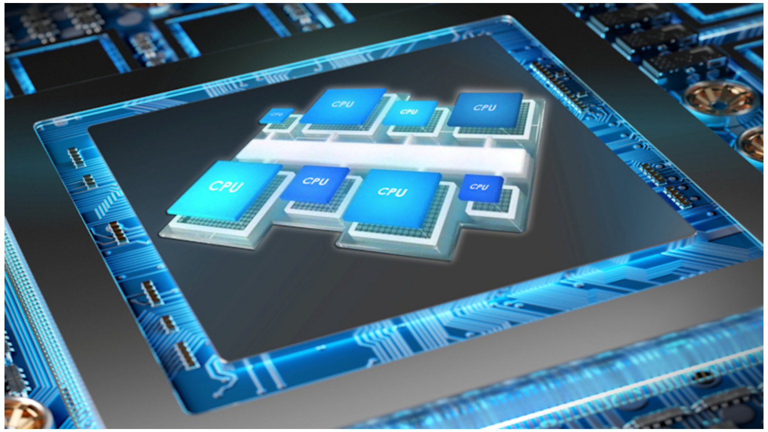
Distributed Network Computing through the Lens of Algebraic Topology (DUCAT)

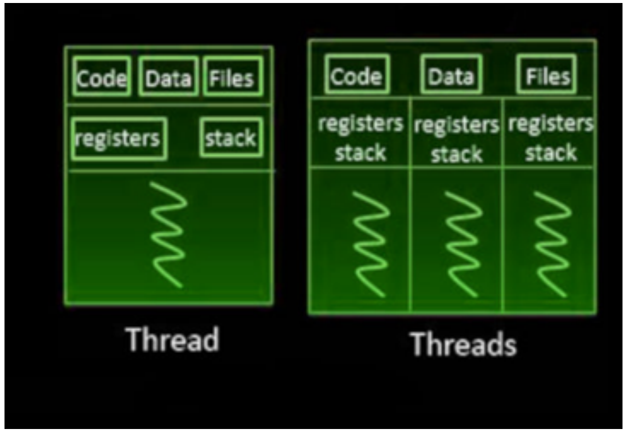
Pierre Fraigniaud

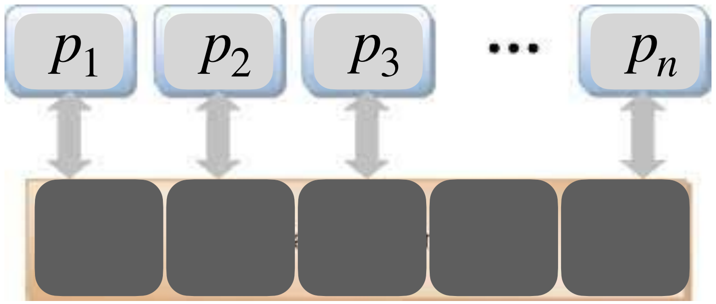
IRIF
Université Paris Cité and CNRS

DUCAT-ESTATE Workshop
Cap Hornu, March 16-18, 2022

Shared Memory Model







Single Writer / Multiple Reader registers

Wait-Free Computing

Code of process $i \in \{1,\dots,n\}$ with input x_i

$V_i \leftarrow x_i$

For $r = 1$ to t do

  write(V_i) in register $M[i]$

  for $j = 1$ to n do $v_j \leftarrow$ read$(M[j])$

  $V_i \leftarrow (v_1, v_2, \dots, v_n)$

decide $y_i = f(V_i)$

Atomic operations

Full Information protocol

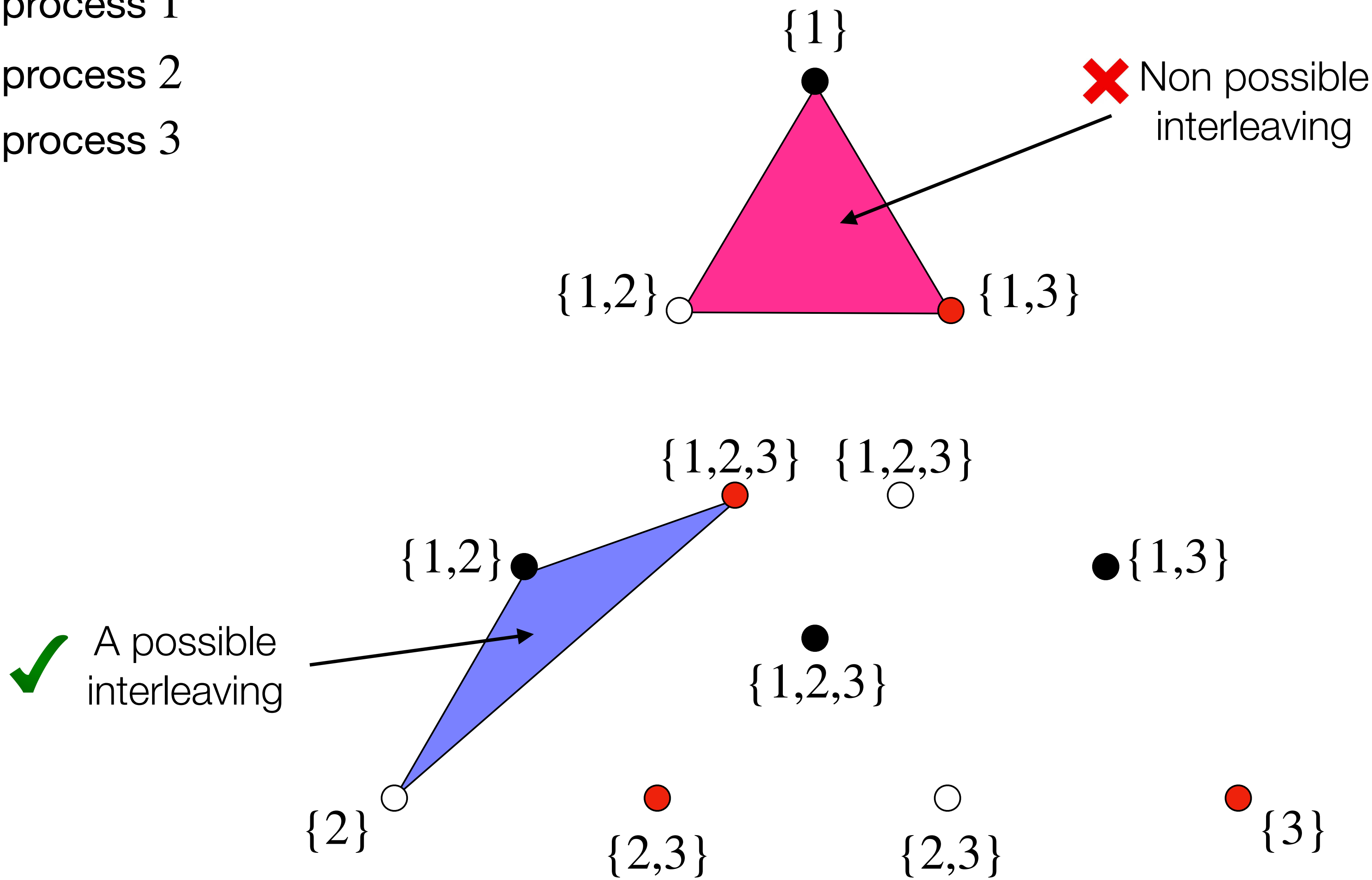
Decision function

Read/Write Interleaving

Assume $n = 3$

Read/Write Interleaving

- ● process 1
- ○ process 2
- ● process 3

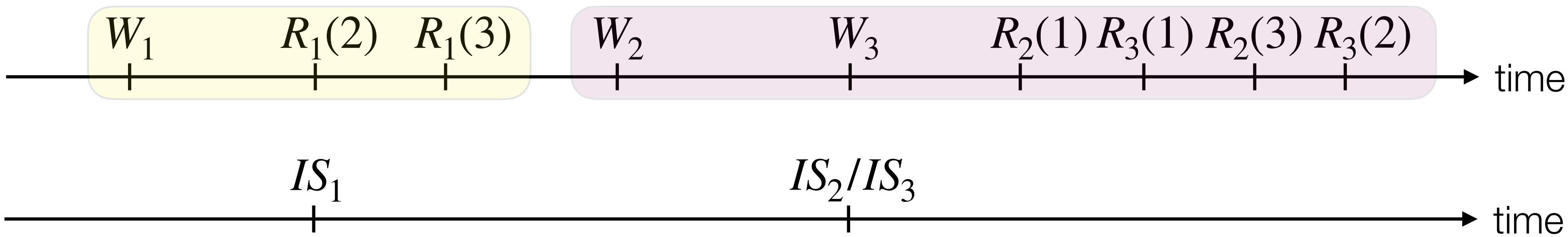


French Grammar

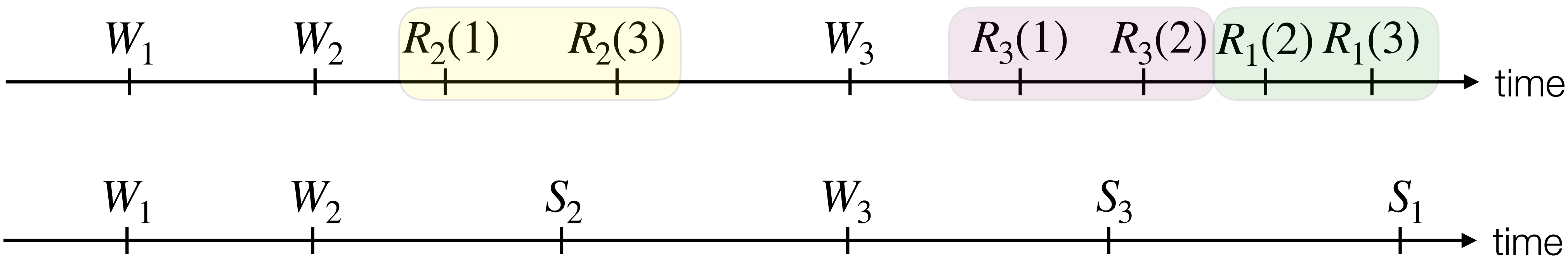
- First Group: all verbs with infinitive form **-er** (but *aller*).
- Second group: all verbs with infinitive form **-ir** and gerundive form **-issant**.
- Third group: all the rest!

Snapshots and Immediate Snapshots

IMMEDIATE SNAPSHOTS

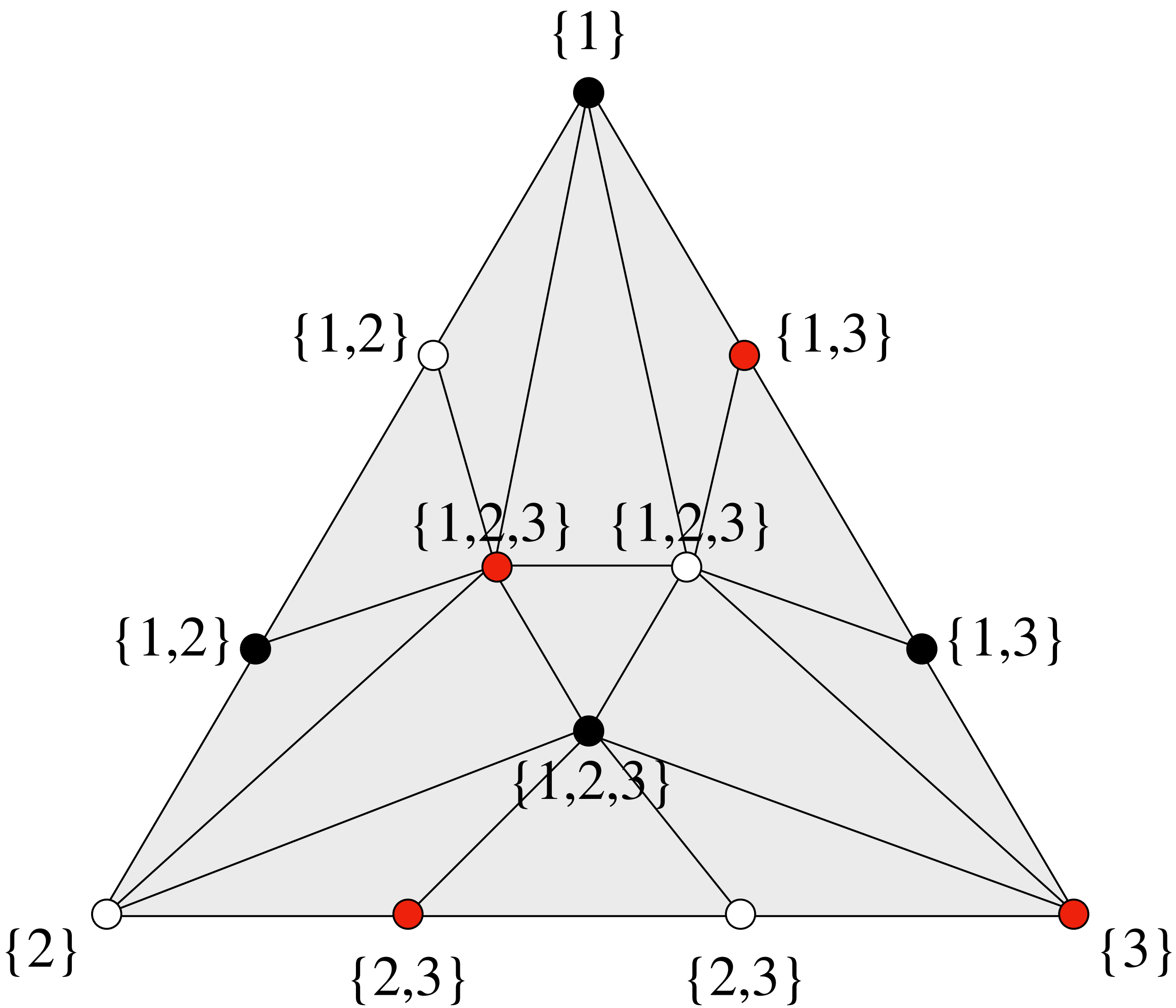


SNAPSHOTS



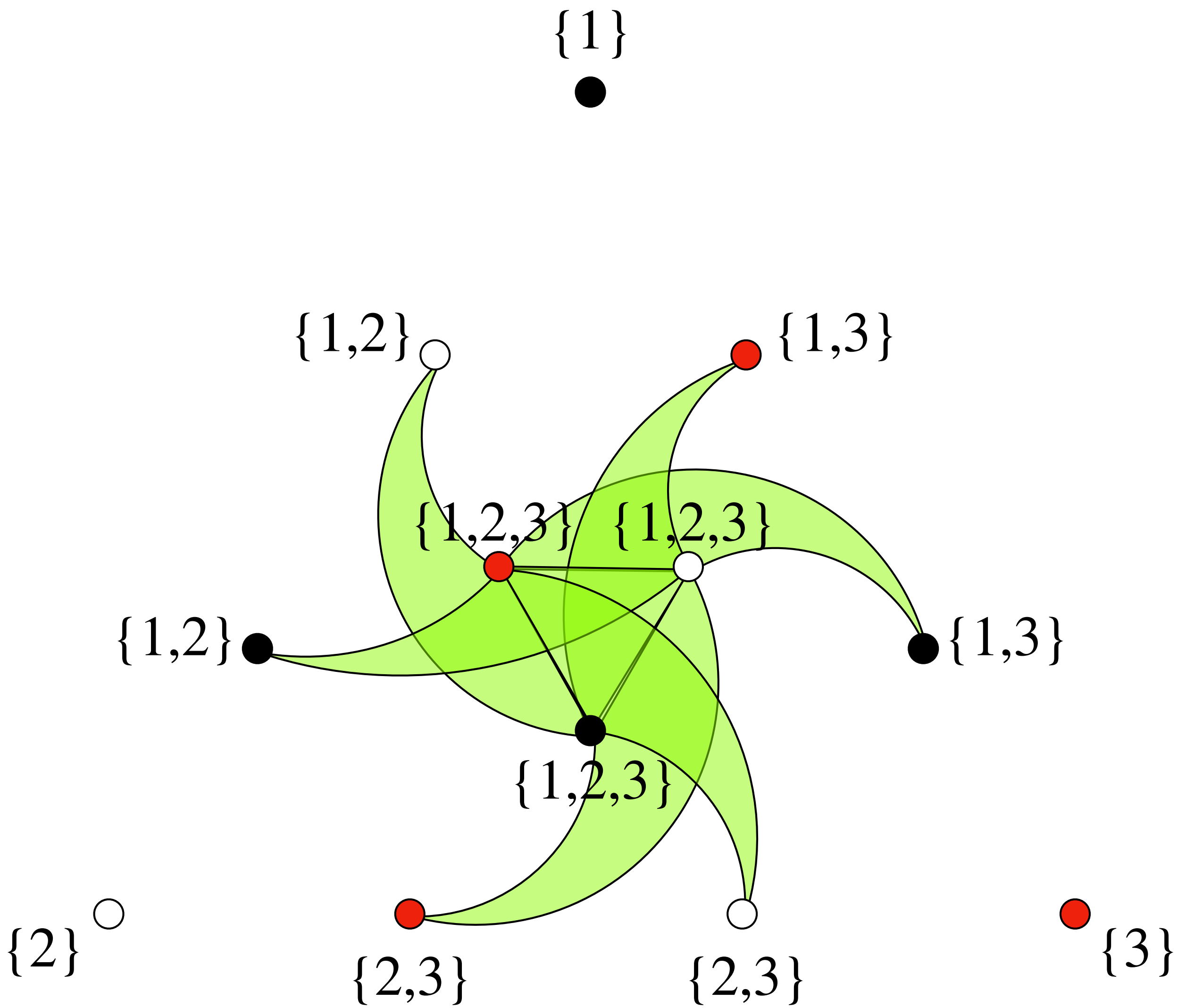
Immediate Snapshots

- ● process 1
- ○ process 2
- ● process 3

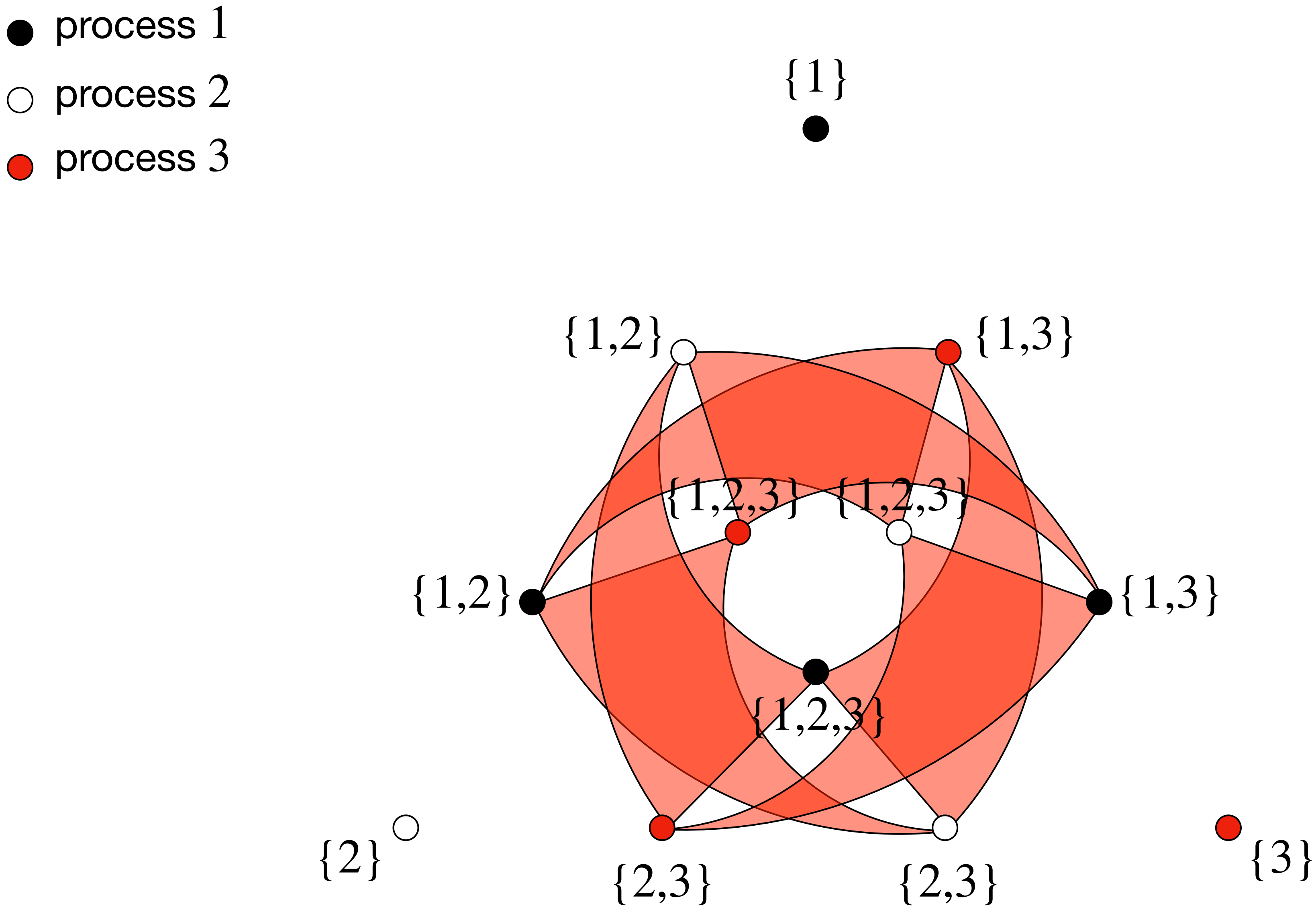


(Non-Immediate) Snapshots

- process 1
- process 2
- process 3



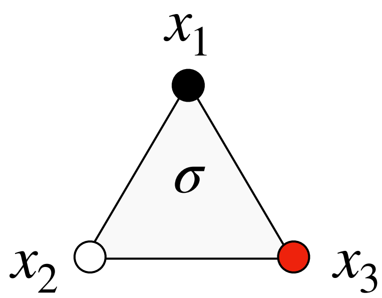
The Rest…



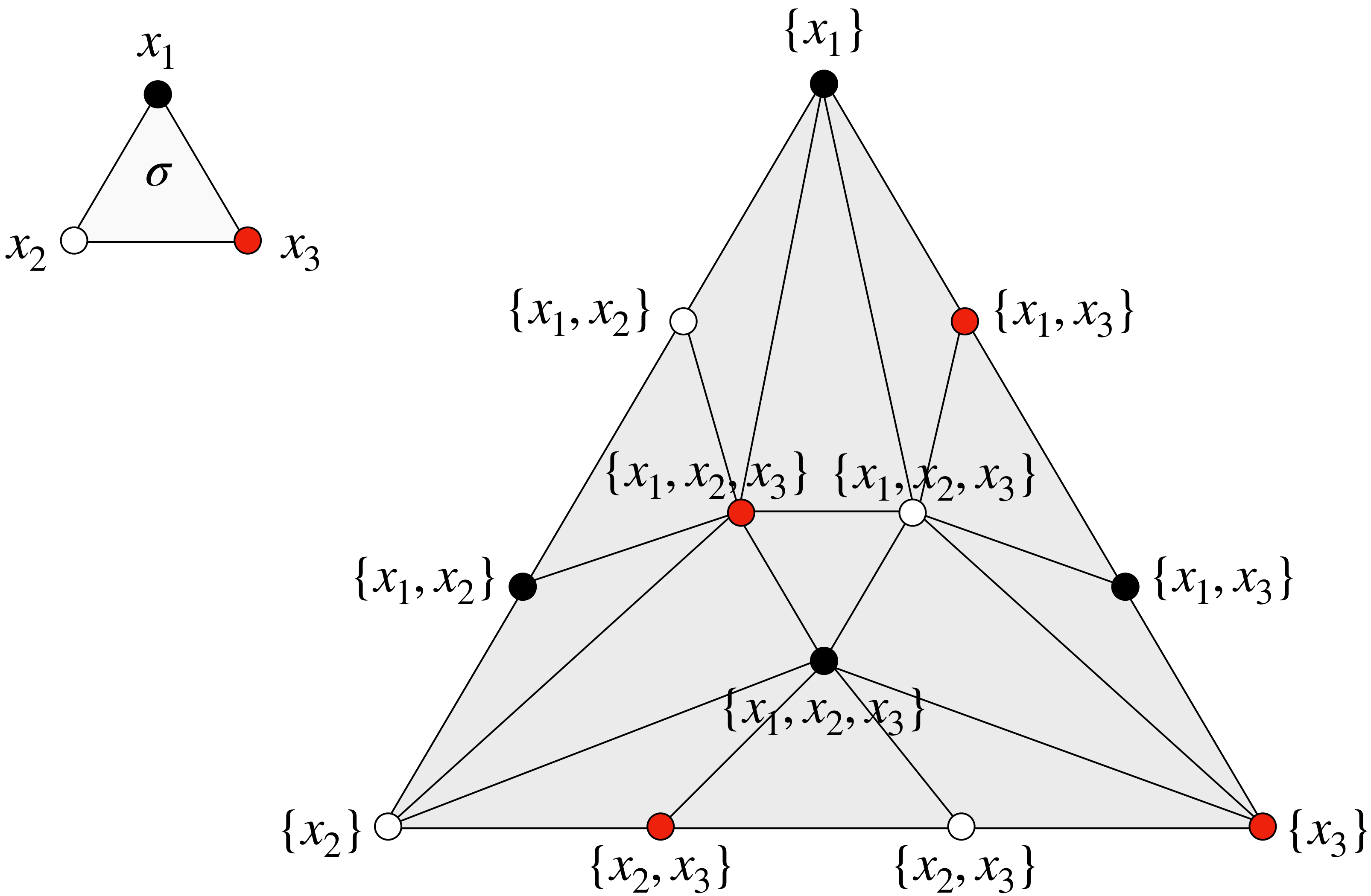
System Configuration

Configuration $\sigma = \{p_1$ in state x_1, p_2 in state x_2, p_3 in state $x_3\}$

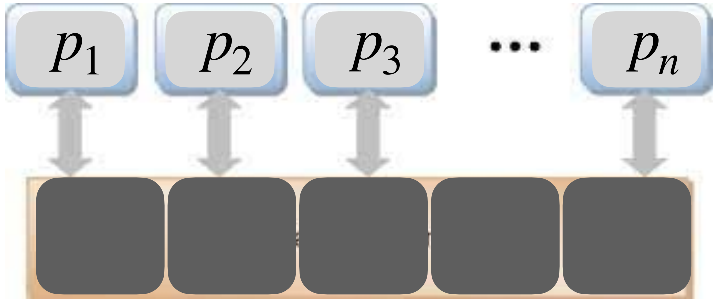
- process 1
- process 2
- process 3

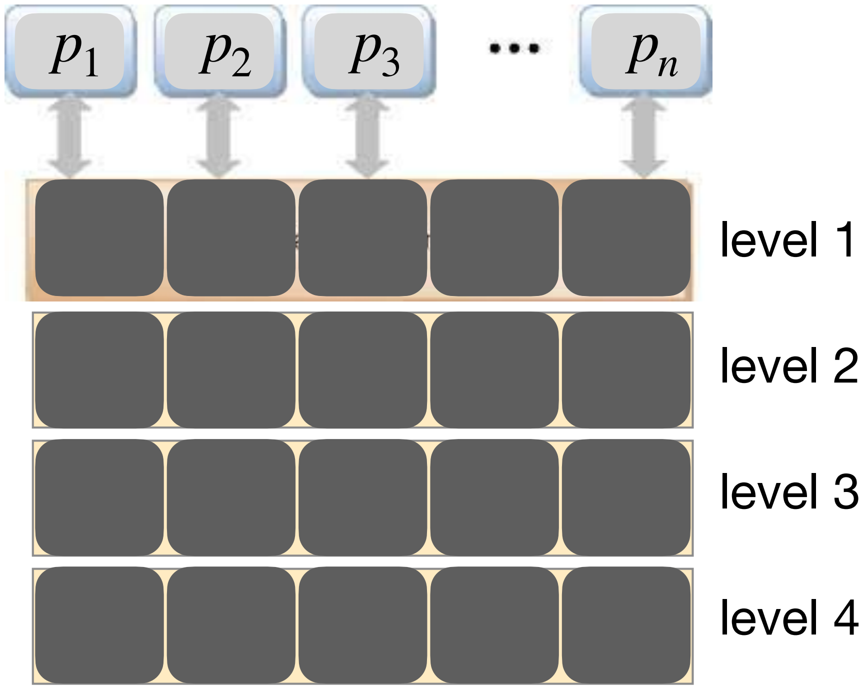


One *Round* Starting from σ



Iterated Model





For every $i = 1,2,\ldots$ the i-th write of each process, as well as all the $n-1$ reads performed after that write are performed in the i-th level of the memory.

Iterated Wait-Free Computing

Code of process $i \in \{1,\ldots,n\}$ with input x_i

$V_i \leftarrow x_i$

For $r = 1$ to t do

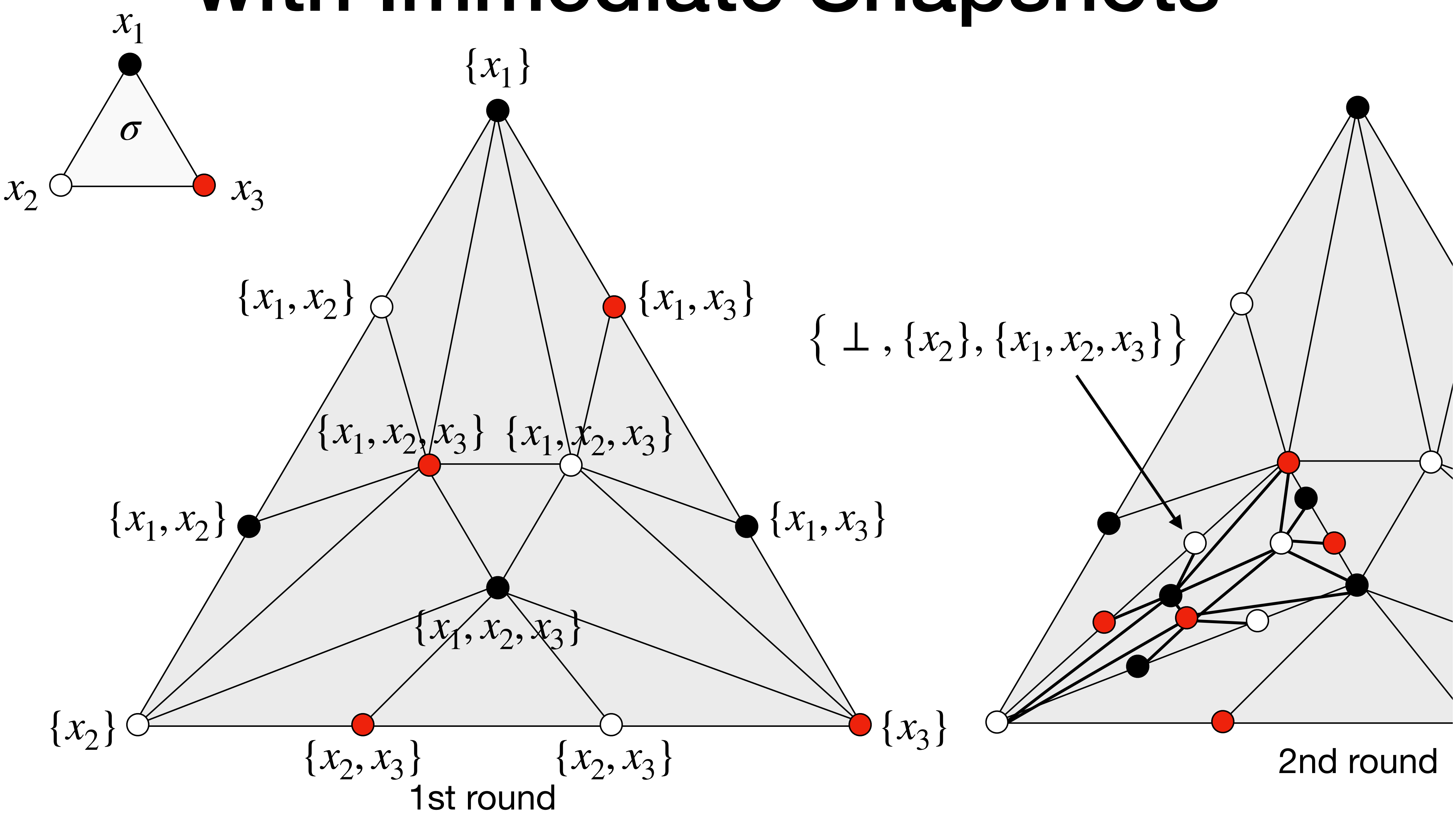
write(V_i) in register $M_r[i]$

for $j = 1$ to n do $v_j \leftarrow$ read($M_r[j]$)

$V_i \leftarrow (v_1, v_2, \ldots, v_n)$

decide $y_i = f(V_i)$

Multi-Round Computation with Immediate Snapshots



Simplicial Complex

- A (simplicial) complex $\mathscr{K}$ over a set V is a collection of non-empty subsets of V closed by inclusion

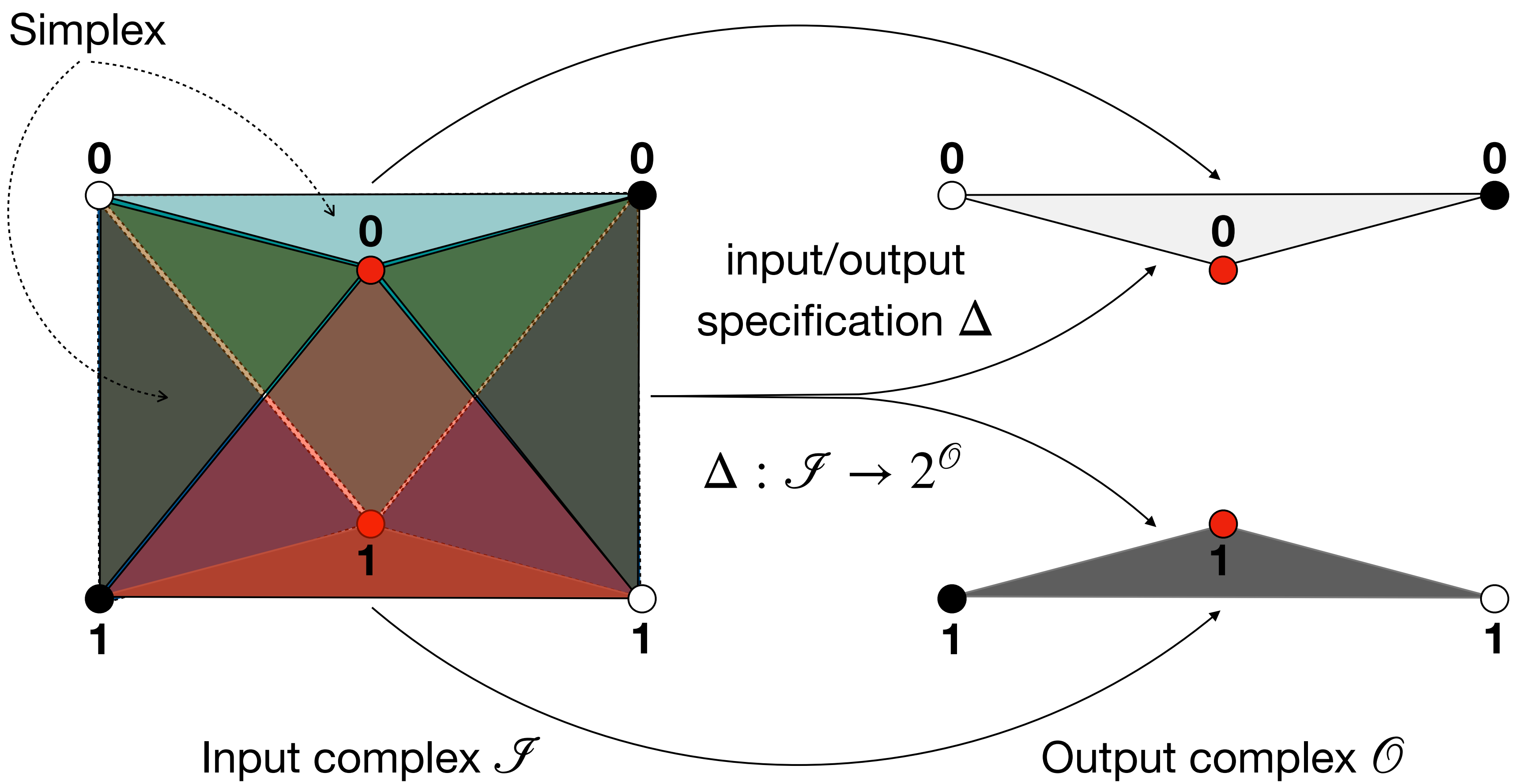
$$(\sigma \in \mathscr{K} \text{ and } \emptyset \neq \tau \subseteq \sigma) \implies \tau \in \mathscr{K}$$

- Any set $\sigma \in \mathscr{K}$ is called a simplex.

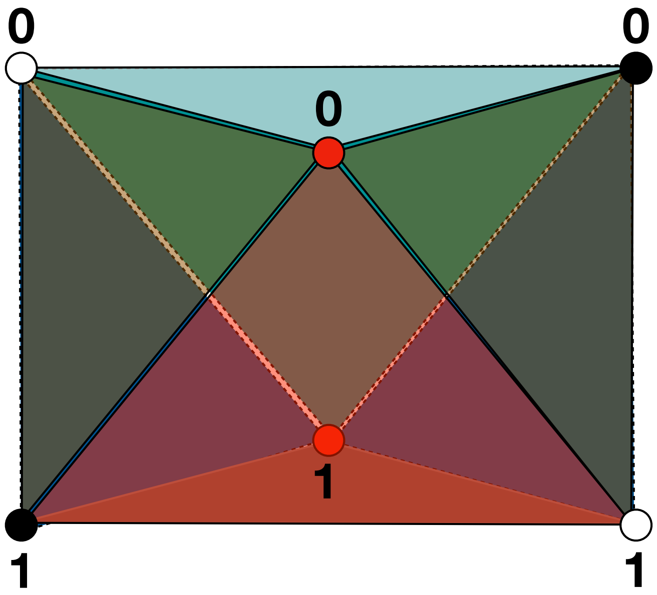
- Any element of V that is in $\mathscr{K}$ is called a vertex.

- Example: A graph $G = (V, E)$ is the complex over V with simplices $V \cup E$

Task $\Pi = (\mathscr{I}, \mathscr{O}, \Delta)$

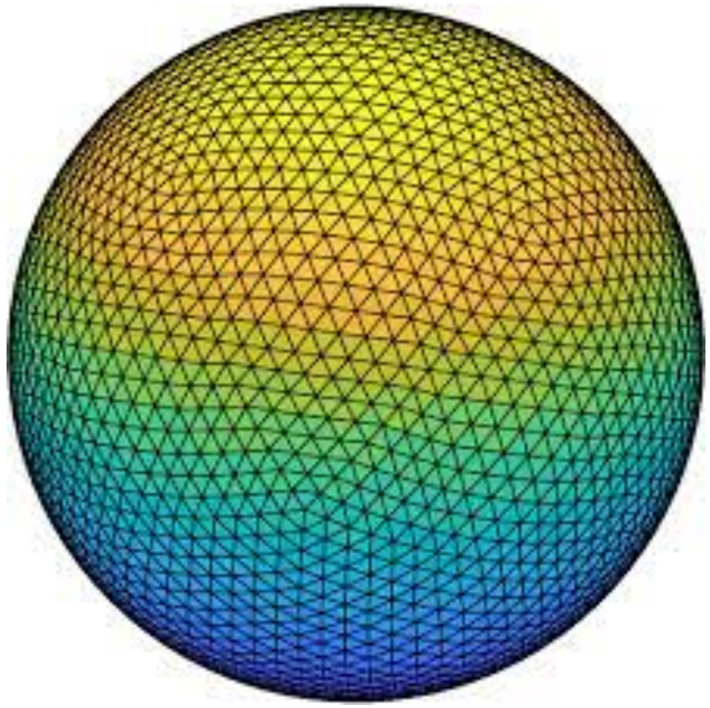


Protocol Complex



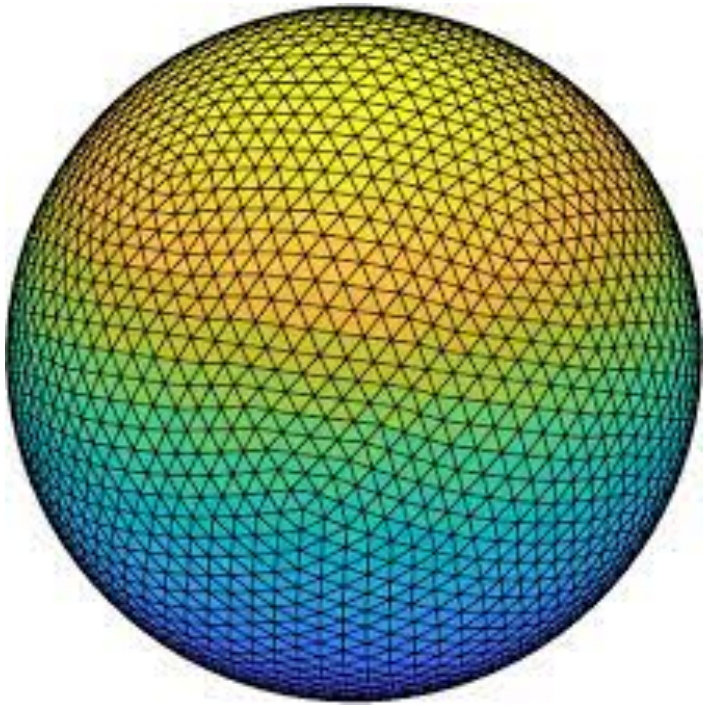
Input complex $\mathscr{I}$

t rounds →

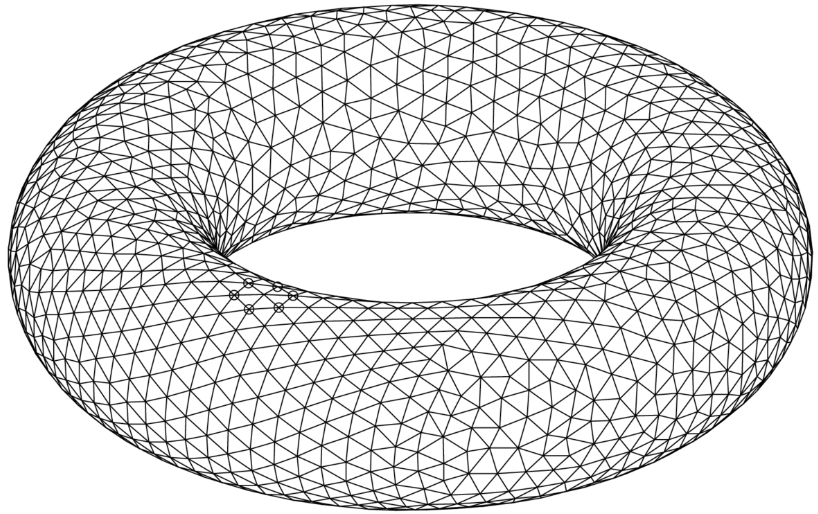


Protocol complex $\mathscr{P}^{(t)}$

Output Computation



decision map f →



Protocol complex $\mathscr{P}^{(t)}$

Output complex $\mathscr{O}$

A simplicial map from $\mathscr{K}$ to $\mathscr{K}'$ is a function $f : V(\mathscr{K}) \rightarrow V(\mathscr{K}')$ such that, for every $\sigma \in \mathscr{K}, f(\sigma) \in \mathscr{K}'$.

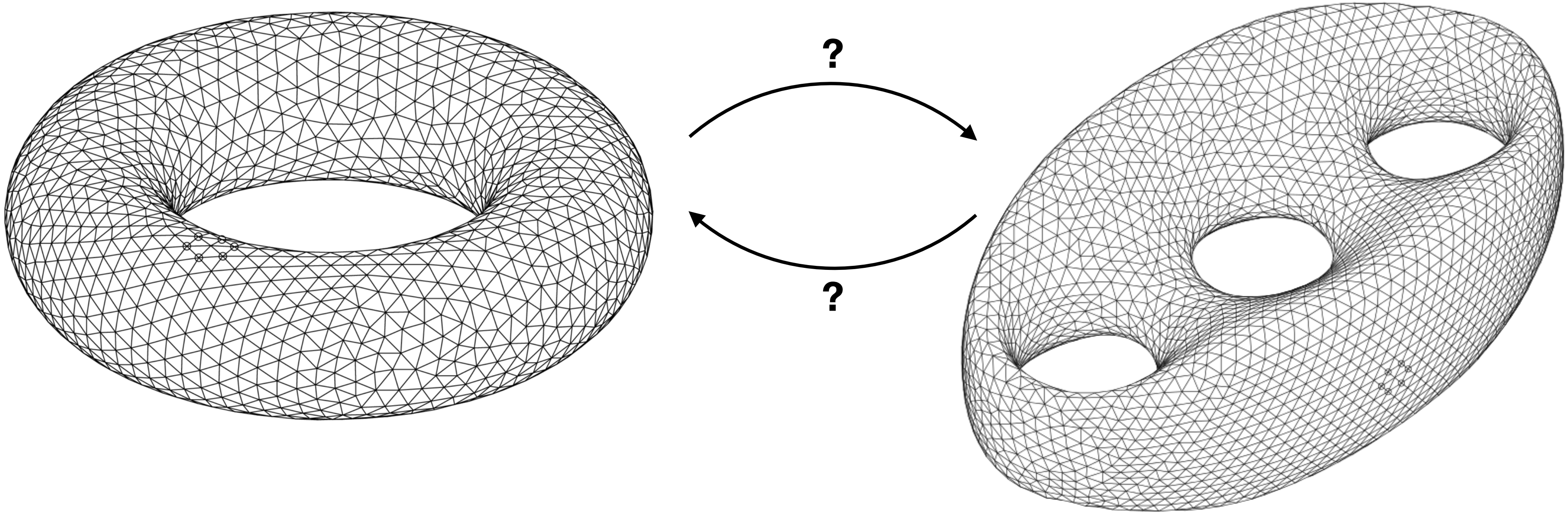
The decision map is a simplicial map from $\mathscr{P}^{(t)}$ to $\mathscr{O}$ that is chromatic (it preserves the IDs of the processes), and agrees with Δ, i.e., $\forall \sigma \in \mathscr{I}$, $f(\mathscr{P}(\sigma)) \subseteq \Delta(\sigma)$.

Wait-Free Solvability

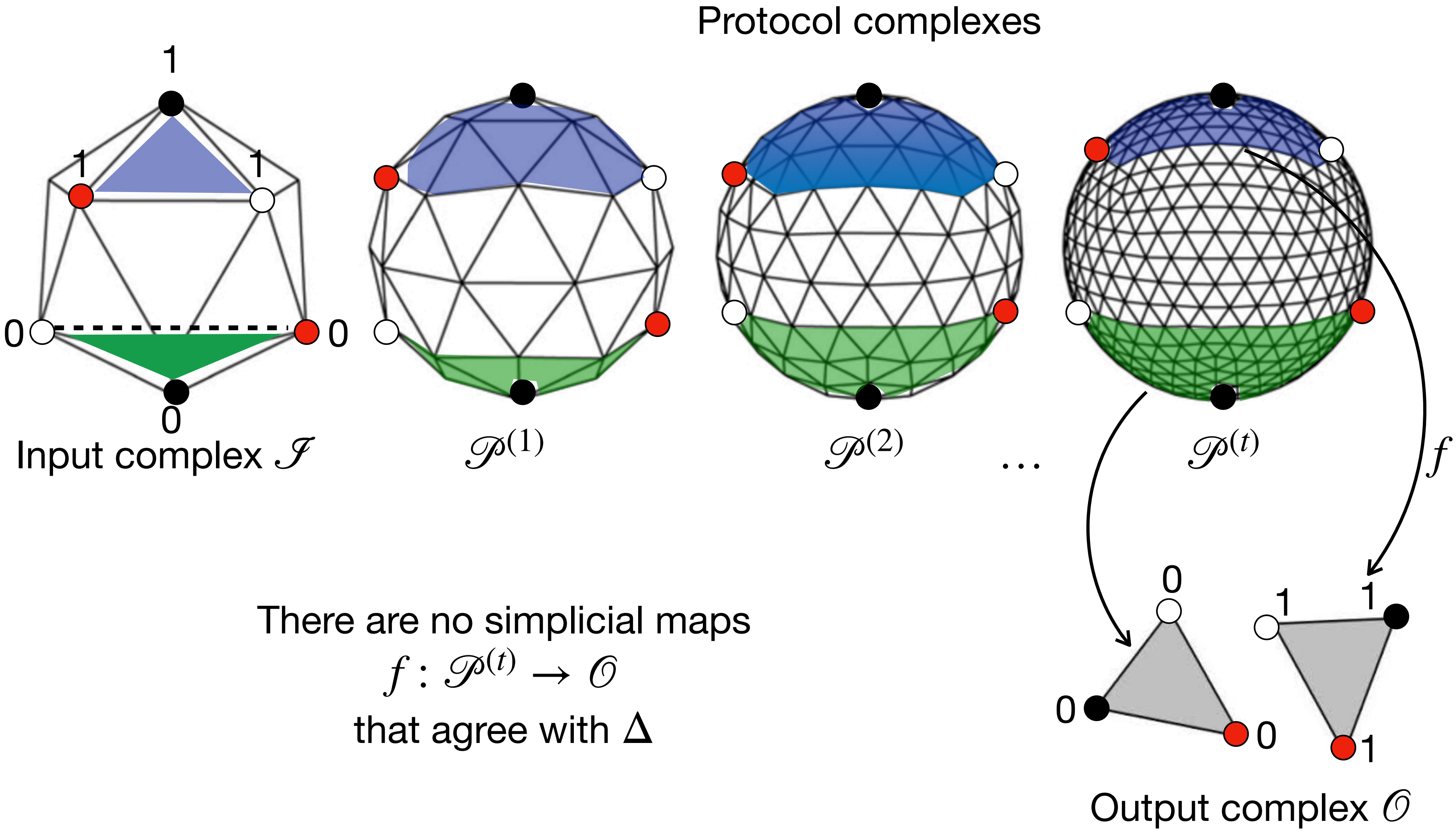
Theorem [Herlihy, Shavit (1999)]

A task $(\mathcal{I}, \mathcal{O}, \Delta)$ is solvable wait-free if and only if there is a simplicial map $f : \mathcal{P} \to \mathcal{O}$ from a chromatic subdivision $\mathcal{P}$ of $\mathcal{I}$ to $\mathcal{O}$ that agrees with Δ.

Topology



Impossibility of Consensus



There are no simplicial maps
$f : \mathscr{P}^{(t)} \to \mathscr{O}$
that agree with Δ

Beyond Wait-Free

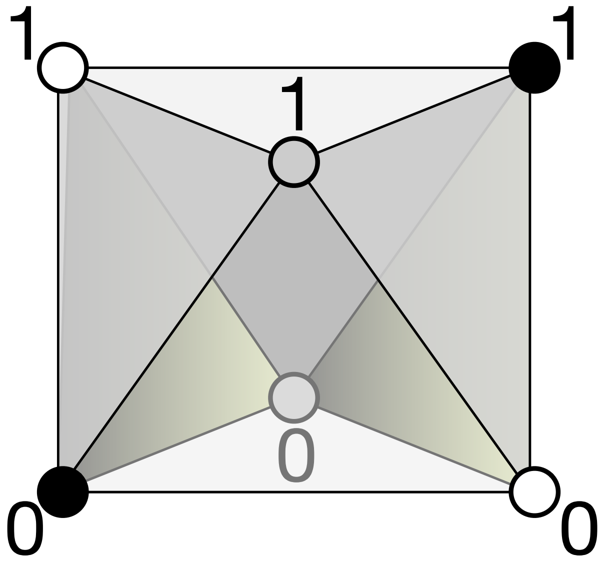
- Other kinds of adversarial models (e.g., t-resilient)
- Stronger forms of failures (e.g., Byzantine)
- Message-passing

DUCAT: Extension of the theory to network computing

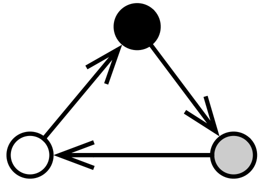
Protocol Complex

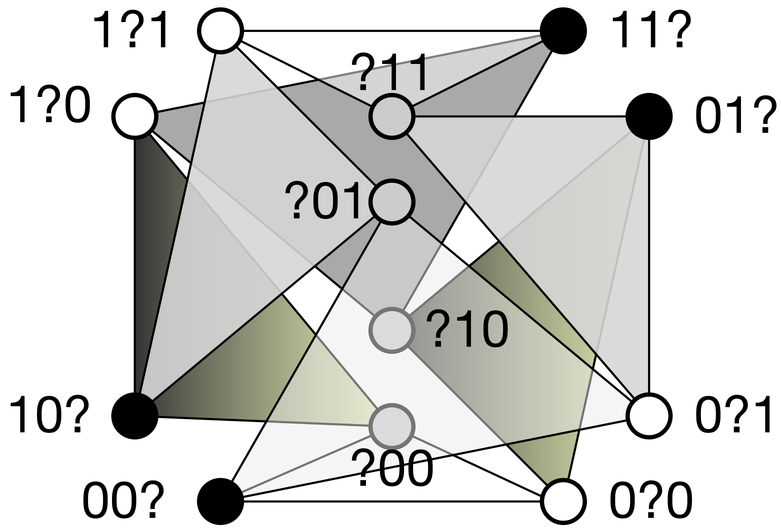
Example 1

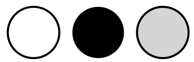
Input complex



Protocol complex for C_3



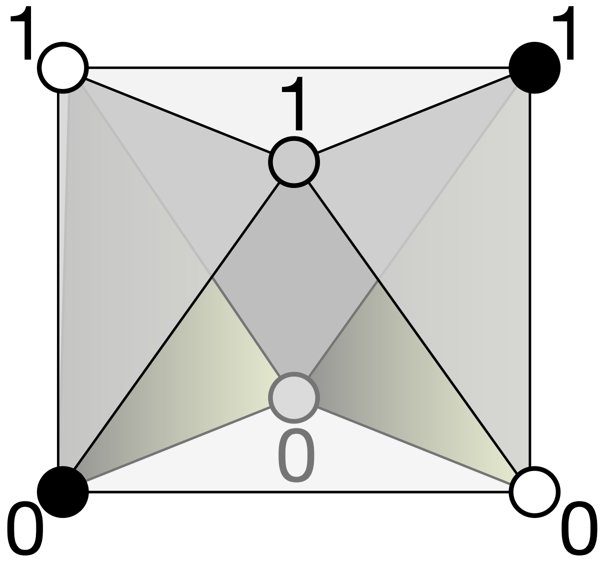




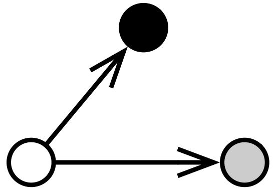
Protocol Complex

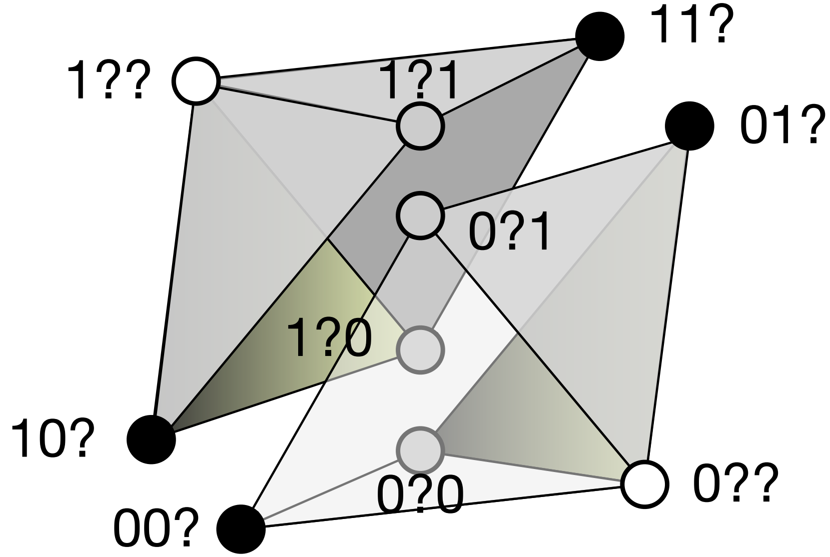
Example 2

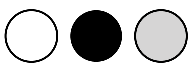
Input complex



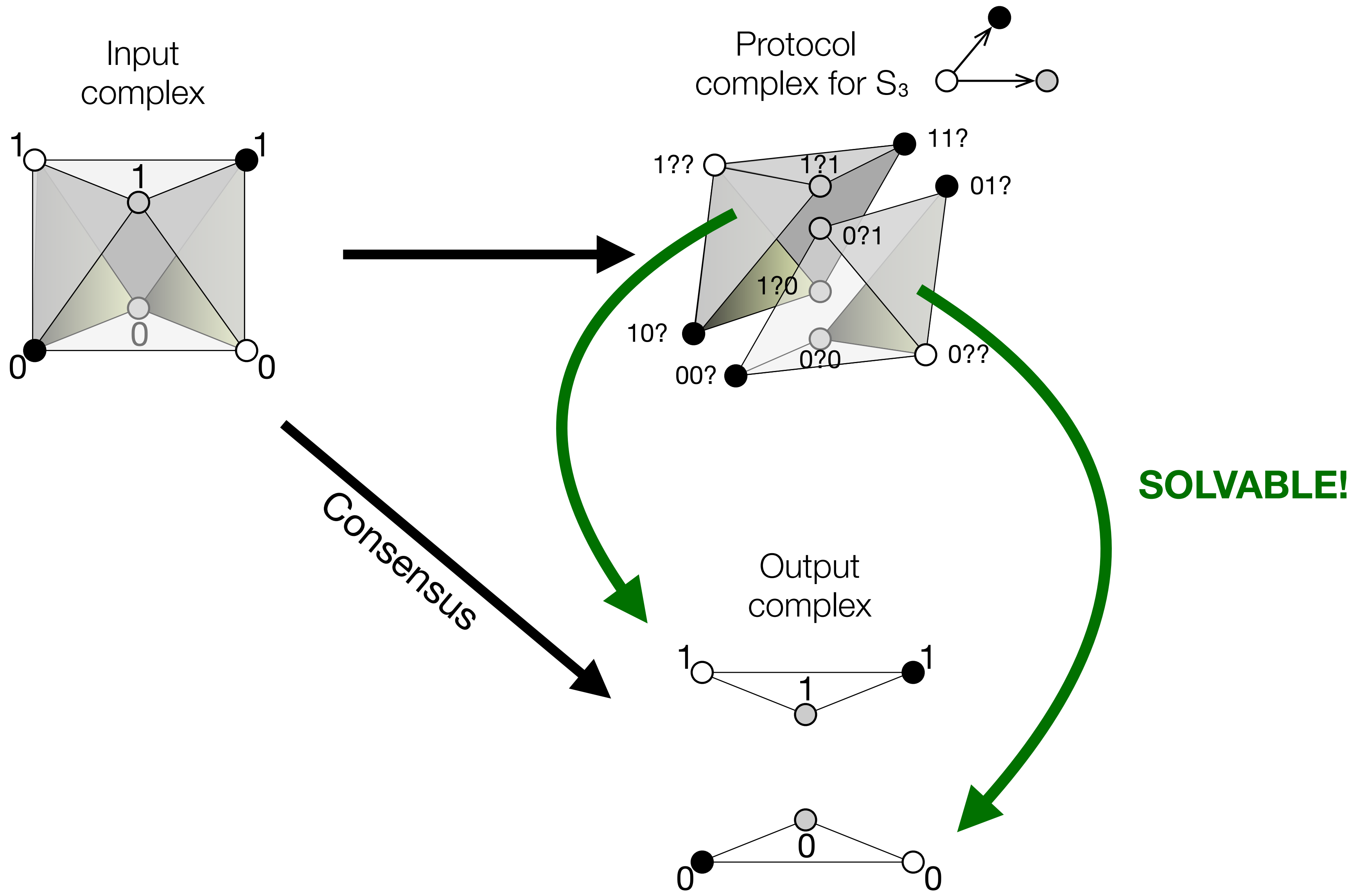
Protocol complex for S_3



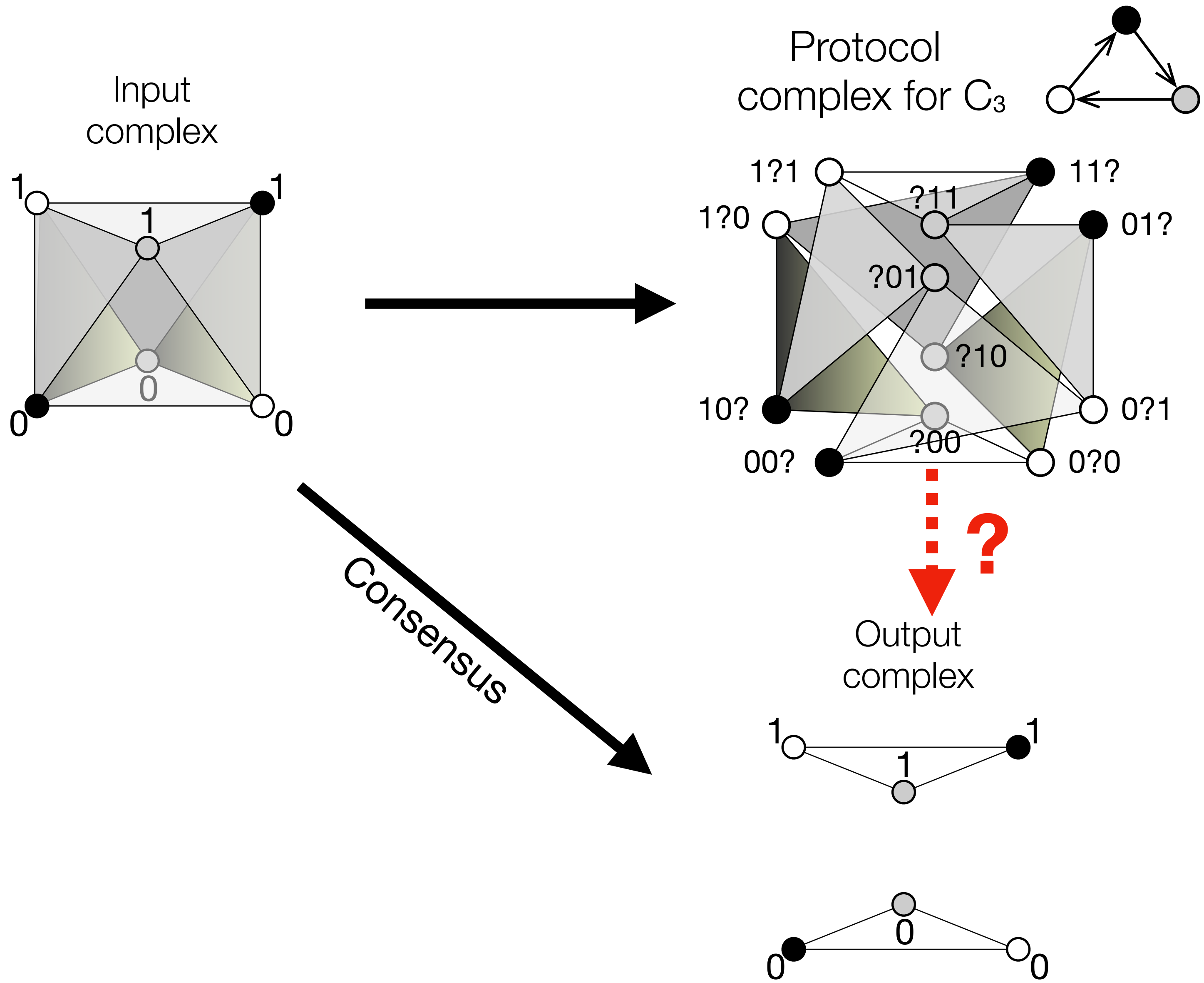




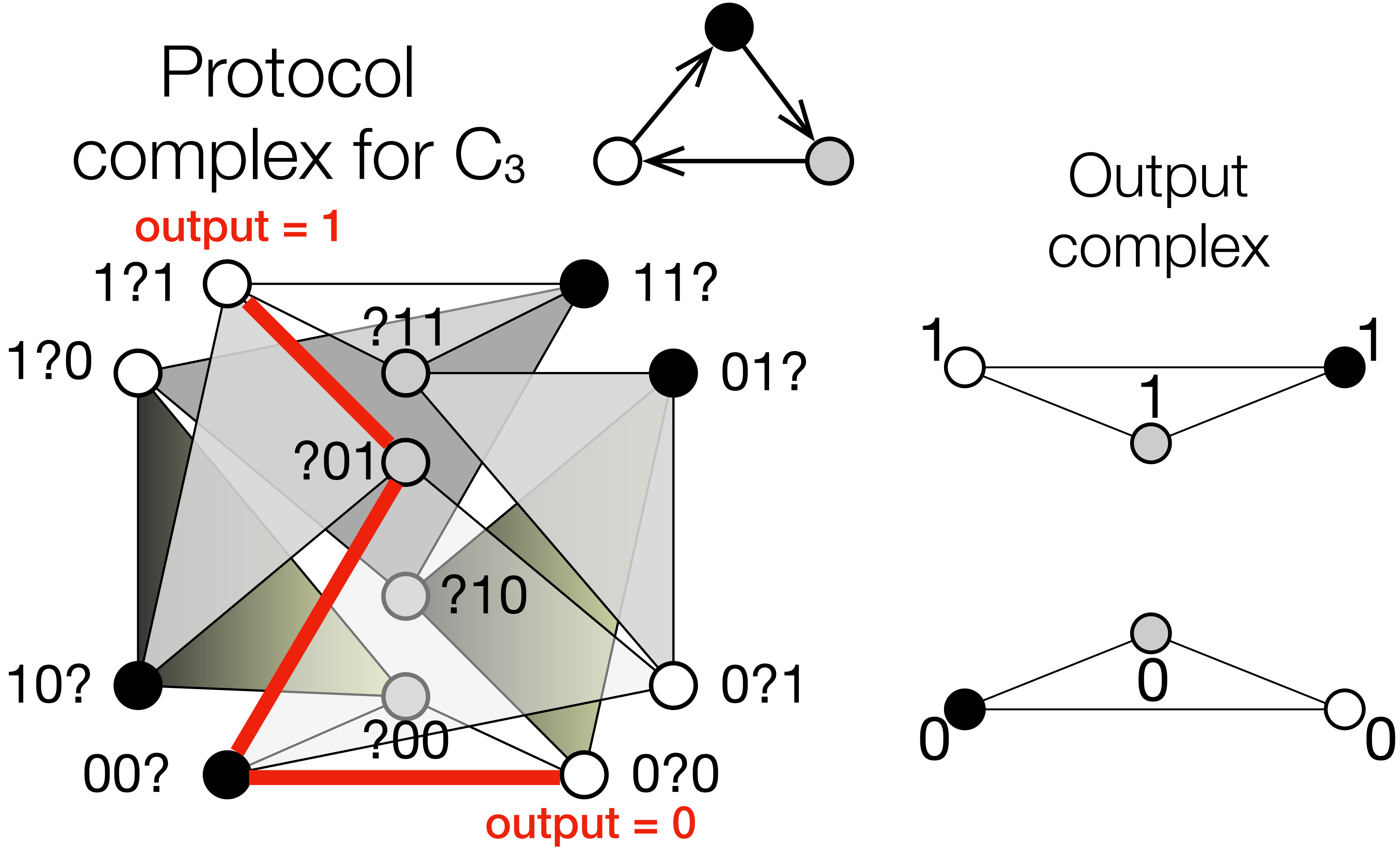
Consensus Solvability 1



Consensus Solvability 2



Path-Connectivity



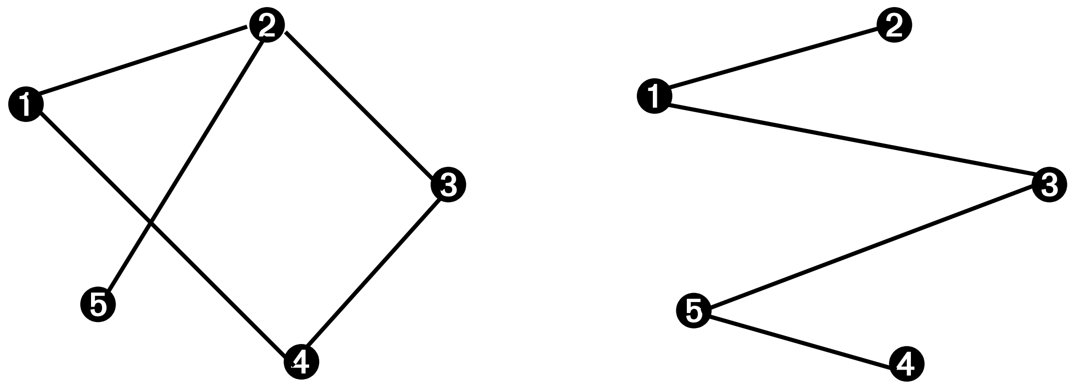
LOCAL Model

LOCAL model: synchronous rounds in a fixed graph G, no failures

Theorem For any $k \geq 1$, k-set-agreement in network G requires at least r rounds, where r is the smallest integer such that $\gamma(G^r) \leq k$.

Dynamic Networks

DYNAMIC networks: synchronous, no failure; A sequence of labeled digraphs $\mathscr{G} = (G_t)_{t \geq 1}$



Round 1: G_1 Round 2: G_2 ...

Corollary 2 For any $k \geq 1$, k-set-agreement in dynamic network $\mathscr{G} = (G_t)_{t \geq 1}$ requires at least r rounds, where r is the smallest integer such that $\mathscr{G}$ has *temporal* domination number $\leq k$

DUCAT

Distributed Network Computing through the Lens of Combinatorial Topology

- IRIF (CNRS and Université de Paris) — Pierre Fraigniaud
- LIS (CNRS and Aix-Marseille University) — Jérémie Chalopin

Project starts: March 15, 2021

Context and Objectives

- **Algorithms design and analysis:** establishing lower bounds or impossibility results is extremely difficult.
- **Combinatorial topology:** extensively used in the context of crash-prone asynchronous shared-memory (or message-passing).
- **Objective of DUCAT:** Extending these results to other models
  - Network computing
  - Dynamic networks
  - Beyond full-information protocols

Expected Outcomes

1. Complexity results: New lower bounds, but also new upper bounds

2. Better understanding of the nature of distributed computing

3. Unified framework for distributed computing

Thank you!