Agreement tasks in synchronous fault-prone networks

STACS 2025

Pierre Fraigniaud[★], **Minh-Hang Nguyen**[★], Ami Paz[†]

★ IRIF, CNRS and Université Paris Cité, France

† LISN, CNRS and Université Paris-Saclay, France

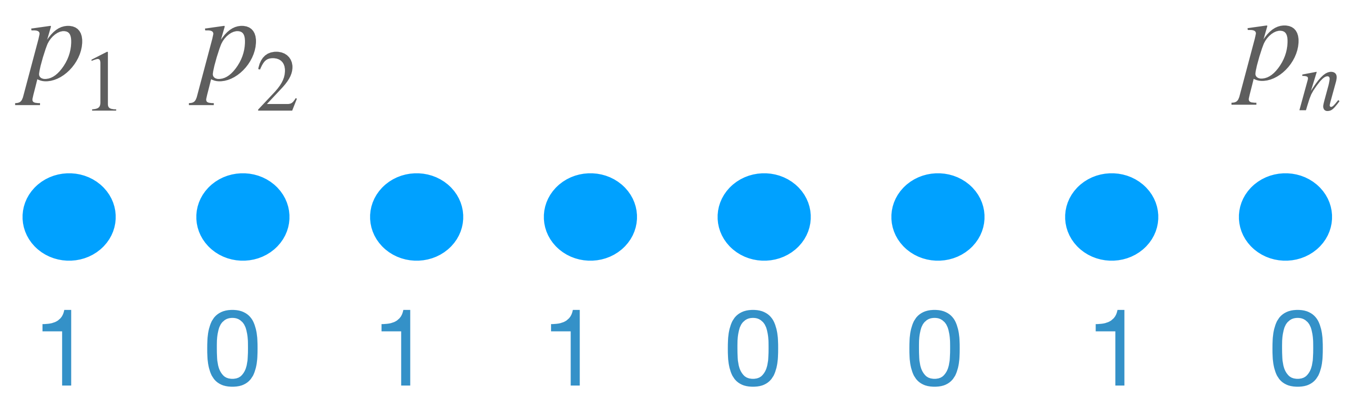
INSTITUT DE RECHERCHE EN INFORMATIQUE FONDAMENTALE

LISN LABORATOIRE INTERDISCIPLINAIRE DES SCIENCES DU NUMÉRIQUE

The European Union's Horizon 2020 research and innovation programme under the Marie Skłodowska -Curie grant No 945332.

Consensus

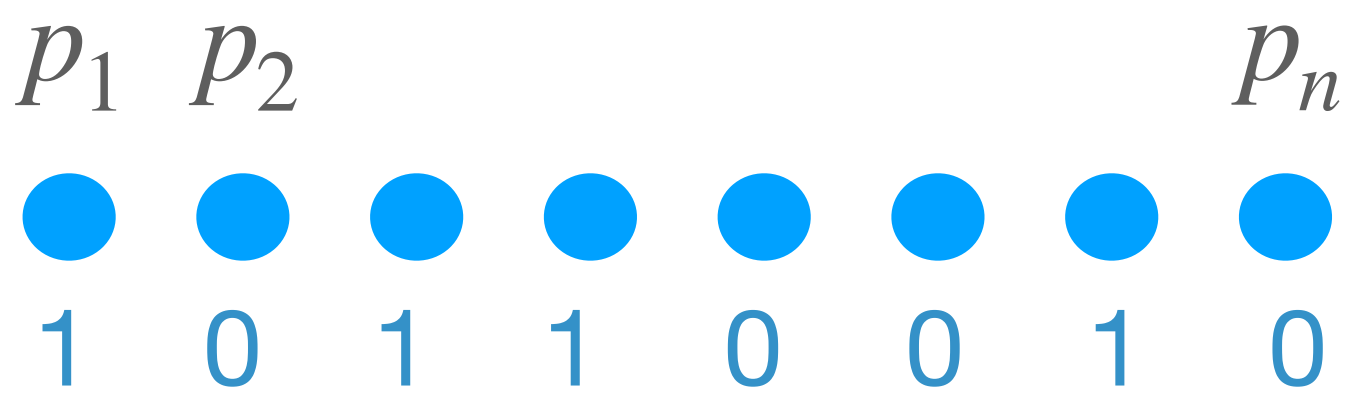
Input: every process has an input in $\{0,1\}$



Output:

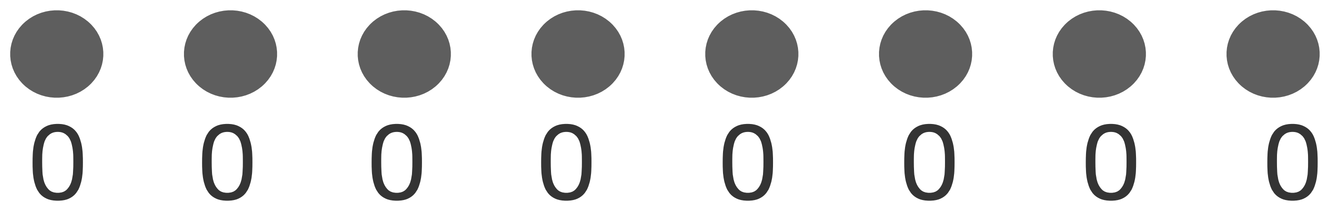
Consensus

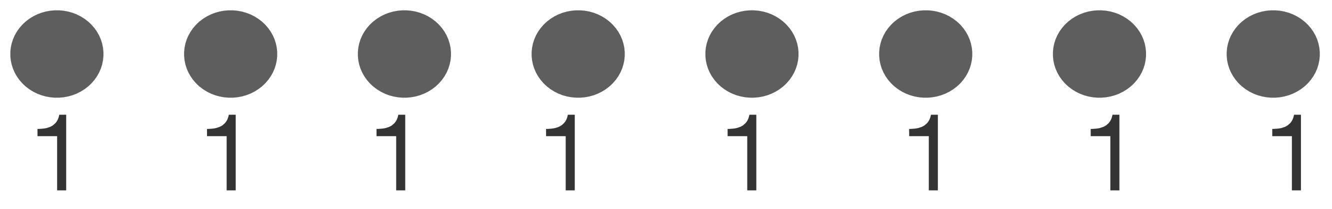
Input: every process has an input in $\{0,1\}$



Output:

Agreement: globally, processes decide on **the same value**



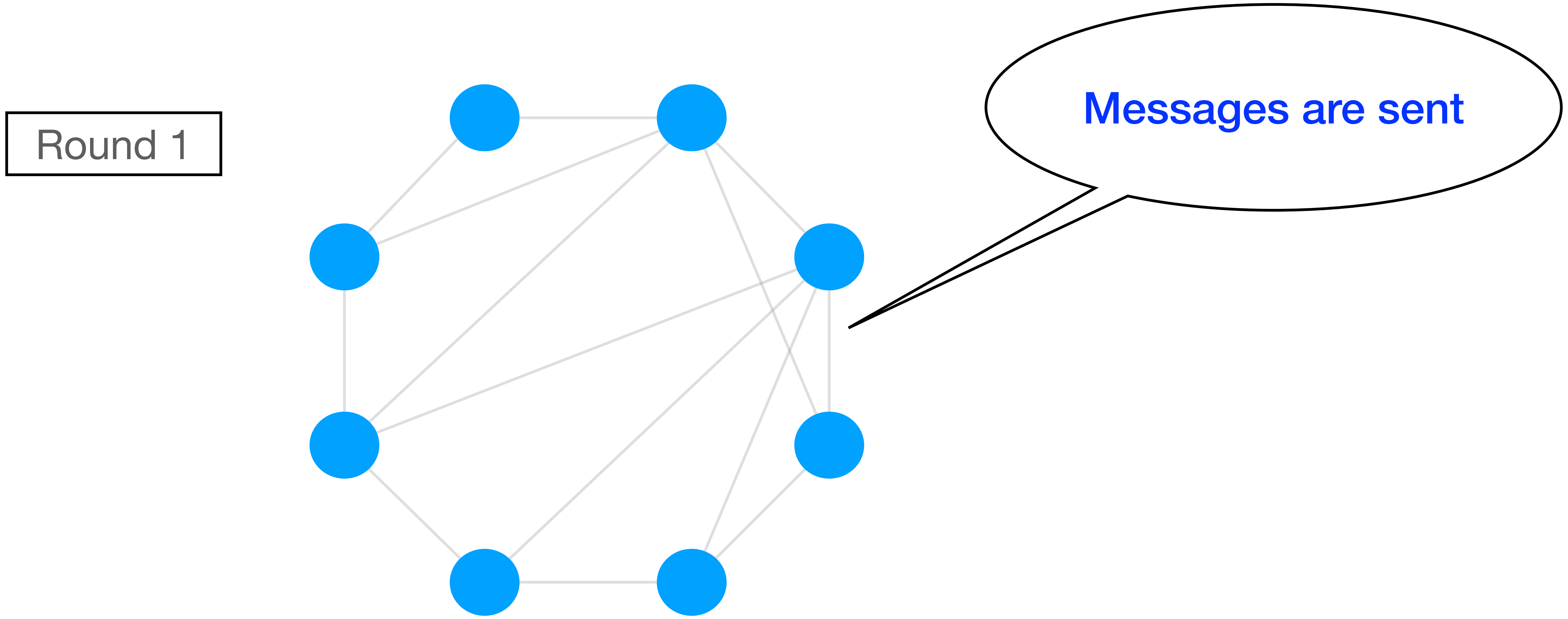


Validity: decide on a value appearing in the inputs

0 0 0 0 0 0 0 0 $\longrightarrow$ 0 0 0 0 0 0 0 0

1 1 1 1 1 1 1 1 $\longrightarrow$ 1 1 1 1 1 1 1 1

Synchronous message-passing Network

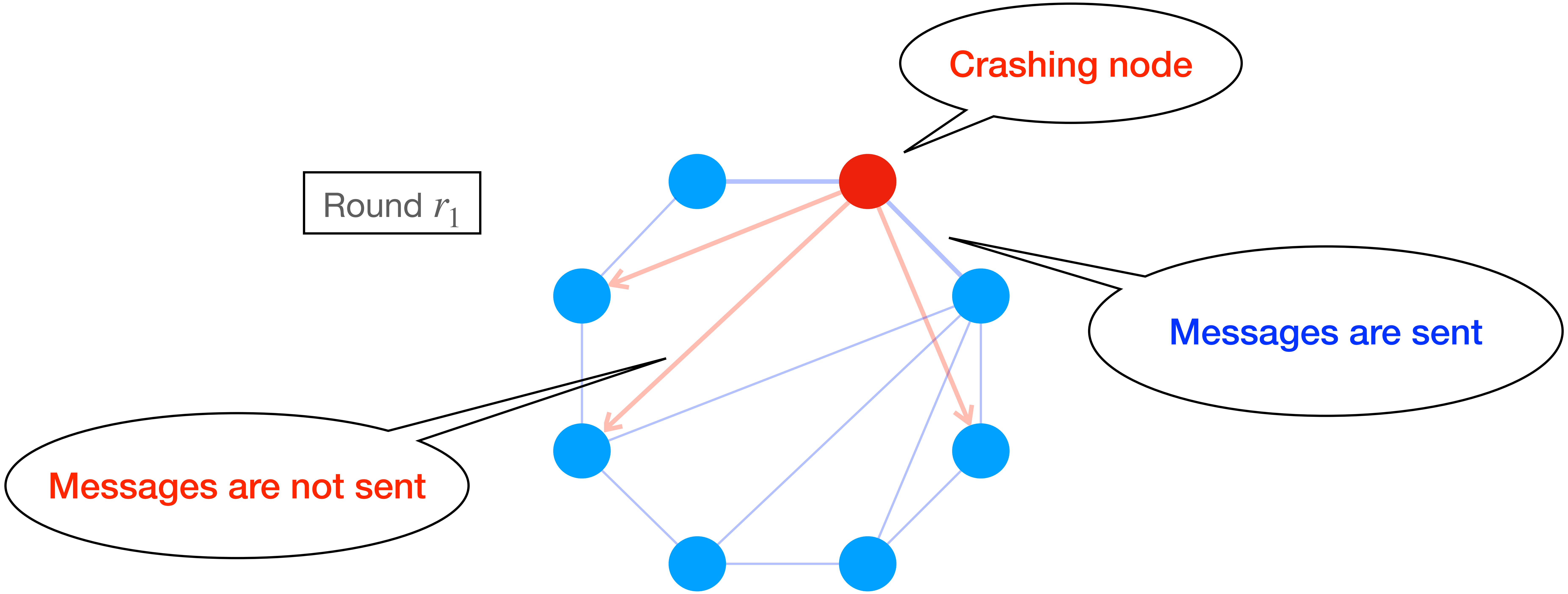


Every round, every process

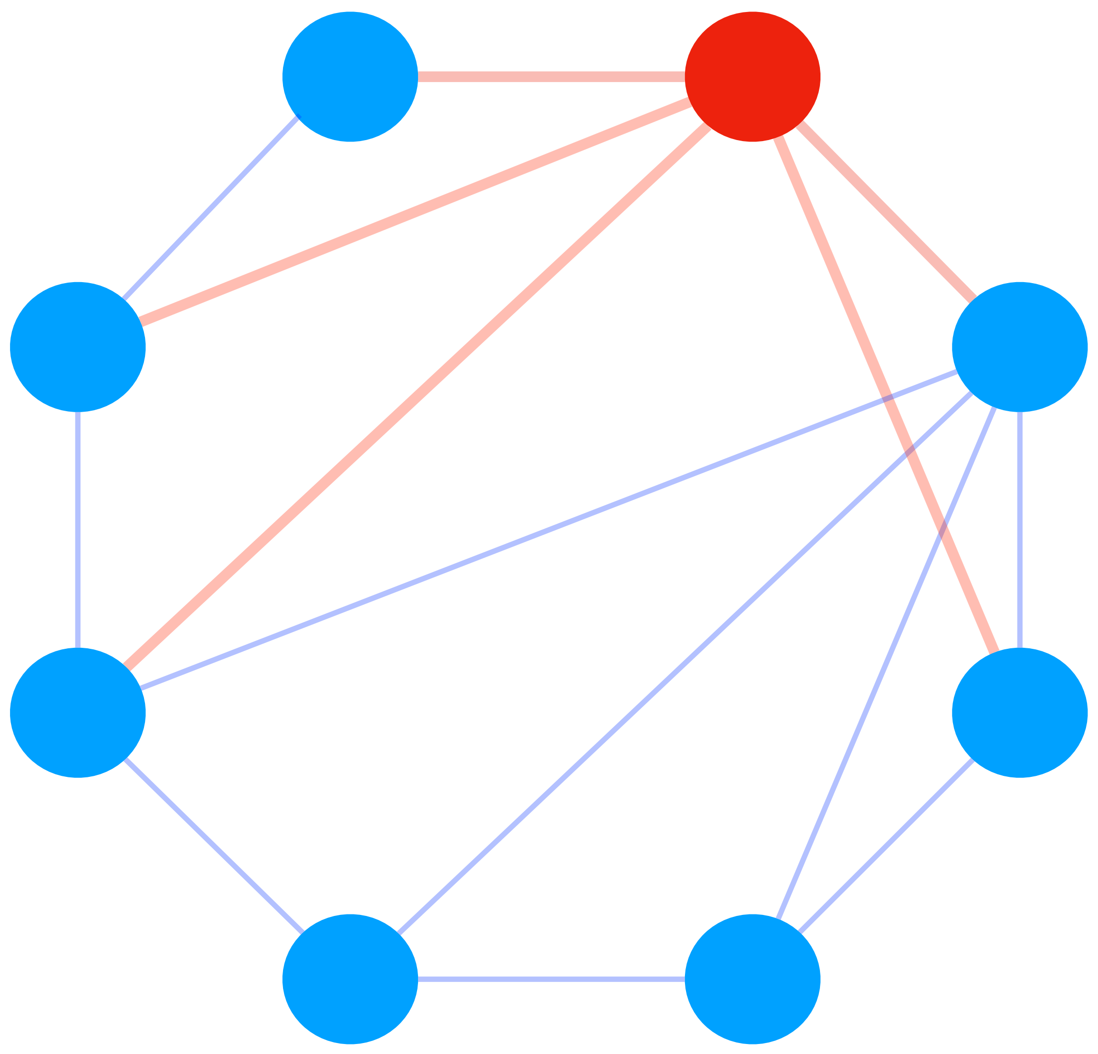
Sends messages to its neighbours.

Receives messages from its neighbours.

Synchronous crash-prone message-passing Network



Synchronous crash-prone message-passing Network

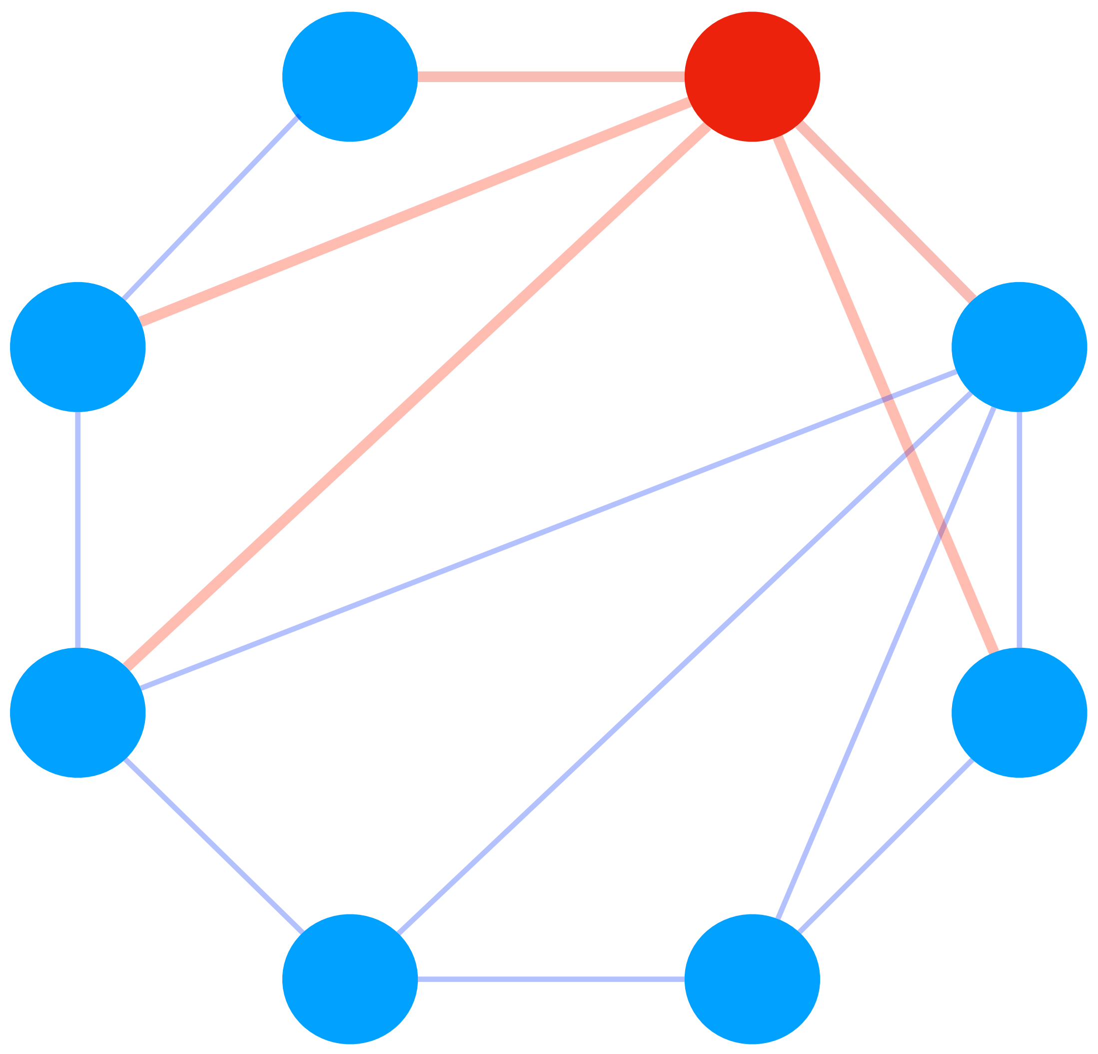


Oblivious Algorithms:

After r rounds, every process knows a part of the global input (view)

Decision: $\{ (p_i, inp_{p_i}), (p_j, inp_{p_j}), \ldots \} \mapsto$ output

Synchronous crash-prone message-passing Network



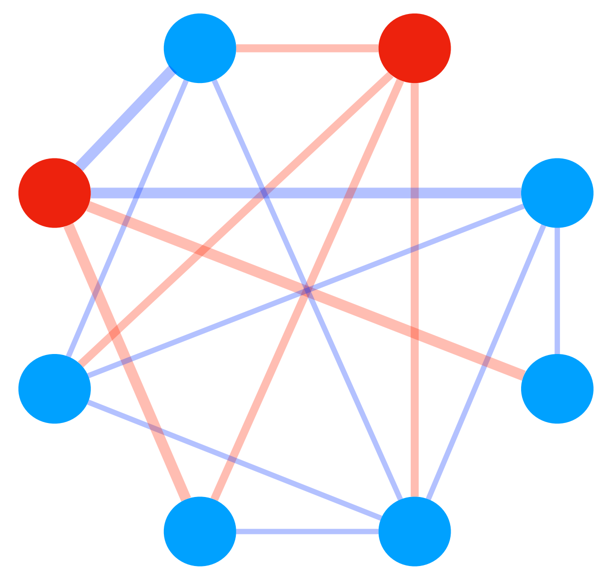
How many rounds do we need to solve consensus?

Oblivious Algorithms:

After r rounds, every process knows a part of the global input (view)

Decision: $\{ (p_i, inp_{p_i}), (p_j, inp_{p_j}), \ldots \} \mapsto$ output

Agreement tasks in crash-prone Network



From cliques to general graphs

	Clique	*General graph*
Consensus	$(t+1)$ [1,2]	UB: $rad(G,t)$ [3] LB: $rad(G,t)$ [4]

[1] *Dolev, Strong* '83

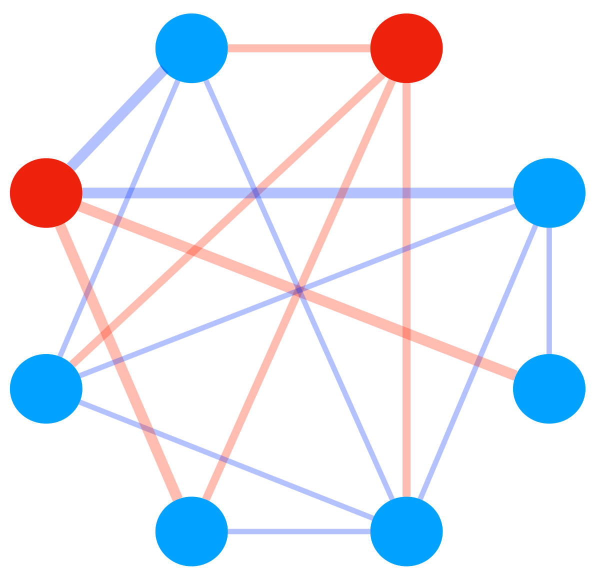
[2] *Aguilera, Toueg* '99

[3] *Castaneda, Fraigniaud, Paz, Rajsbaum, Roy, Travers* '23

[4] *Fraigniaud, N., Paz,* '25

Theorem: *For every graph G and every $t < \kappa(G)$, consensus in G cannot be solved in less than $rad(G,t)$ rounds by an oblivious algorithm in the t-resilient model.*

Agreement tasks in crash-prone Network



From cliques to general graphs

	Clique	***General graph***
Consensus	$(t+1)$ [1,2]	UB: $rad(G,t)$ [3] LB: $rad(G,t)$ [4]

Theorem: *For every graph G and every $t < \kappa(G)$, consensus in G cannot be solved in less than $rad(G,t)$ rounds by an oblivious algorithm in the t-resilient model.*

New graph parameter: Radius under failures

Failure Patterns φ_j : Which nodes crash, when and how they crash.

Eccentricity $ecc(v,\varphi)$: Broadcast time of v under φ.

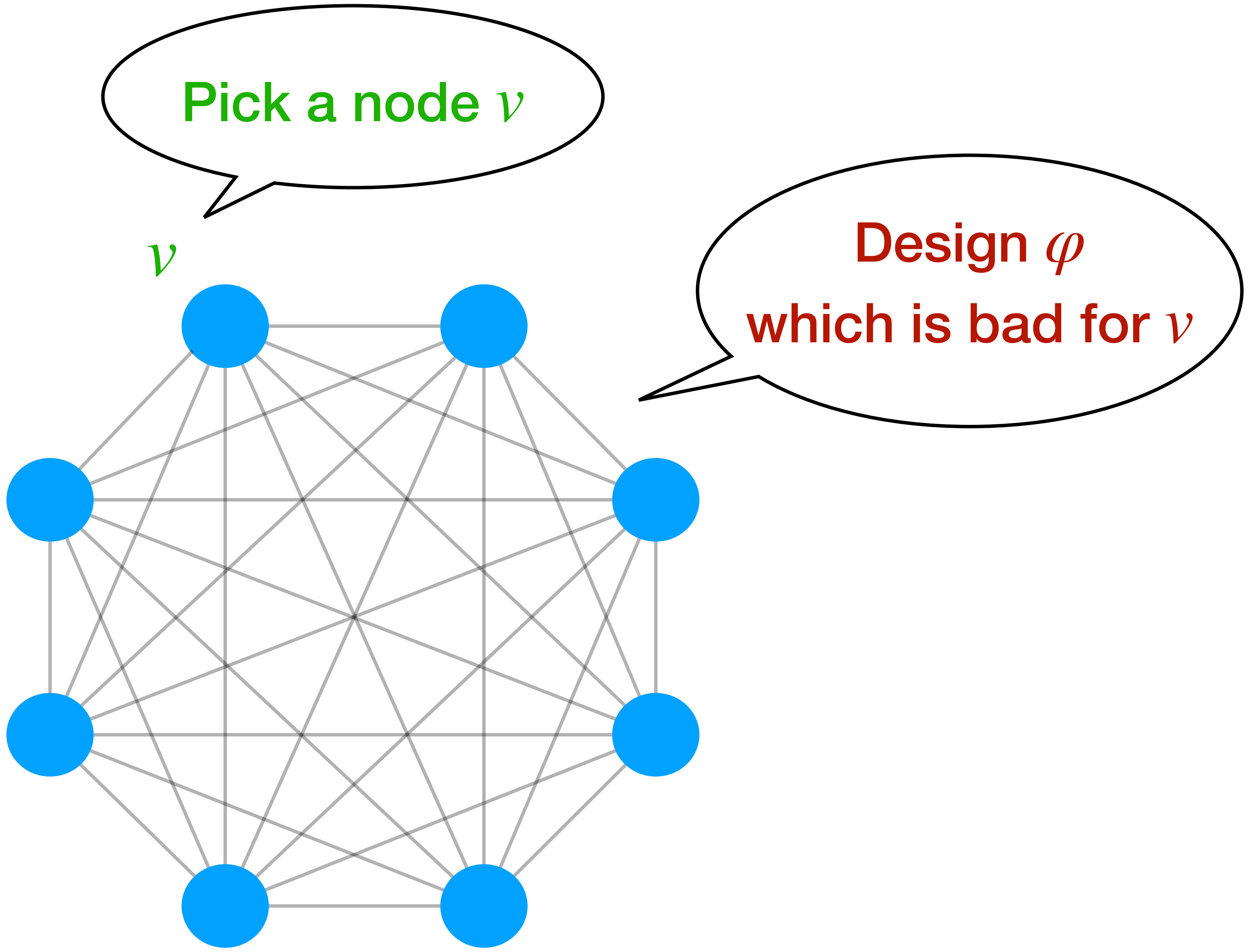
	φ_1	φ_j	φ_ℓ	Min
v_1		broadcast time		Max
v_i		$ecc(v_i,\varphi_j)$		Max
v_n	25	∞	10	**25**

$$rad(G,t) = \min_{v} \max_{\varphi} \{ecc(v,\varphi)\}$$

Best node

Worst failure pattern for v

Radius under failures



$$rad(G, t) = t + 1$$

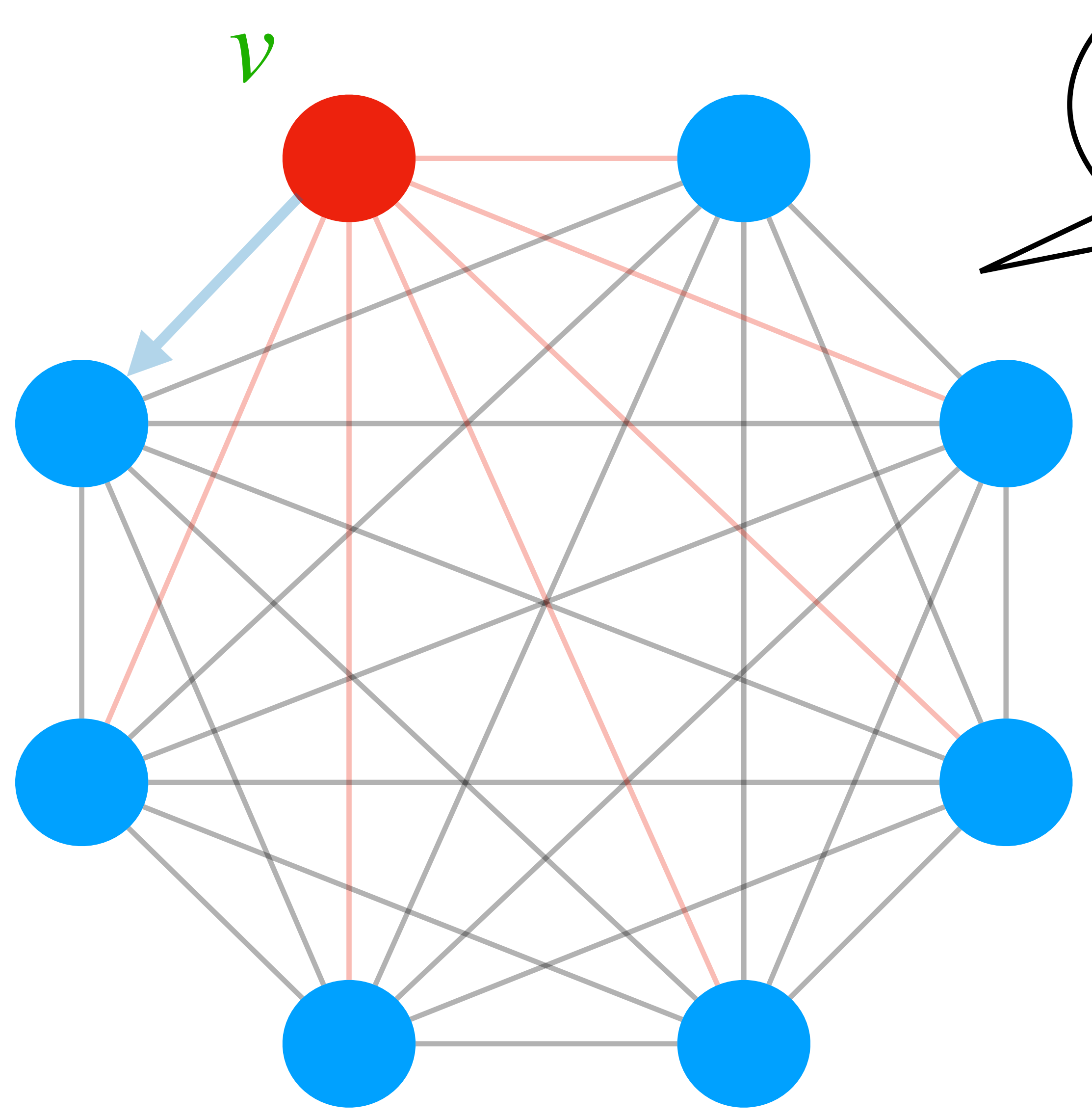
$$rad(G, t) = \min_{v} \max_{\varphi} \{ecc(v, \varphi)\}$$

Best node

Worst failure pattern for v

Radius under failures

Round 1



Design φ
which is bad for v

$rad(G,t) = t + 1$

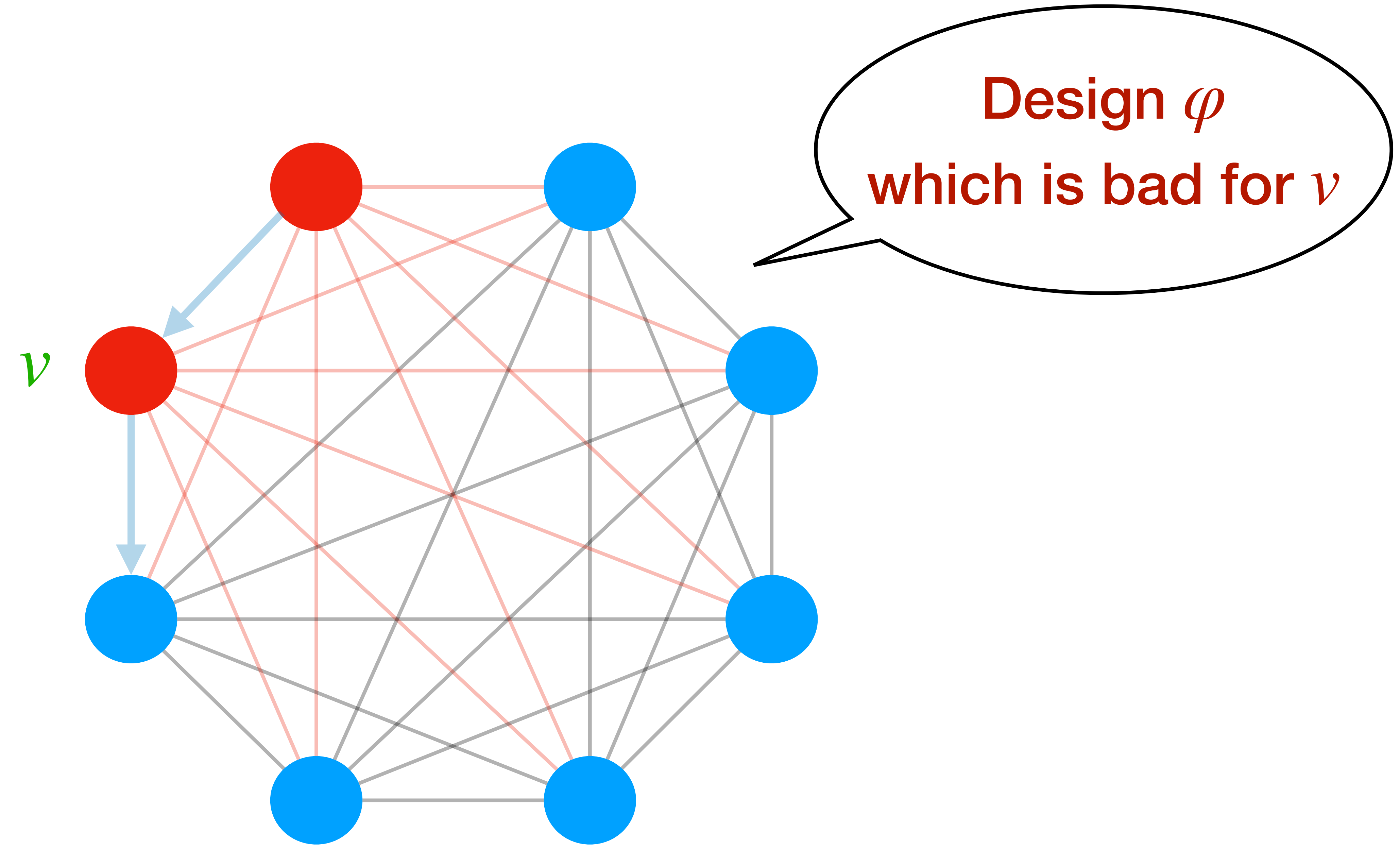
$$rad(G,t) = \min_{v} \max_{\varphi}\{ecc(v,\varphi)\}$$

Best node

Worst failure pattern for v

Radius under failures

Round 2

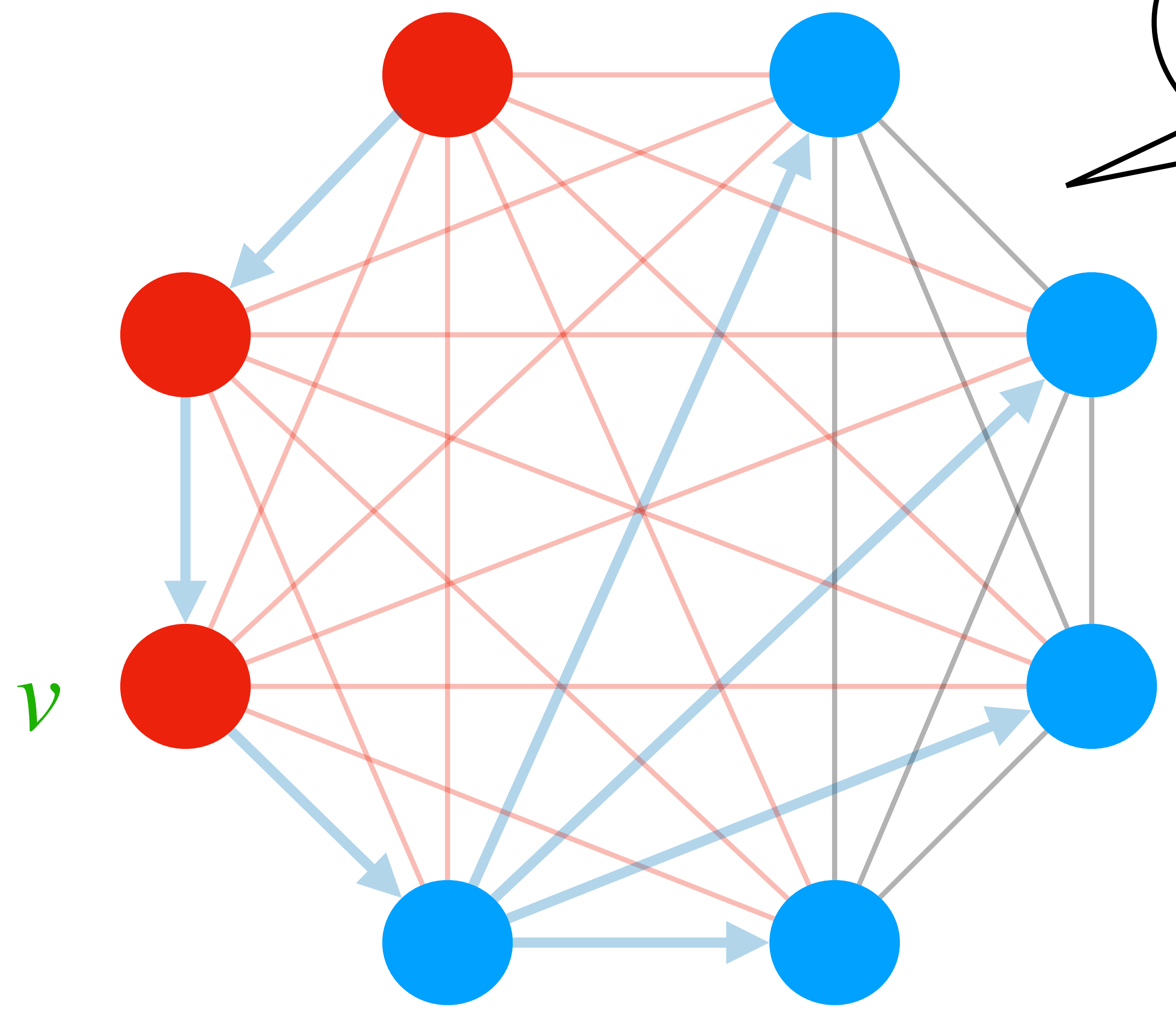


$rad(G, t) = t + 1$

$$rad(G, t) = \min_{v} \max_{\varphi} \{ecc(v, \varphi)\}$$

Radius under failures

Round t



Design φ
which is bad for v

$rad(G, t) = t + 1$

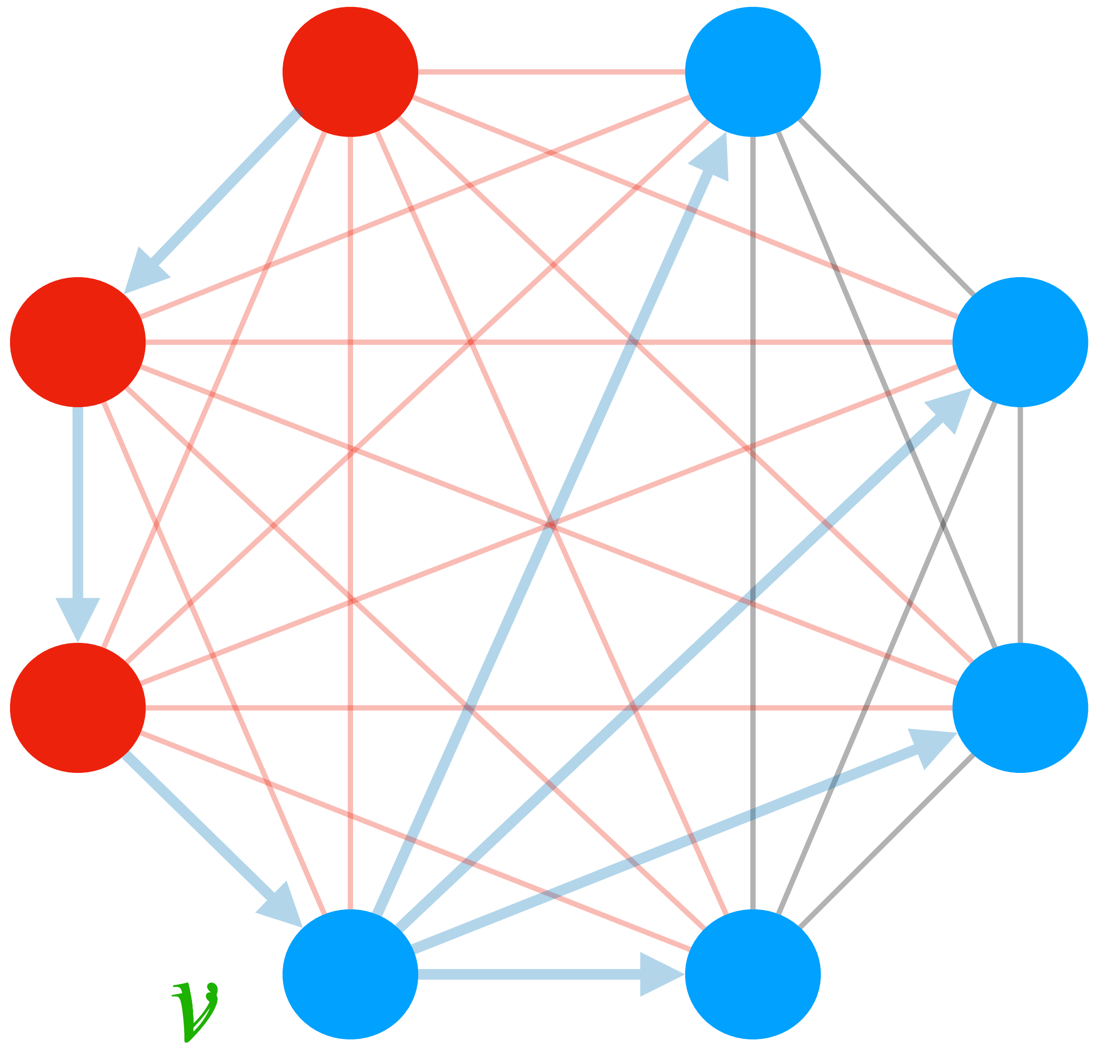
$$rad(G,t) = \min_{v} \max_{\varphi} \{ecc(v, \varphi)\}$$

Best node

Worst failure pattern
for v

Radius under failures

Round $t+1$



$rad(G,t) = t+1$

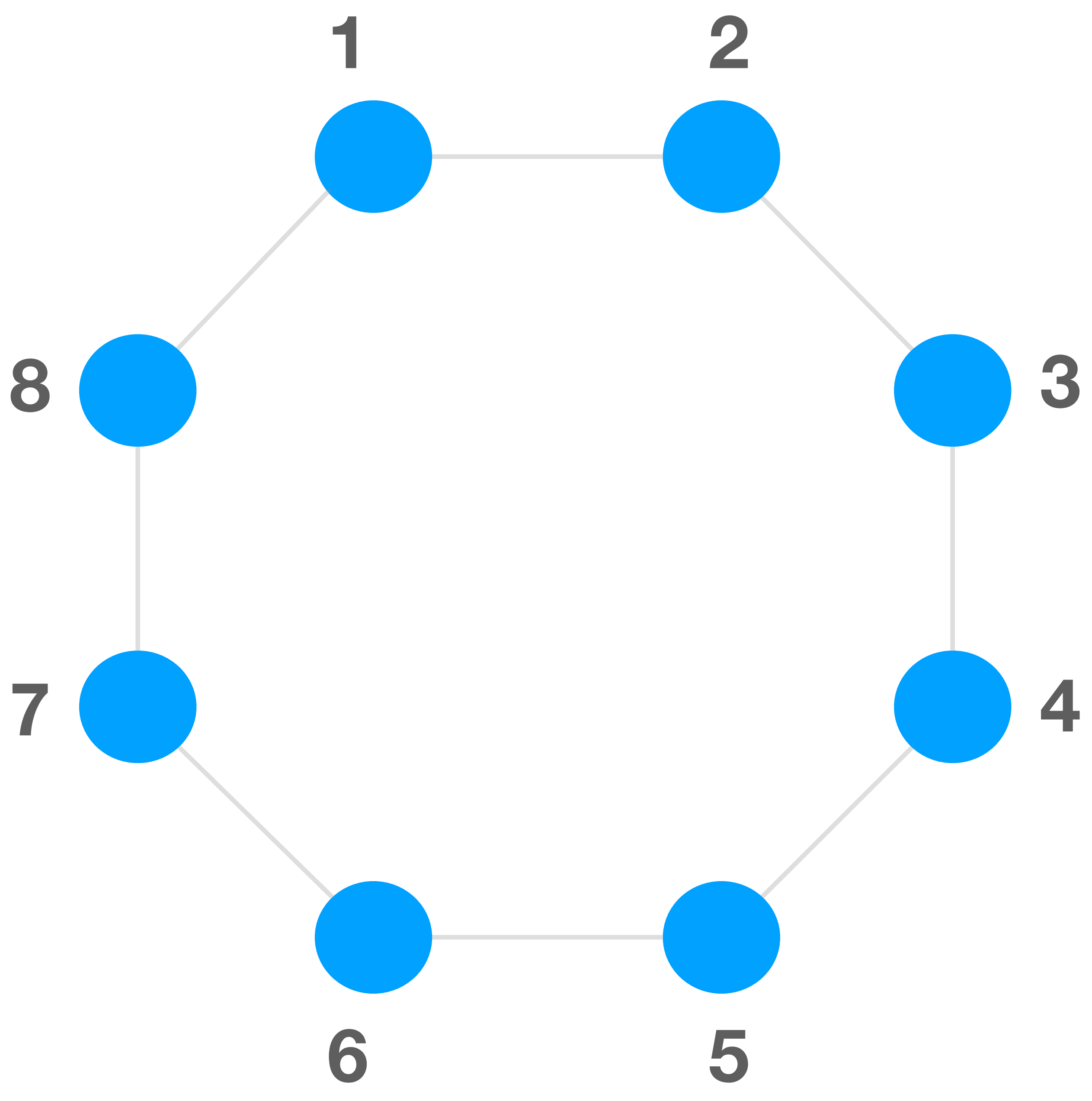
$\max_{\varphi} ecc(v,\varphi) \ = t+1$

$rad(G,t) = \min_{v} \max_{\varphi} \{ecc(v,\varphi)\}$

Best node

Worst failure pattern for v

Main tool: Information flow graph



G

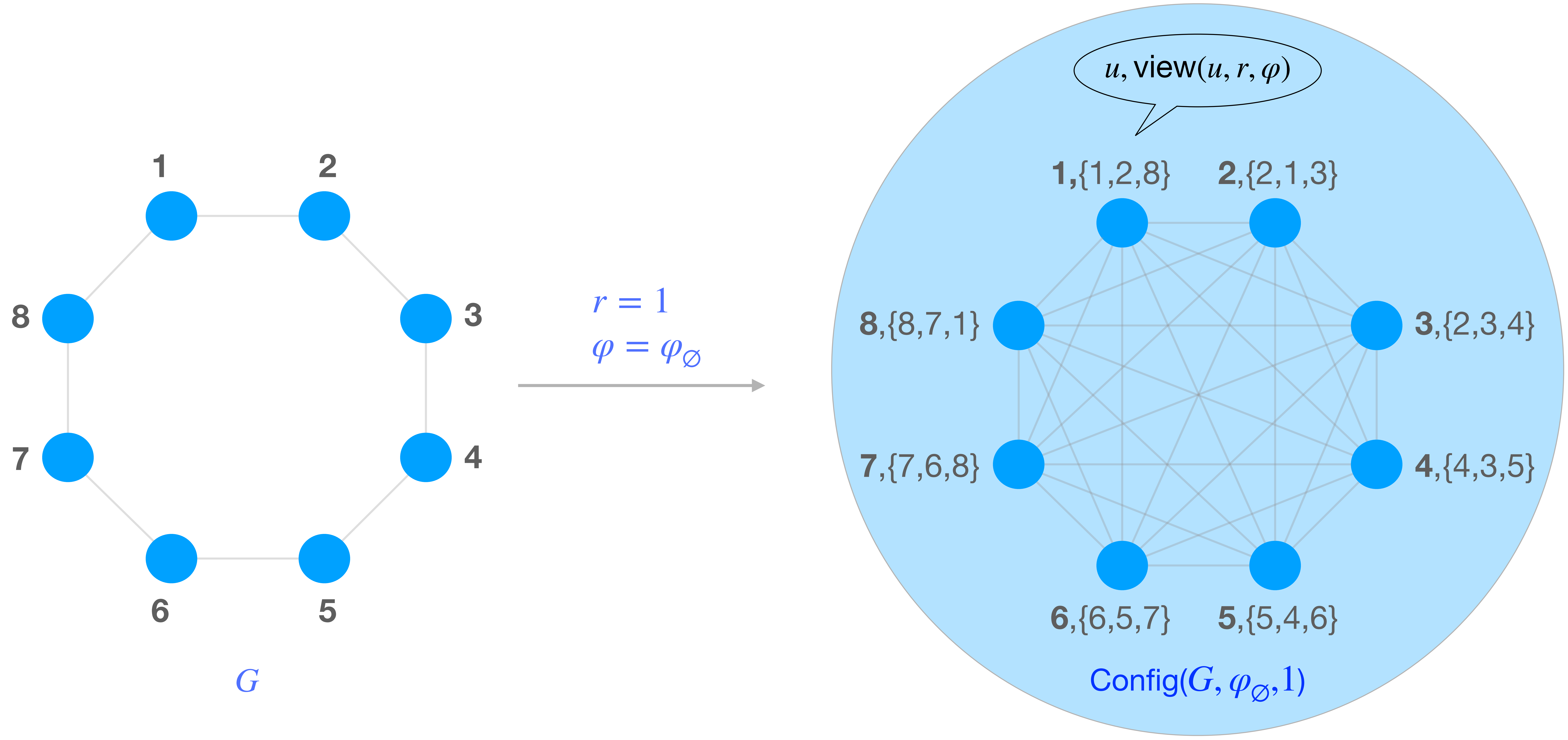
$r = 1$

$\varphi = \varphi_{\varnothing}$

Config($G, \varphi_{\varnothing}, 1$)

Config(G, φ, r): Local states of processes under φ after r rounds

Main tool: Information flow graph



Config(G, φ, r): Local states of processes under φ after r rounds

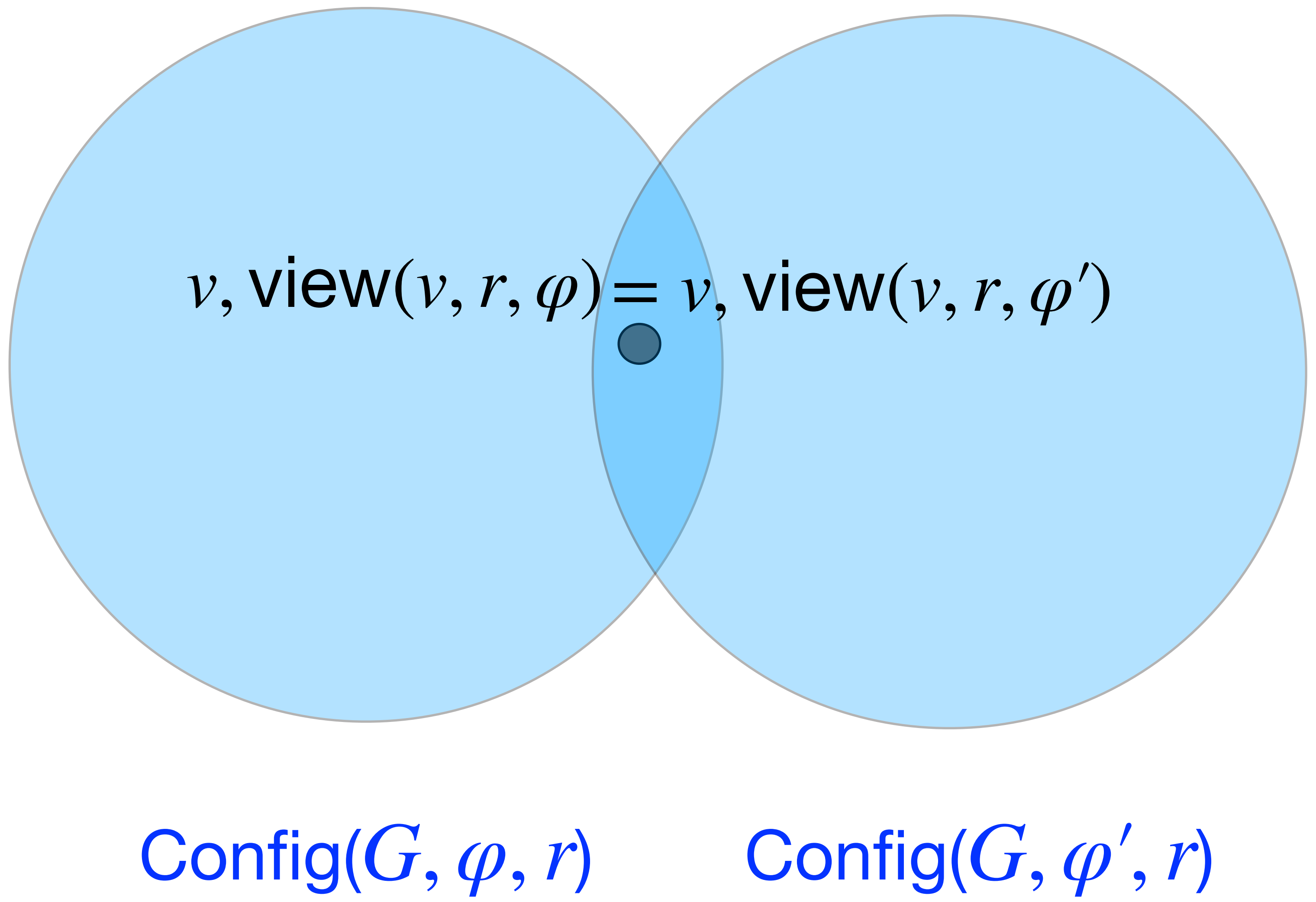
Main tool: Information flow graph

G : A fix graph

Φ : Set of failure patterns

$\varphi, \varphi' \in \Phi$

Information flow graph $IF(G, r, \Phi)$

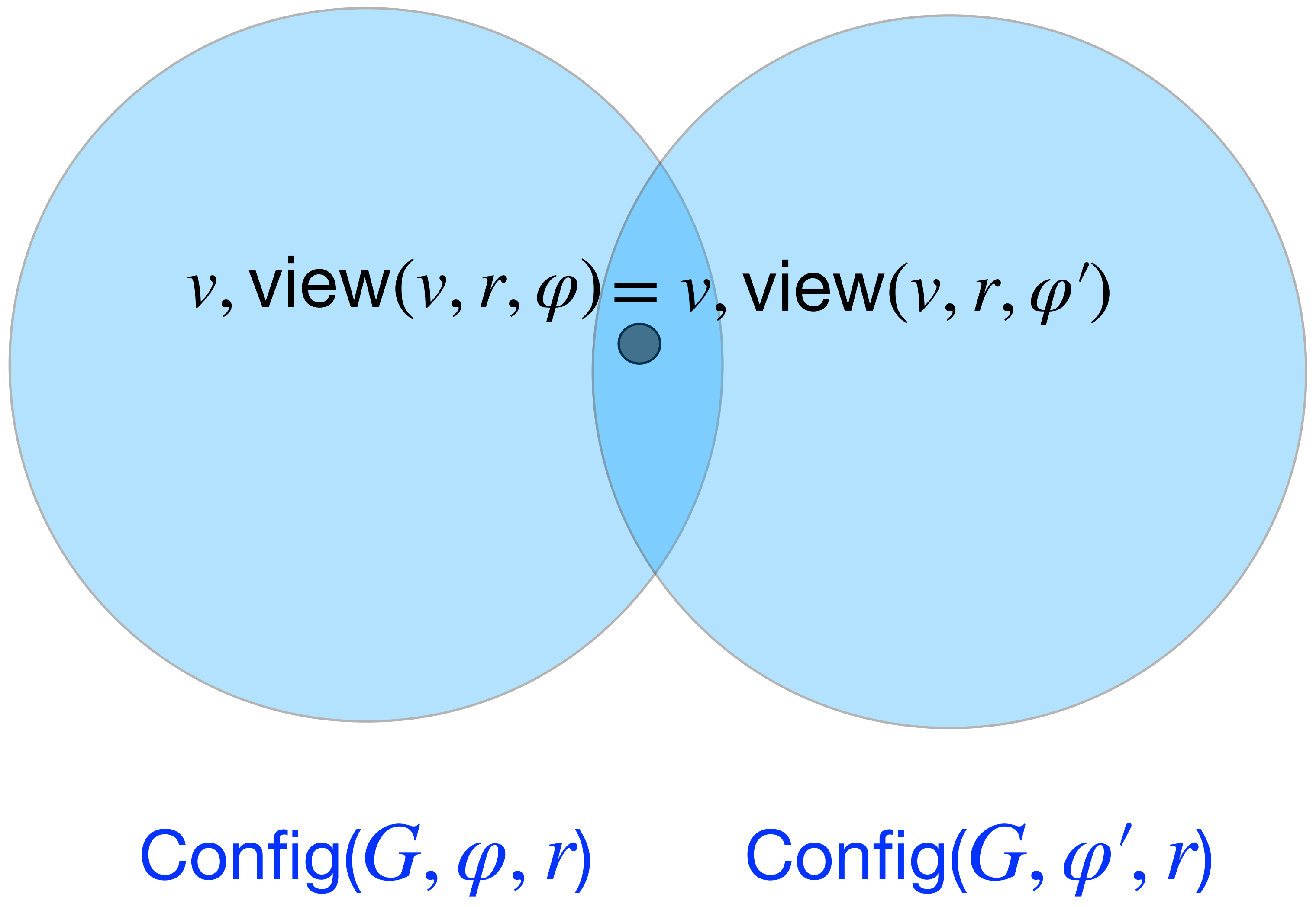


Main tool: Information flow graph

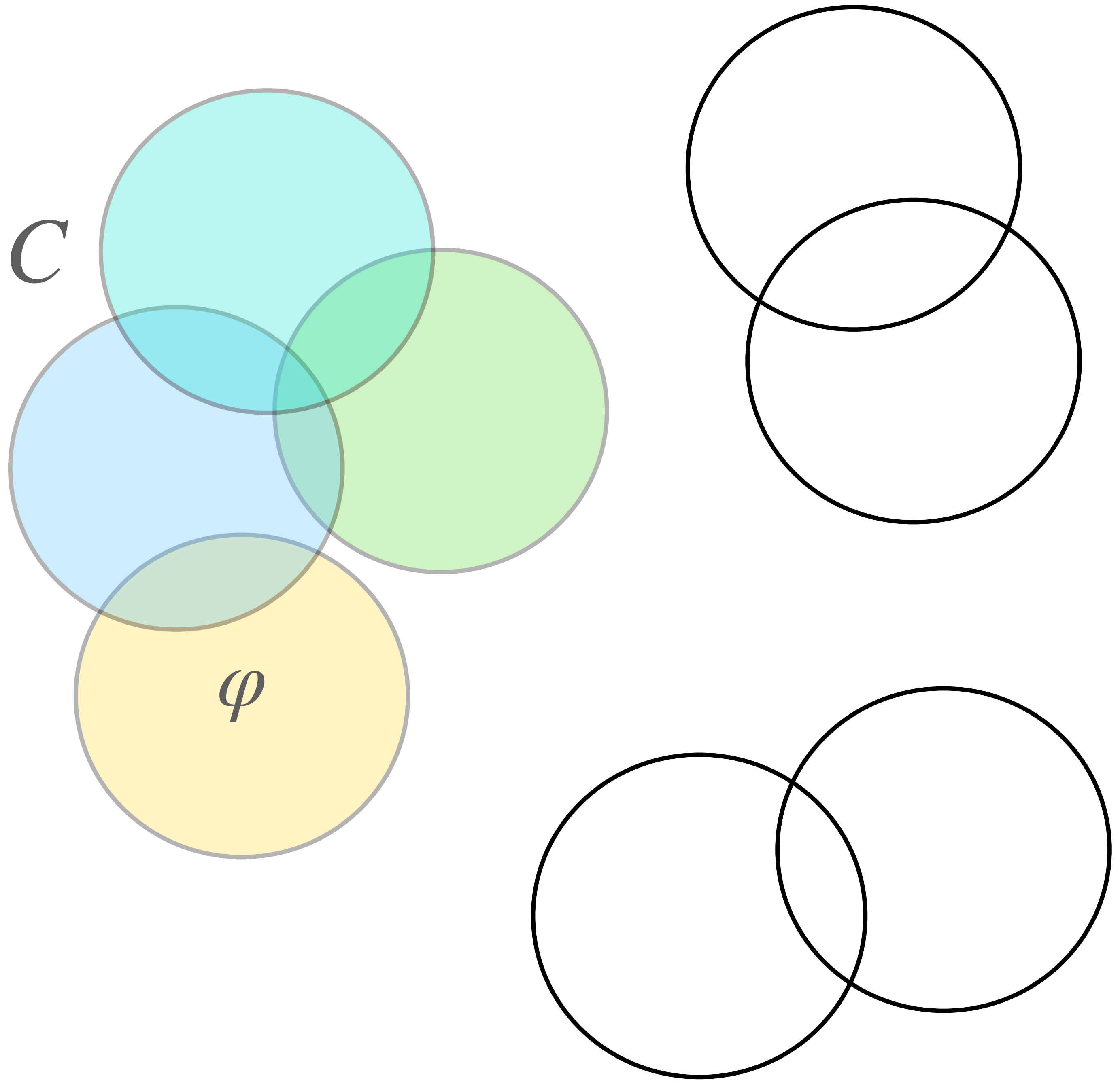
G : A fix graph

Φ : Set of failure patterns

$\varphi, \varphi' \in \Phi$



Information flow graph $IF(G, r, \Phi)$

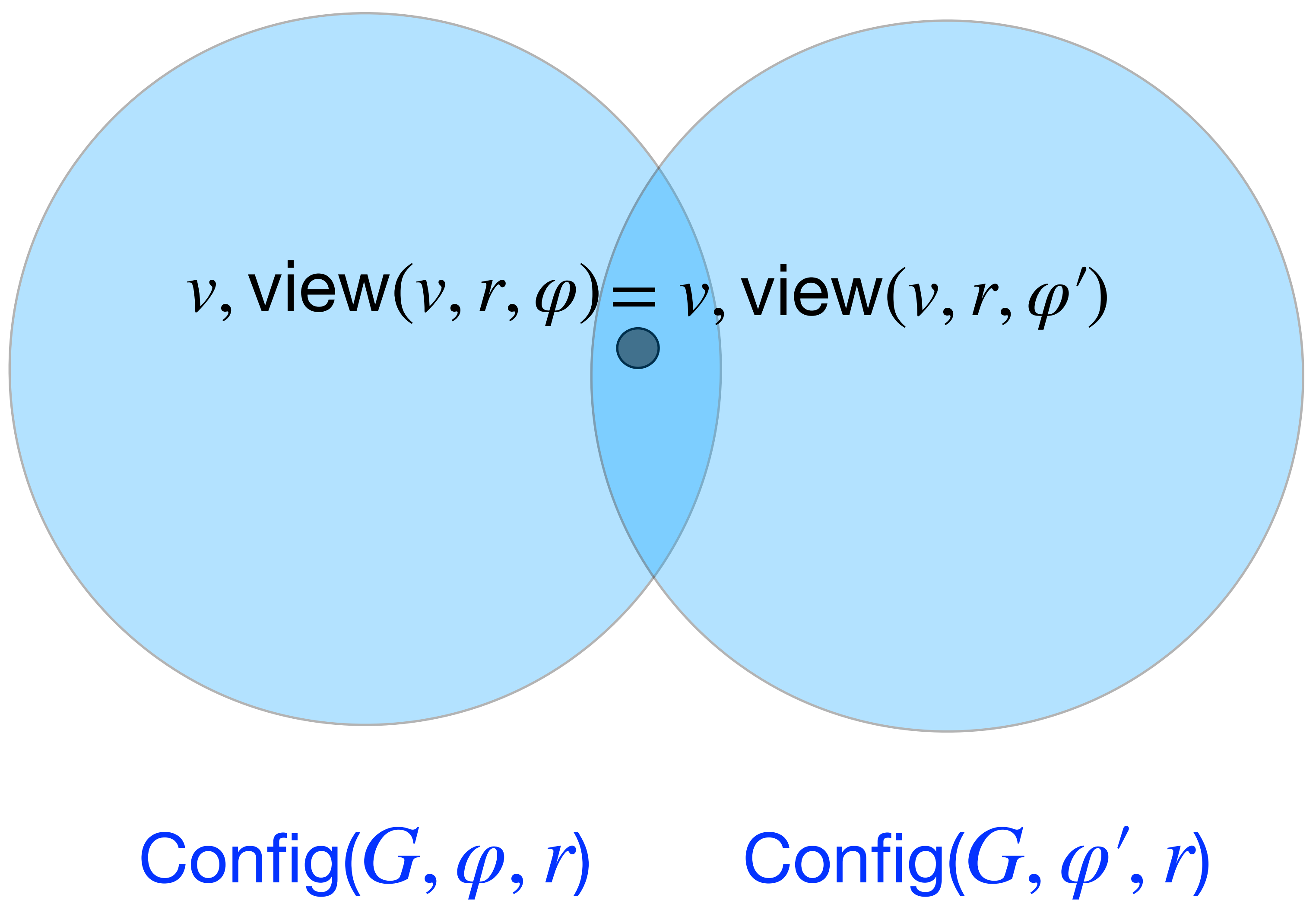


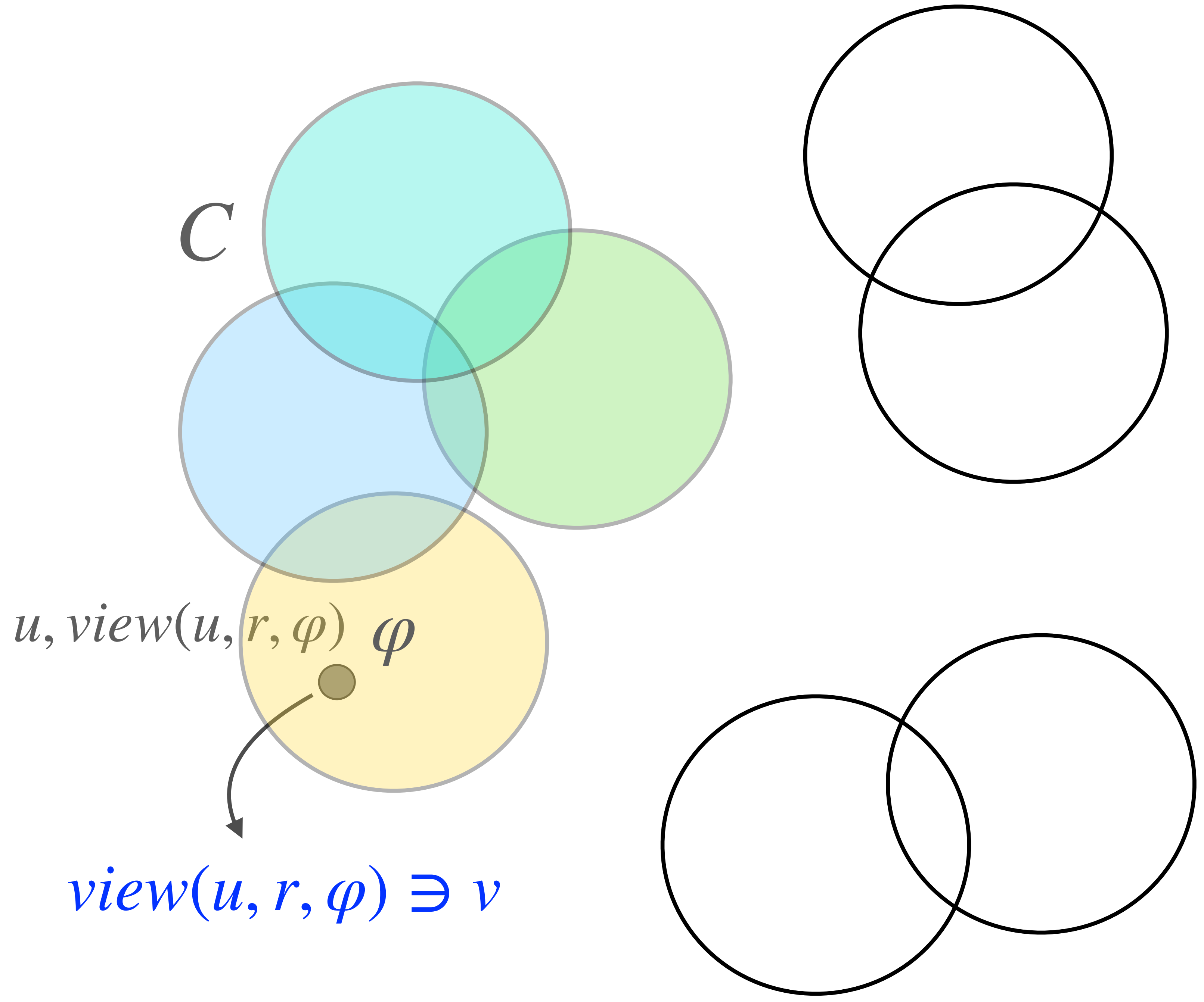
Main tool: Information flow graph

G : A fix graph Φ : Set of failure patterns

$\varphi, \varphi' \in \Phi$

Information flow graph $IF(G, r, \Phi)$



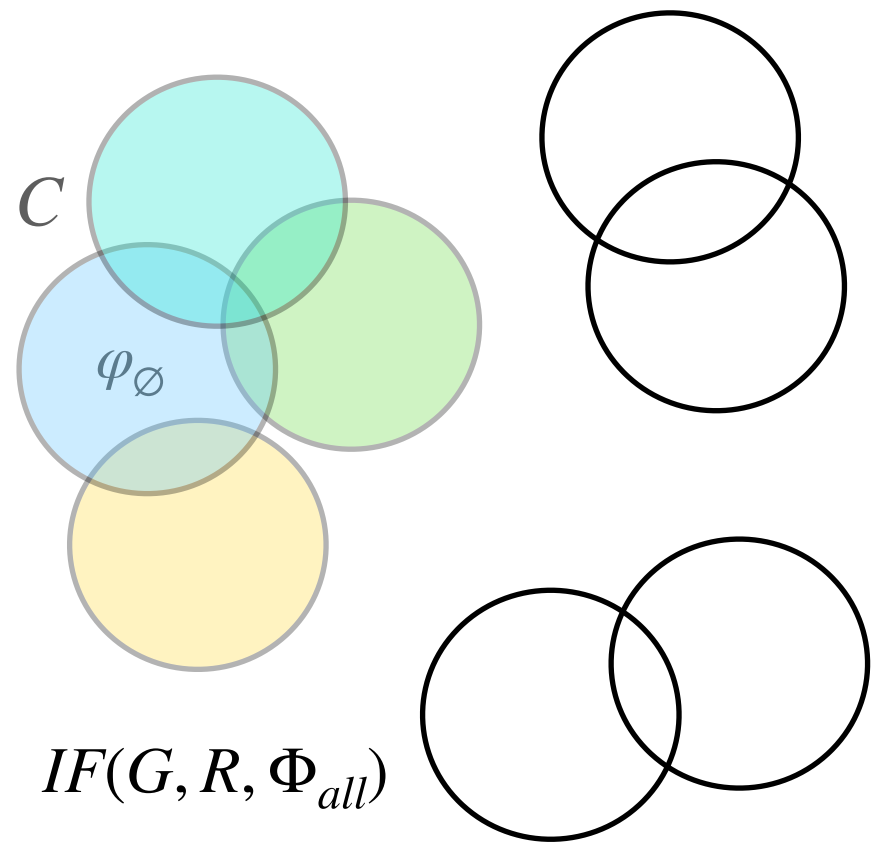


Given G and Φ, there is an oblivious algorithm solving consensus in ***r rounds*** *iff every* ***connected component*** *C of $IF(G, r, \Phi)$* ***has a dominating node*** *v.*

Castaneda, Fraigniaud, Paz, Rajsbaum, Roy, Travers'19

Outline of the proof

LB: $rad(G, t)$



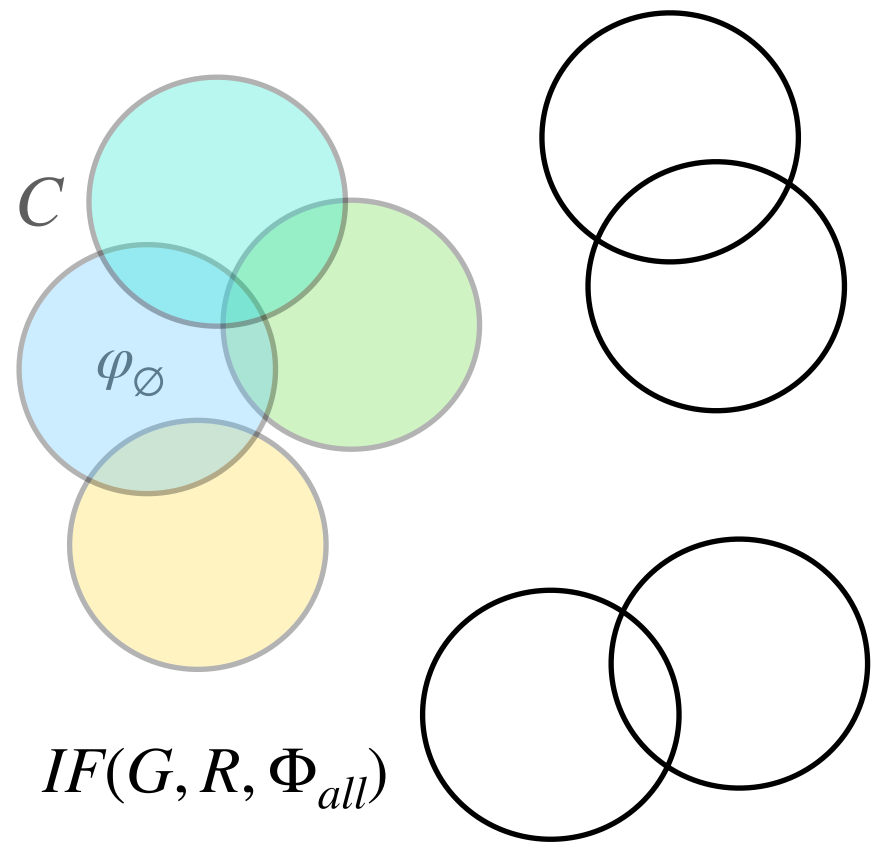
$R < rad(G, t) \rightarrow \exists \varphi, ecc(v, \varphi) > R$

A well chosen φ_v → φ_v, $\varphi_\varnothing$ are in the same component C → No node dominating component C

No oblivious algorithm solving consensus in R rounds

Outline of the proof

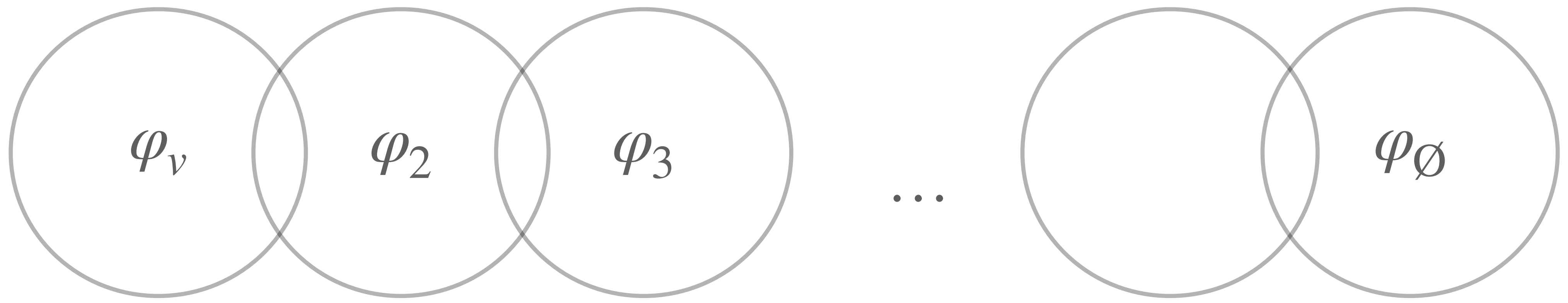
LB: $rad(G, t)$



$R < rad(G, t) \rightarrow \exists \varphi, ecc(v, \varphi) > R$

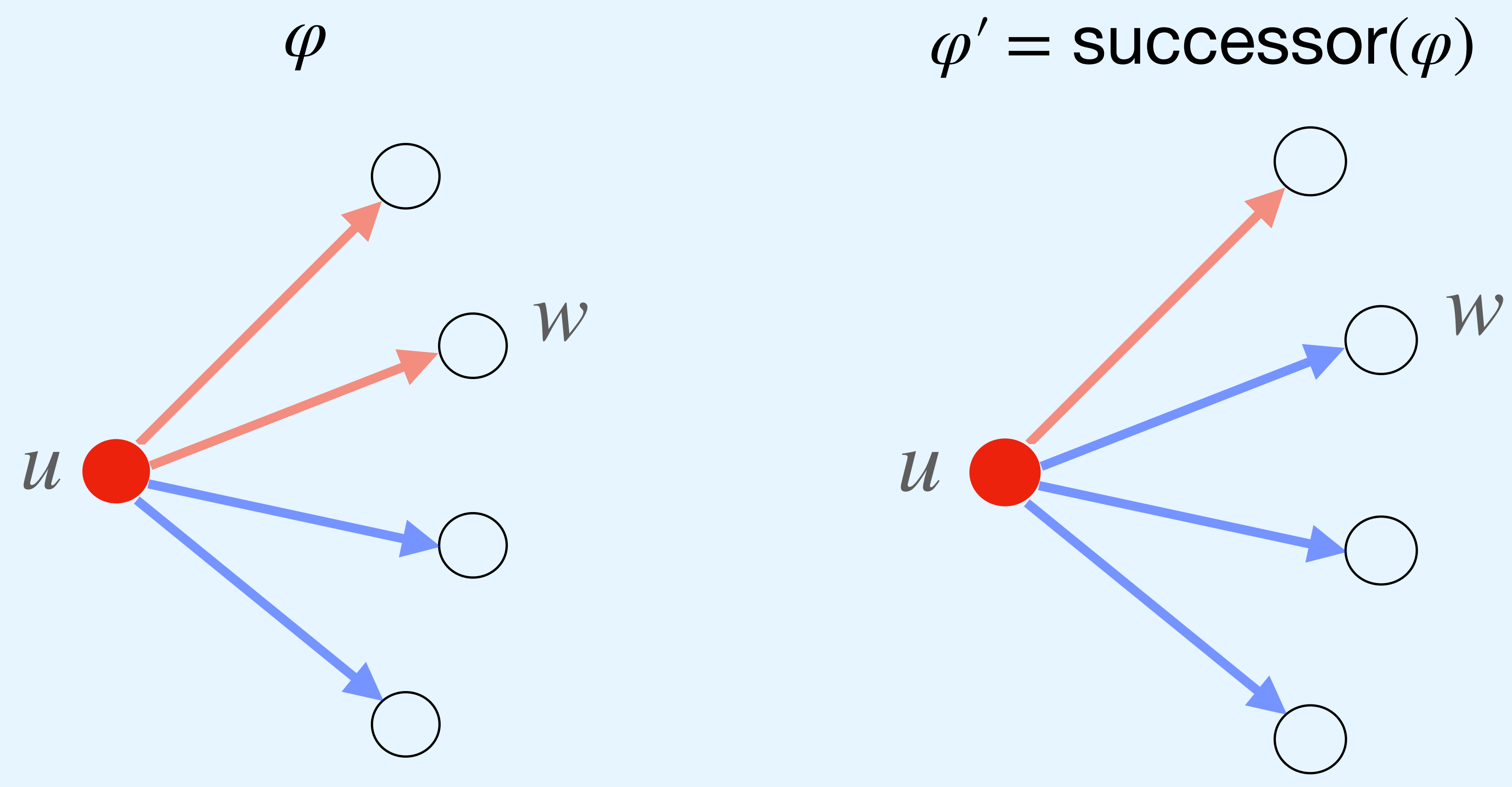
A well chosen φ_v → Build a sequence of failure patterns → **φ_v, $\varphi_\varnothing$ are in the same component C** → **No node dominating component C**

No oblivious algorithm solving consensus in R rounds

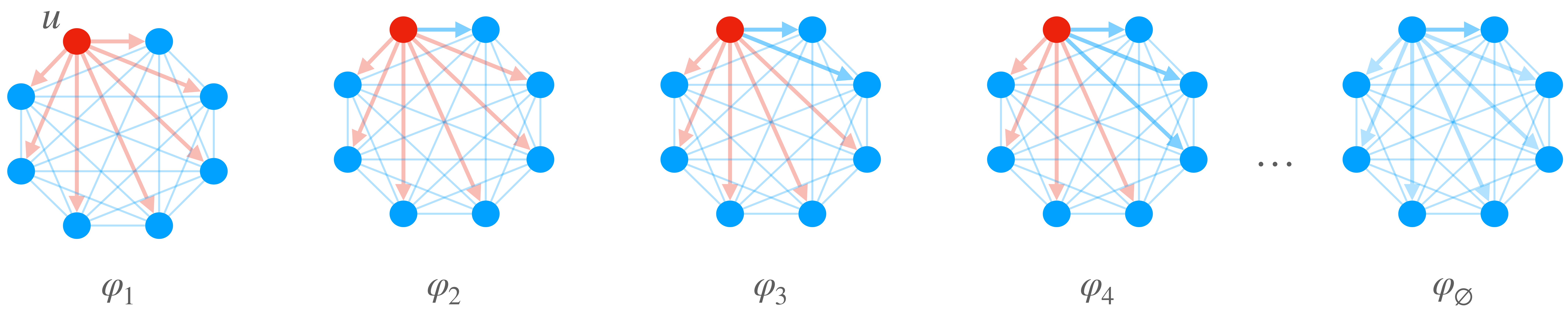


There is a node can not distinguish φ_i, φ_{i+1}.

Successor of a failure pattern



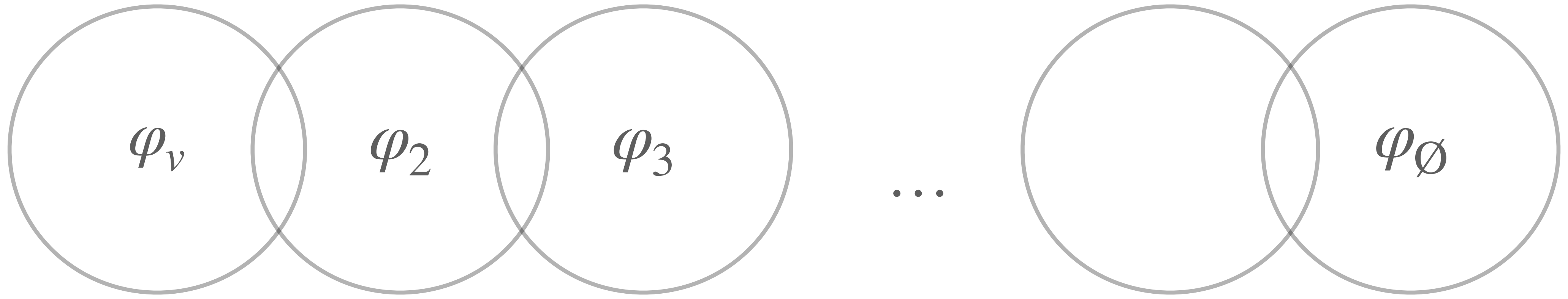
- Node u crashing last in φ.
- φ' is identical to φ but node u sends message to one more correct node w in φ' before crashing.
- There is a node with the same view in φ, φ'.



Outline of the proof

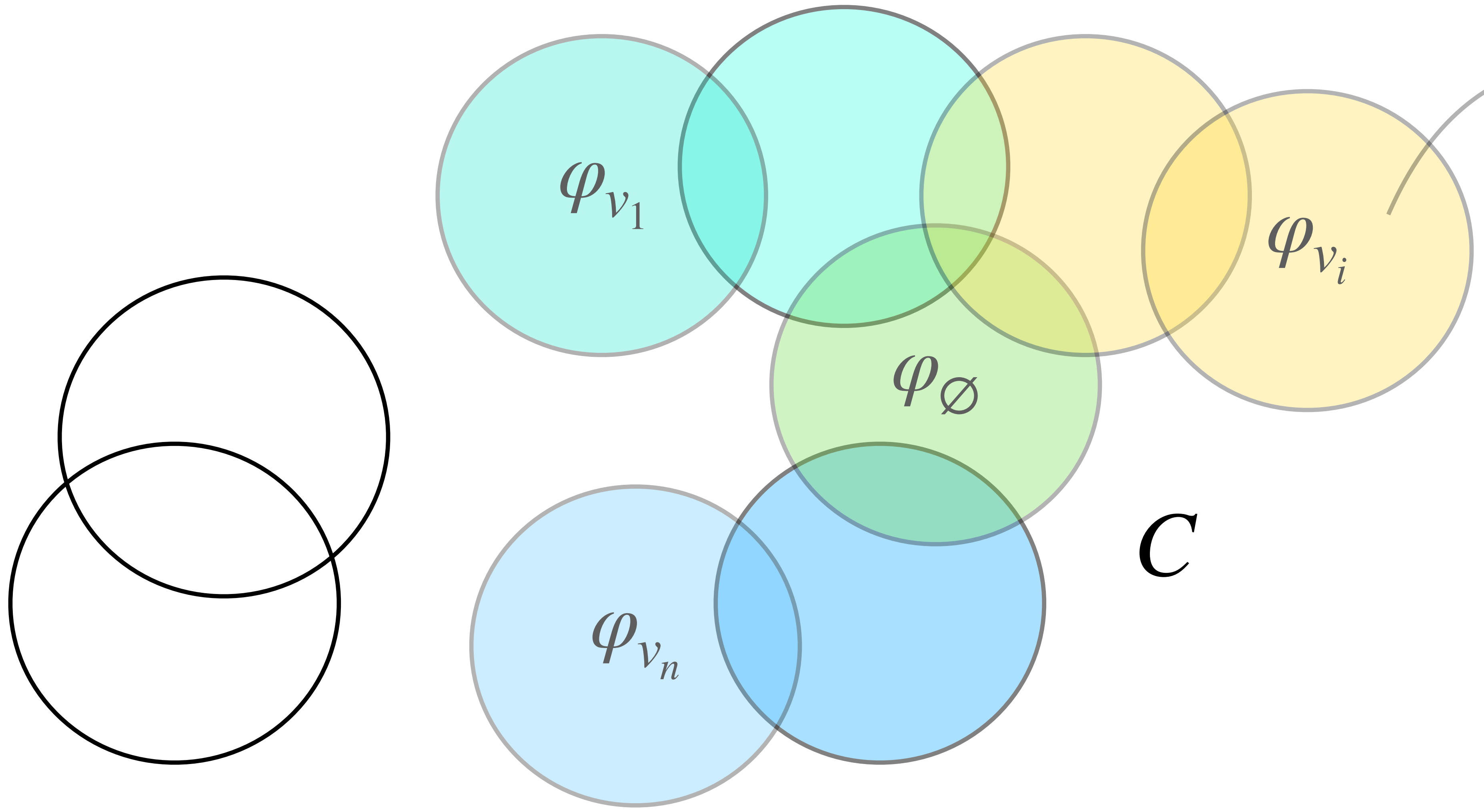
Theorem: *For every graph G and every $t < \kappa(G)$, consensus in G cannot be solved in less than $rad(G, t)$ rounds by an oblivious algorithm in the t-resilient model.*

$IF(G, R, \Phi_{\text{all}}), R < rad(G, t)$



φ_v : v can not broadcast in φ_v in R rounds.

$\varphi_{i+1} = succ(\varphi_i)$: There is a node can not distinguish φ_i, φ_{i+1}.



v_i does not dominate φ_{v_i}

Component C contains $\varphi_\varnothing, \varphi_{v_1}, \ldots, \varphi_{v_n}$.

There is no node dominating C.

Conclusion

Beyond the connectivity threshold: arbitrary t.

Local consensus: Consensus in each connected component of G after removing crashing nodes.

Conclusion

Beyond the connectivity threshold: Arbitrary t.

Local consensus: Consensus in each connected component of G after removing crashing nodes.

	Clique	*General graph*
Consensus	$(t+1)$ [1,2]	UB: $rad(G,t)$ [3] LB: $rad(G,t)$ [4]
K-set agreement	$\lfloor \frac{t}{k} \rfloor + 1$ [5]	Open

[1] *Dolev, Strong'83*

[2] *Aguilera, Toueg'99*

[3] *Castaneda, Fraigniaud, Paz, Rajsbaum, Roy, Travers'23*

[4] *Fraigniaud, N., Paz,'24*

[5] *Chaudhuri, Herlihy, Lynch, Tuttle,'00*

Q: *Can we do better with non-oblivious consensus algorithms?*

Q: *k-set agreement in general graph?*

Thank you!