

Ring Algorithm

Group 9

CSE-A

February 11, 2026

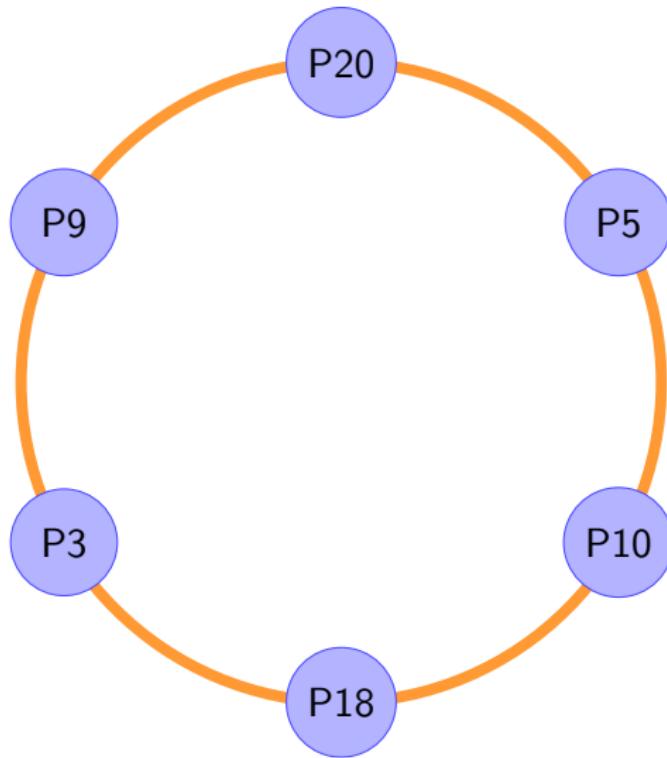
Election Algorithm Overview

- An **election algorithm** is used to choose a unique process to act as a **coordinator**.
- All processes in the system must **agree on the selected coordinator**.
- If the current coordinator retires or fails, a **new election** is initiated.
- A process that initiates an election is said to **call the election**.
- At any time, a process can be:
 - **Participant** – currently involved in an election
 - **Non-participant** – not involved in an election
- Two common election algorithms are:
 - Ring-based Election Algorithm
 - Bully Algorithm

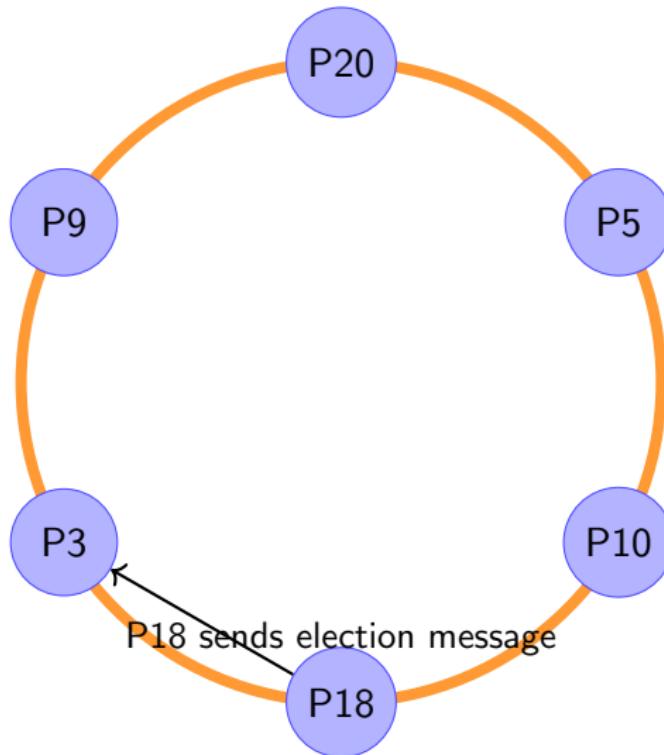
Ring-Based Election Algorithm

- Processes are arranged in a **logical ring**.
- Each process P_i communicates only with $P_{(i+1) \bmod N}$.
- All messages move **clockwise** around the ring.
- Initially, all processes are marked as **non-participants**.
- Any process can start an election by:
 - Marking itself as a participant
 - Sending an **election message** containing its identifier to its neighbour
- When a process receives an election message:
 - If received ID is **greater**, forward the message
 - If received ID is **smaller** and receiver is not a participant, replace it with its own ID and forward
 - If already a participant, it does not forward the message
- If a process receives its **own ID**, it becomes the **coordinator**.
- The coordinator sends an **elected message** around the ring to announce itself.

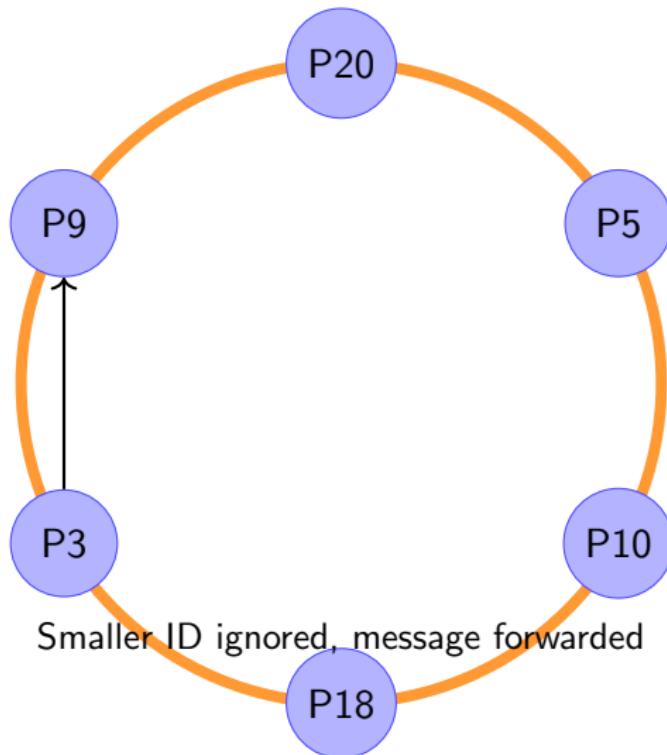
Ring Algorithm – Step 1: Initial State



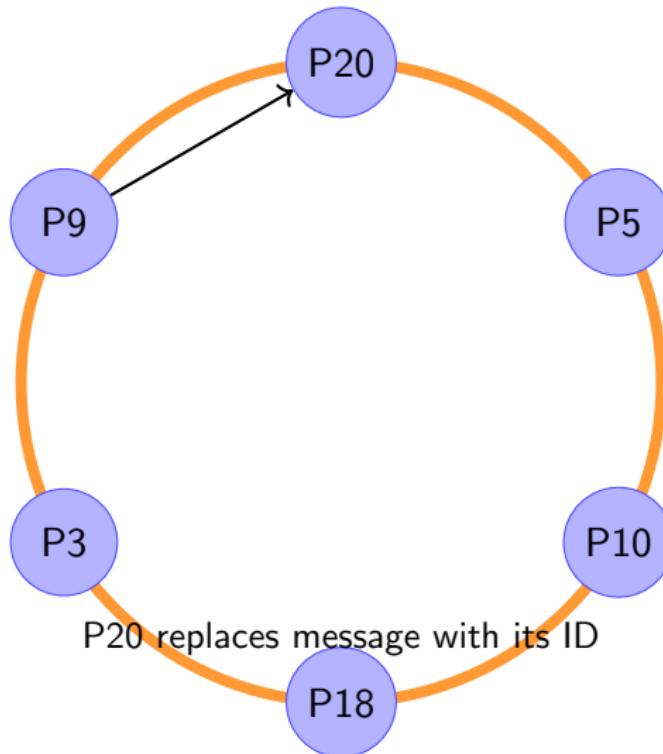
Ring Algorithm – Step 2: Election Initiation



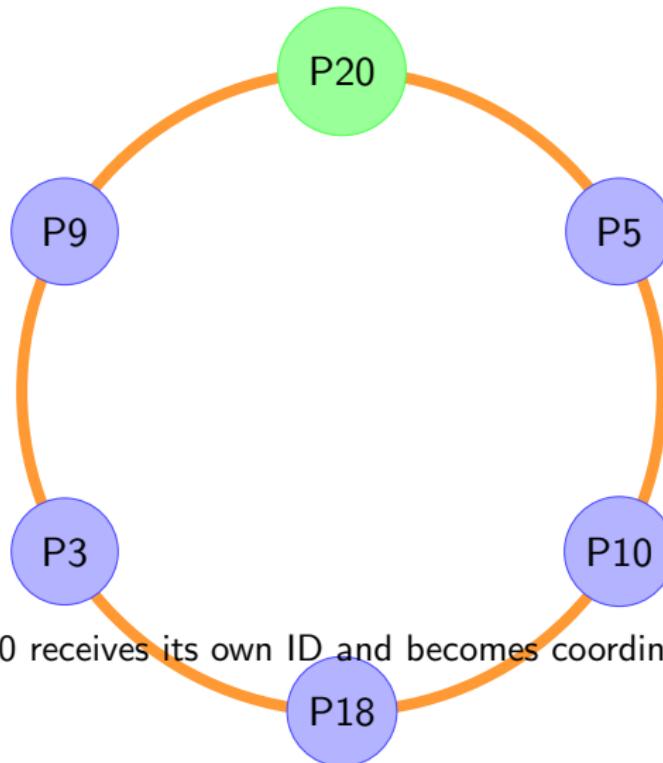
Ring Algorithm – Step 3: ID Comparison



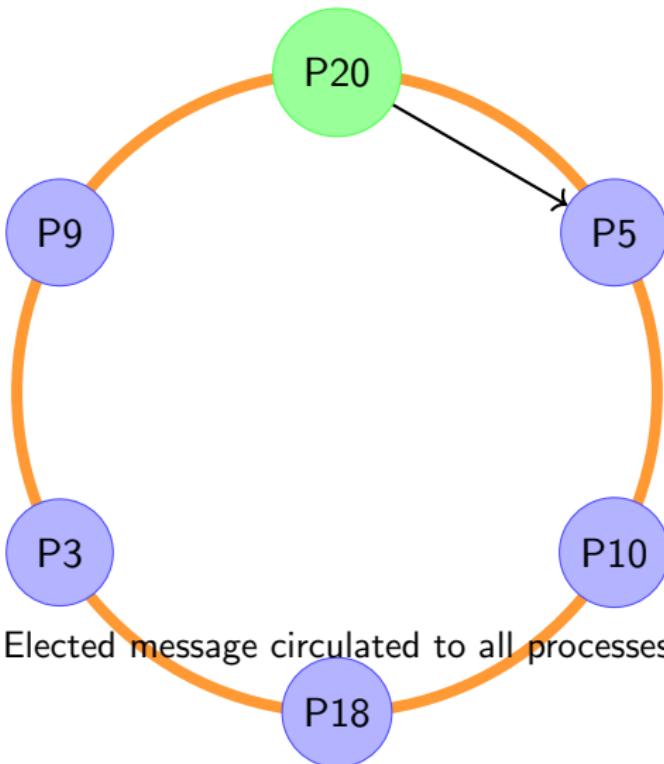
Ring Algorithm – Step 4: Higher ID Replacement



Ring Algorithm – Step 5: Coordinator Election



Ring Algorithm – Step 6: Coordinator Announcement



Need of Ring Algorithm in IoT Networks

- IoT networks consist of multiple **sensors and smart devices**.
- One device must act as a **coordinator / cluster head**.
- If the coordinator fails, the system must:
 - Detect failure
 - Elect a new leader automatically
- Ring Algorithm ensures:
 - Fault tolerance
 - Continuous network operation
 - Distributed control (No central dependency)

Importance in IoT Systems

- Suitable for **low-power IoT devices**.
- Requires **minimal memory and computation**.
- Simple message passing mechanism.
- Efficient for **small and medium IoT clusters**.
- Ensures agreement among all nodes on selected coordinator.

Real-Time Applications in IoT

- **Industrial IoT** – Machine sensors elect cluster head.
- **Smart Cities** – Traffic controllers elect coordinator.
- **Healthcare IoT** – Wearable devices select master node.
- **Smart Agriculture** – Soil sensors elect data aggregator.
- **Smart Homes** – Devices elect hub if main hub fails.

Requirements to Implement in IoT System

- **Hardware Requirements:**

- Microcontroller (ESP32 / Arduino / Raspberry Pi)
- Network module (Wi-Fi / ZigBee / LoRa)

- **Software Requirements:**

- Embedded C / Python
- Communication protocol (MQTT / TCP-IP)
- Failure detection (Heartbeat mechanism)

- Each node must:

- Know its next neighbour
- Have a unique identifier
- Support reliable message passing

Thank You