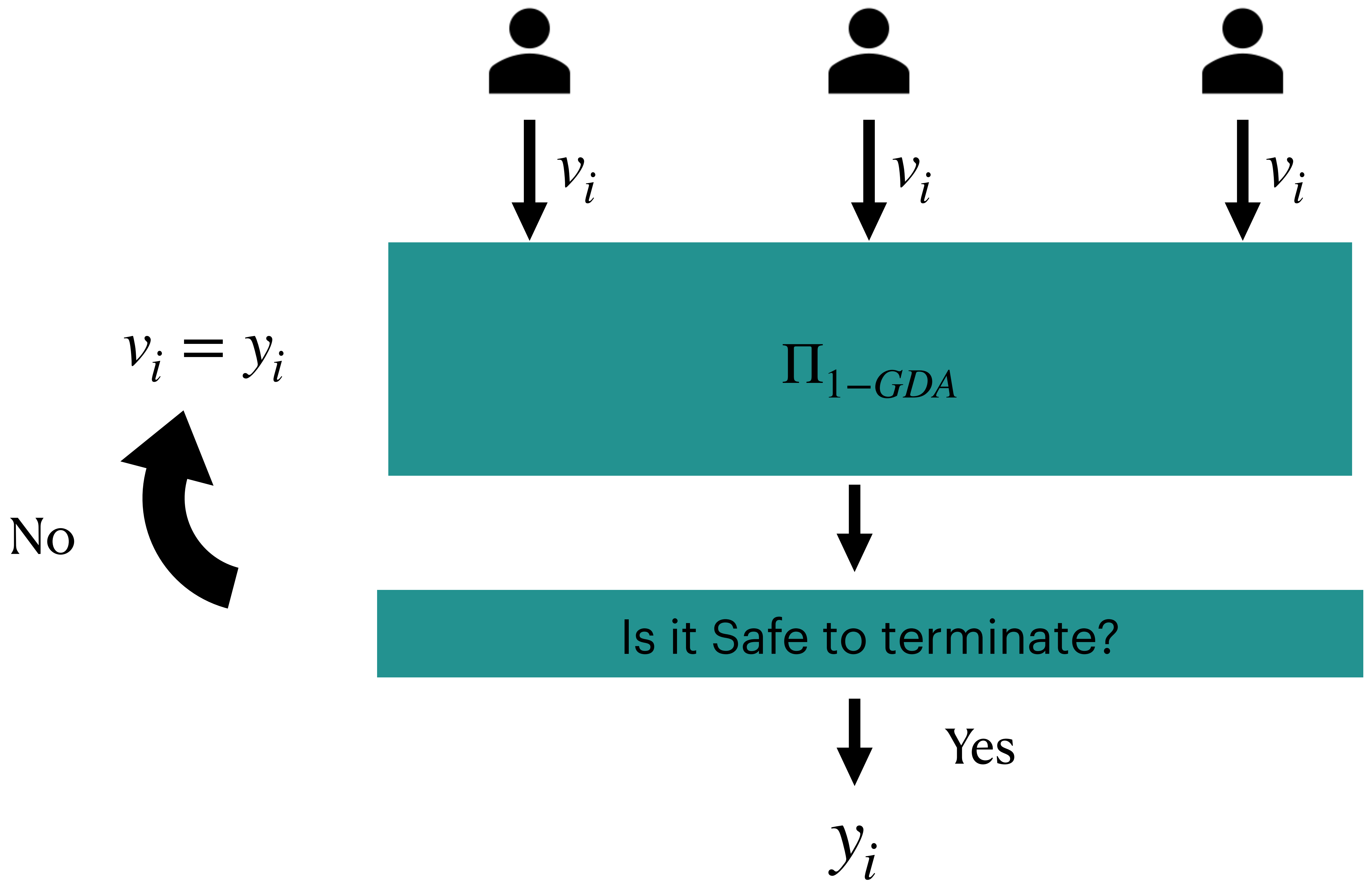
$y_i \in \{0,1\}, g_i \in \{0,1\}$

Building Block 4: 1-Graded *d*-Detecting Agreement

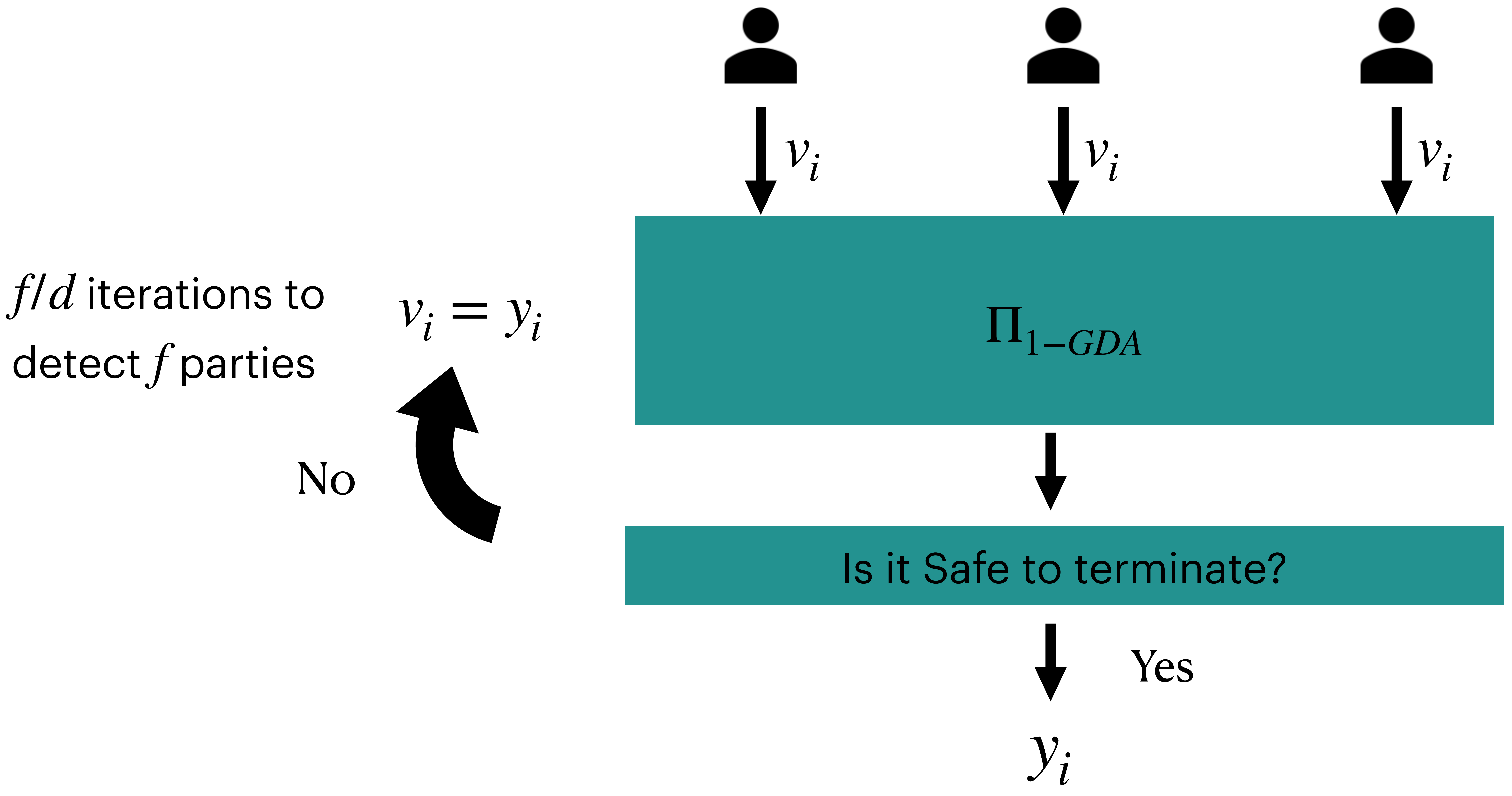


- If all honest parties input the same value v, they output $y_i = v, g_i = 1$
- If an honest party outputs $g_i = 0$, honest parties detect at least d malicious parties
- $d + 5$ round complexity

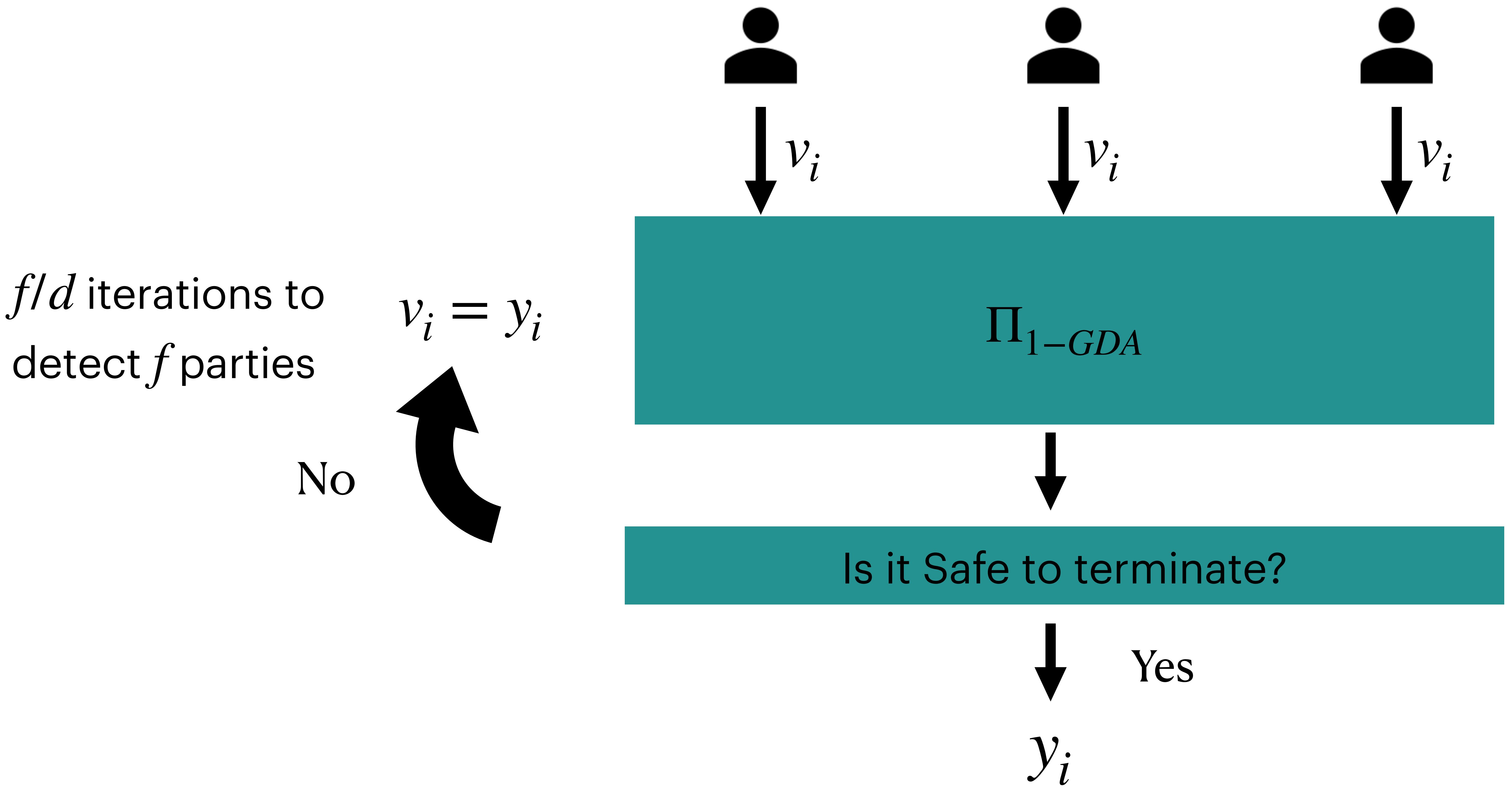
Building Block 5: Byzantine Agreement



Building Block 5: Byzantine Agreement



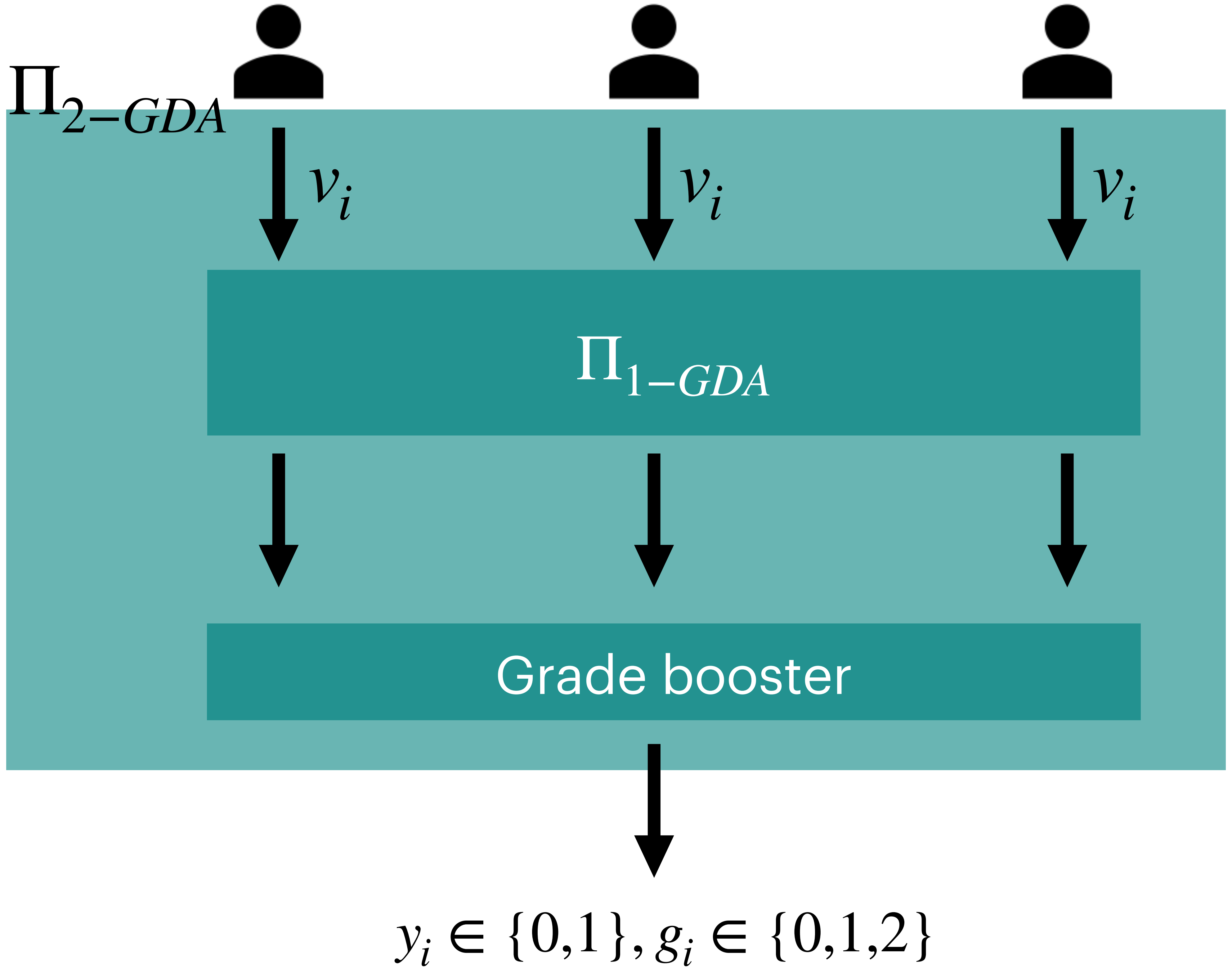
Building Block 5: Byzantine Agreement



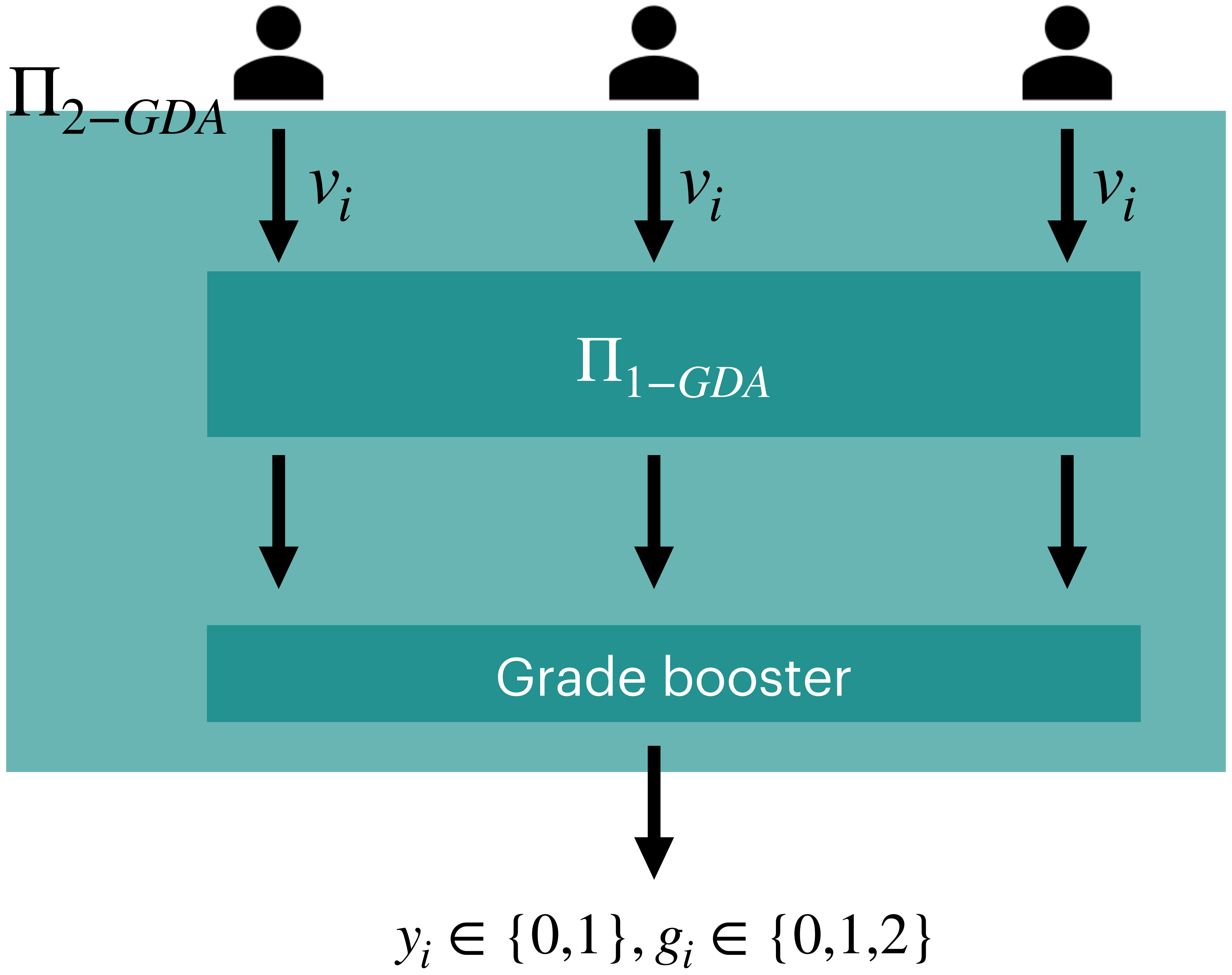
Round Complexity: $d + 5(f/d + 2)$

Construction of Π_{BA^r}

Building Block 1: 2-Graded *d*-Detecting Agreement

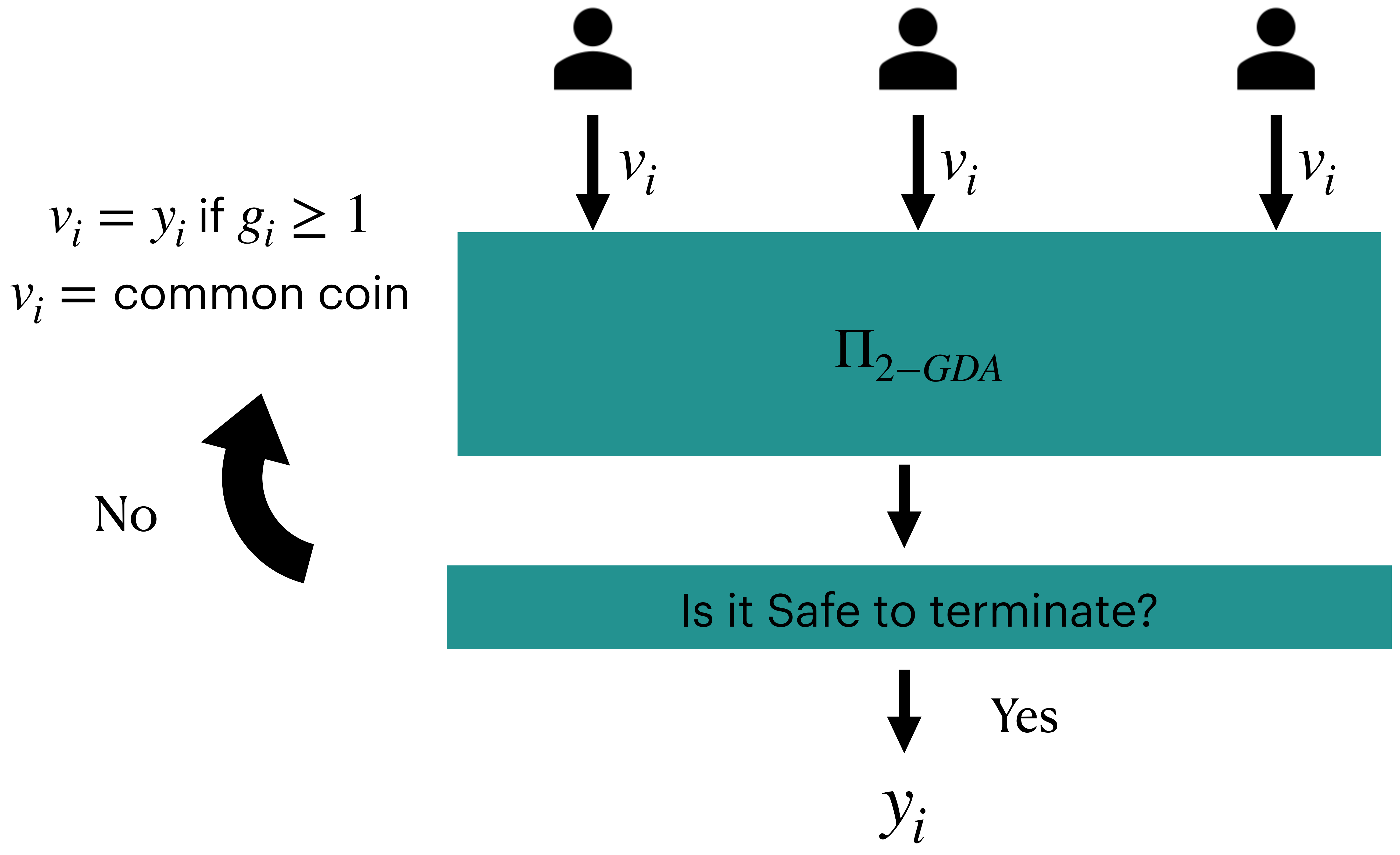


Building Block 1: 2-Graded *d*-Detecting Agreement

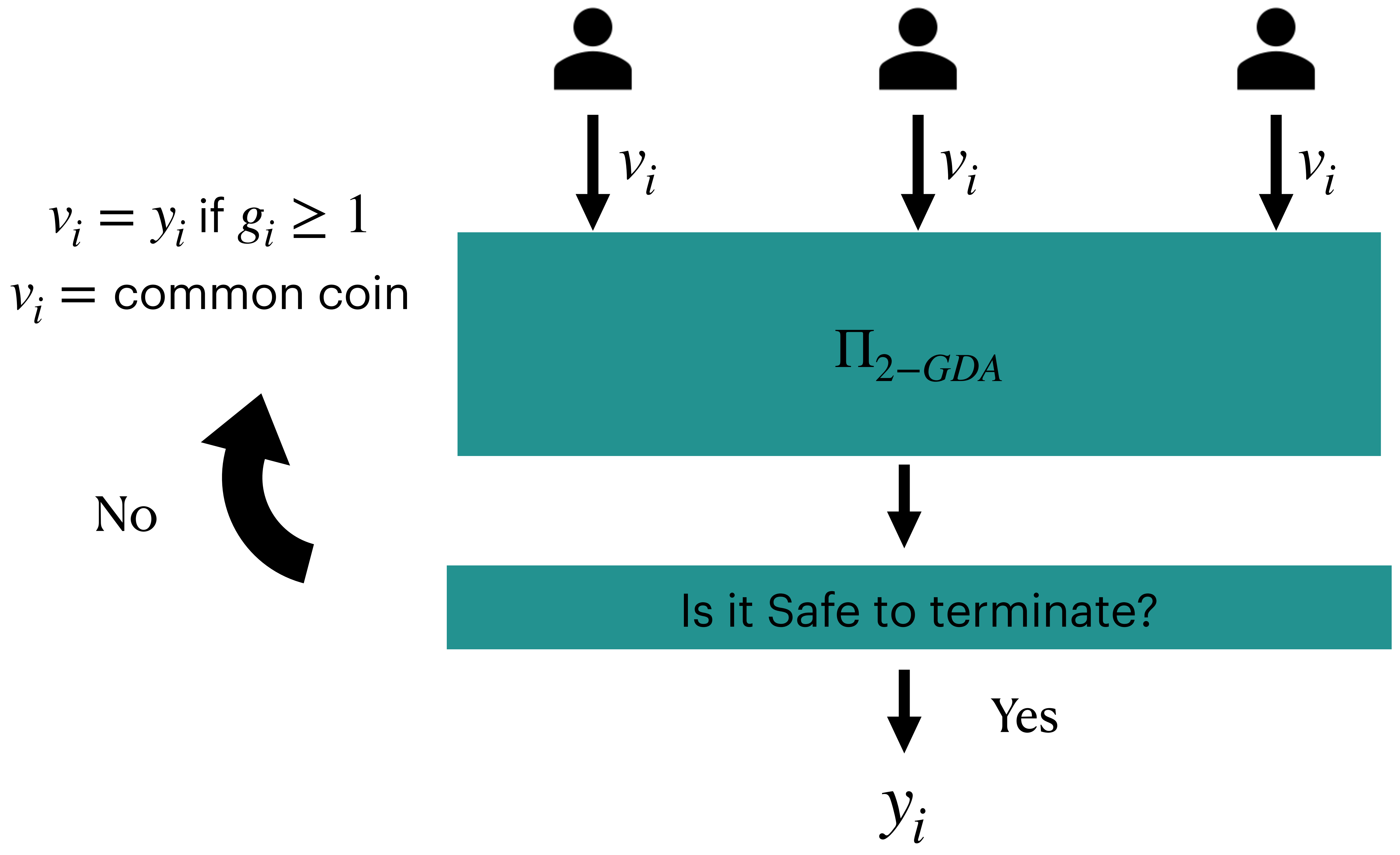


- If all honest parties input the same value v, they output $y_i = v, g_i = 2$
- If an honest party outputs $g_i < 2$, honest parties detect at least d malicious parties
- $d + 9$ round complexity

Building Block 2: Byzantine Agreement



Building Block 2: Byzantine Agreement



v_i

v_i

v_i

$v_i = y_i$ if $g_i \geq 1$

$v_i =$ common coin

Π_{2-GDA}

No

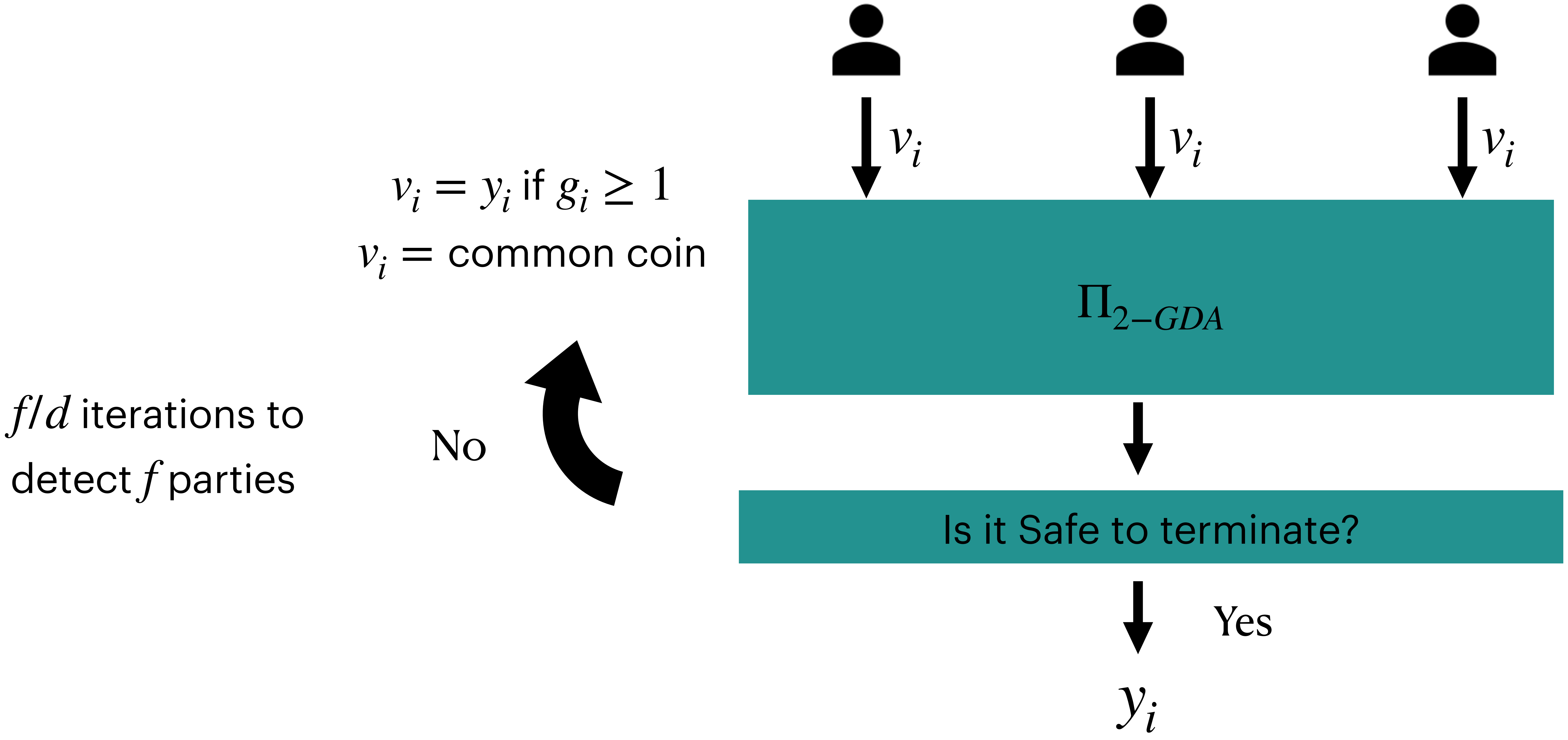
Is it Safe to terminate?

Yes

y_i

Expected Round Complexity: $O(1)$

Building Block 2: Byzantine Agreement



v_i v_i v_i

$v_i = y_i$ if $g_i \geq 1$

$v_i =$ common coin

Π_{2-GDA}

f/d iterations to detect f parties

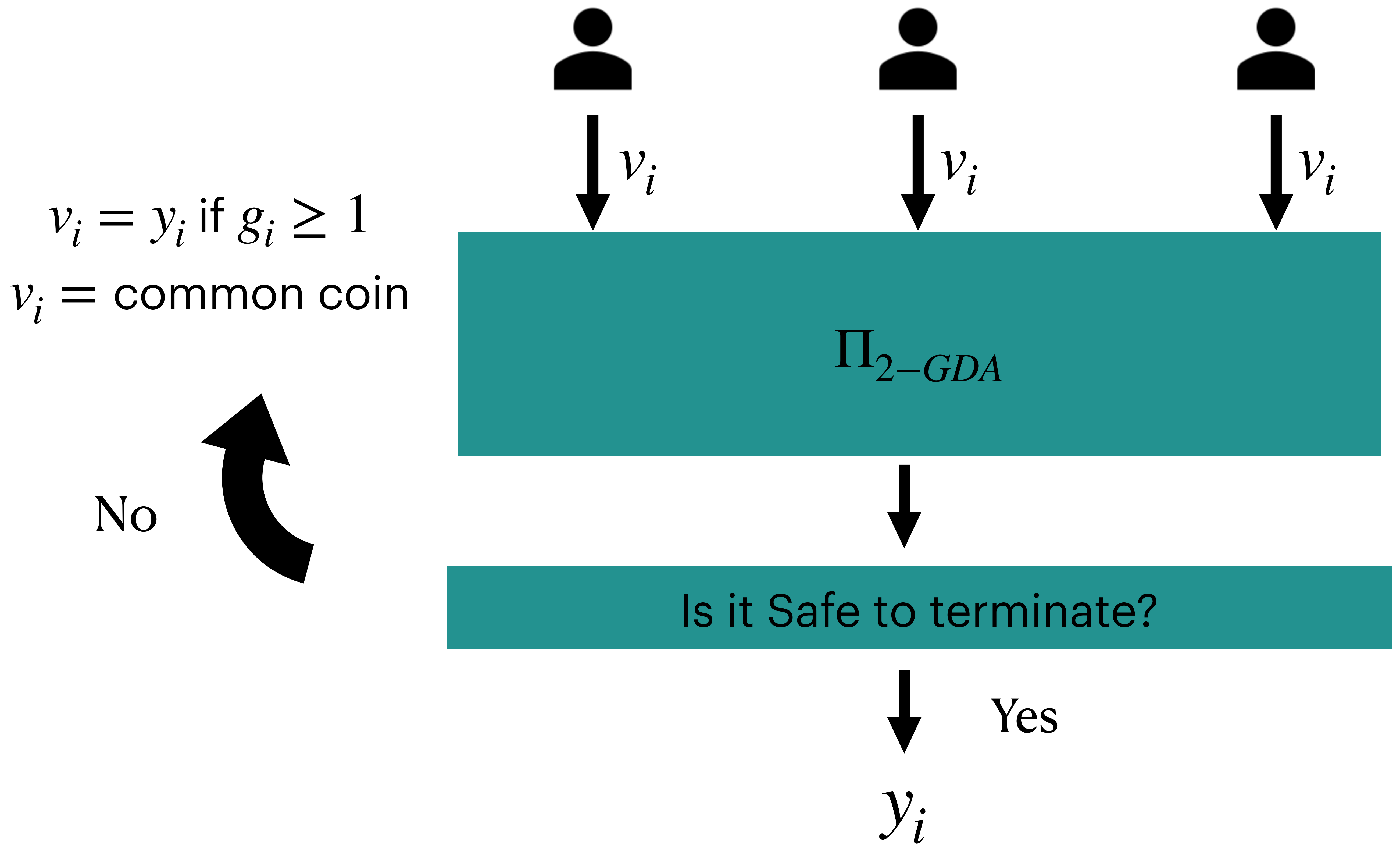
No

Is it Safe to terminate?

Yes

y_i

Building Block 2: Byzantine Agreement



v_i

v_i

v_i

$v_i = y_i$ if $g_i \geq 1$

$v_i =$ common coin

Π_{2-GDA}

No

Is it Safe to terminate?

Yes

y_i

f/d iterations to detect f parties

Worst-Case Round Complexity: $d + 9(f/d + 2)$

Thank you!