20
JAN
2022

2-3 PM

An (n-1) rounds Algo for Distributed Sorting on a line network

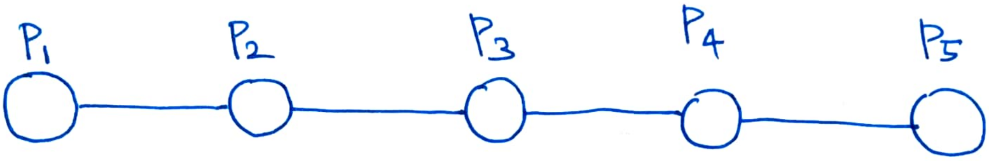
Sasasi 's Algo

Odd-even Transposition Sorting Algo

- $\boxed{n}$ rounds
  ↓
  (n-1) rounds (Sasaki's Algo)

Line Network:

n=5



Assumptions:

→ n = 5

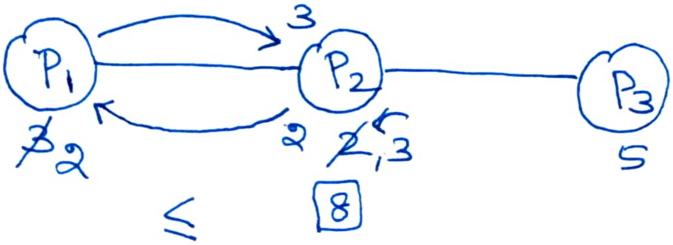
→ fault-free networks (No process or no channel will fail) → No failures

↳ processes / channels are reliable

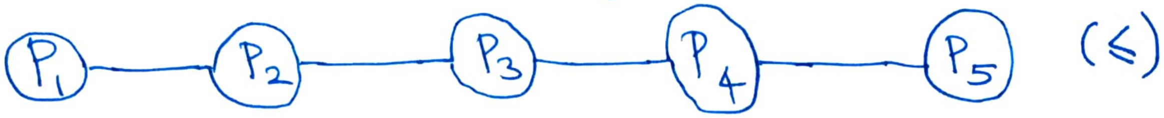
Q: Can we improve the dist. sorting Algo?

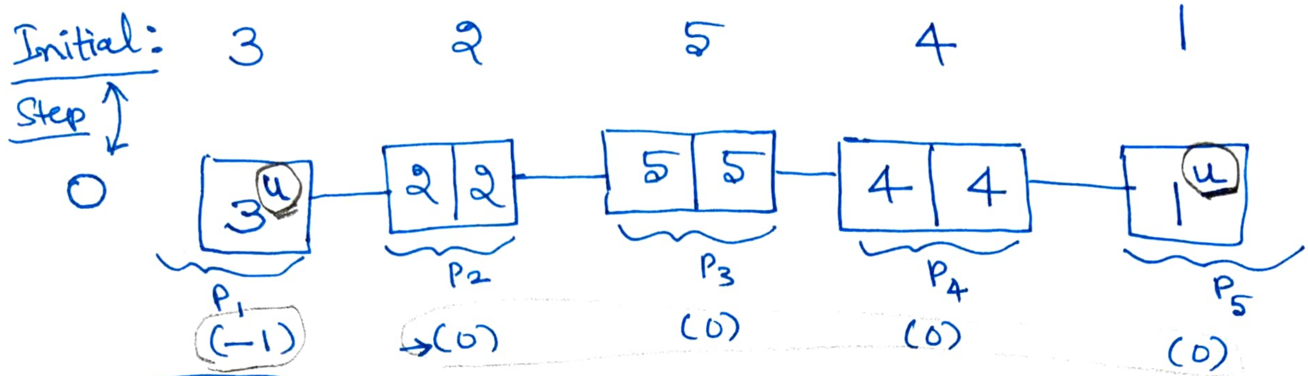
A: Yes.

How?



Perform Dist. Sorting in $(n-1)$ rounds!!

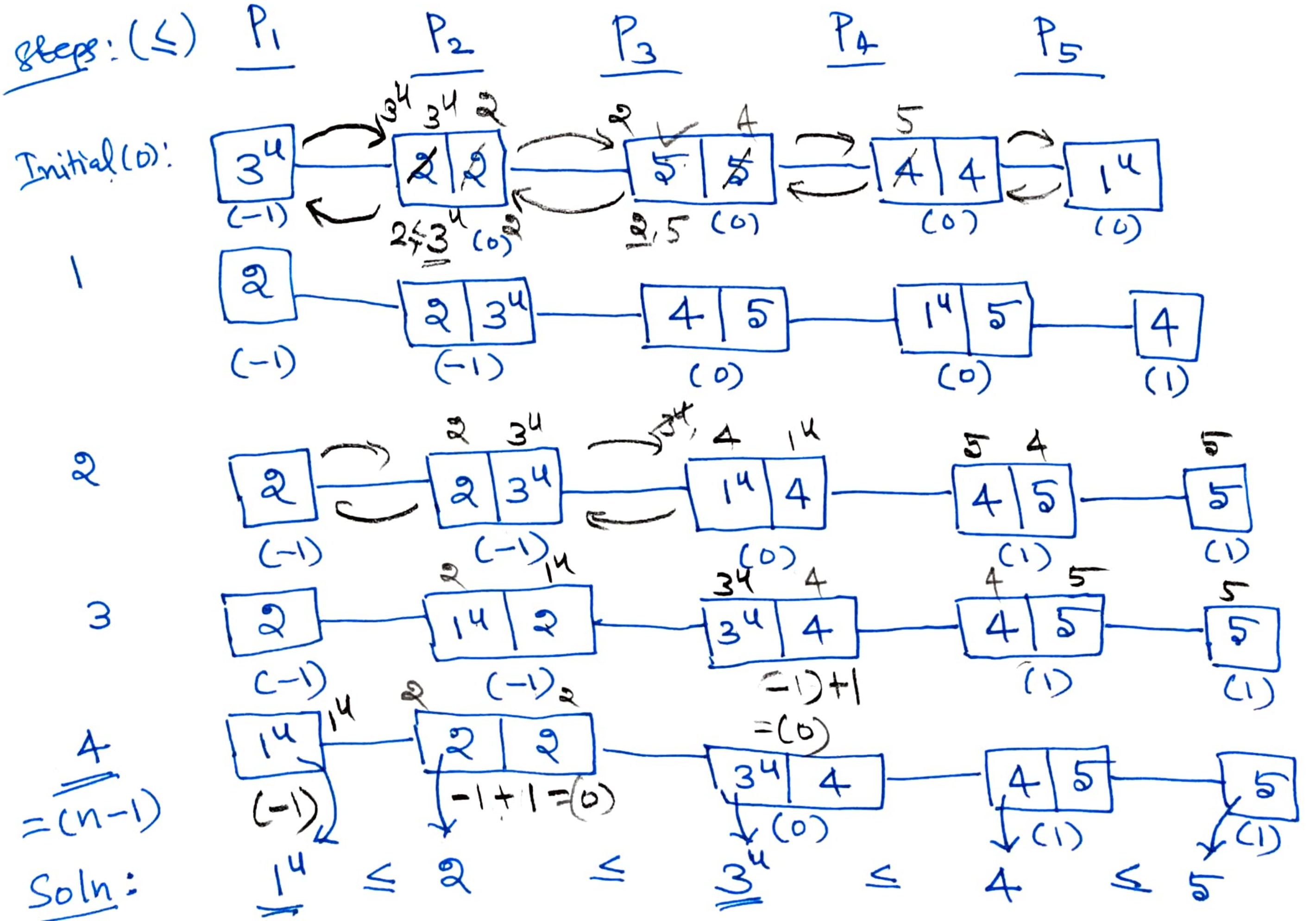




Area ← update?

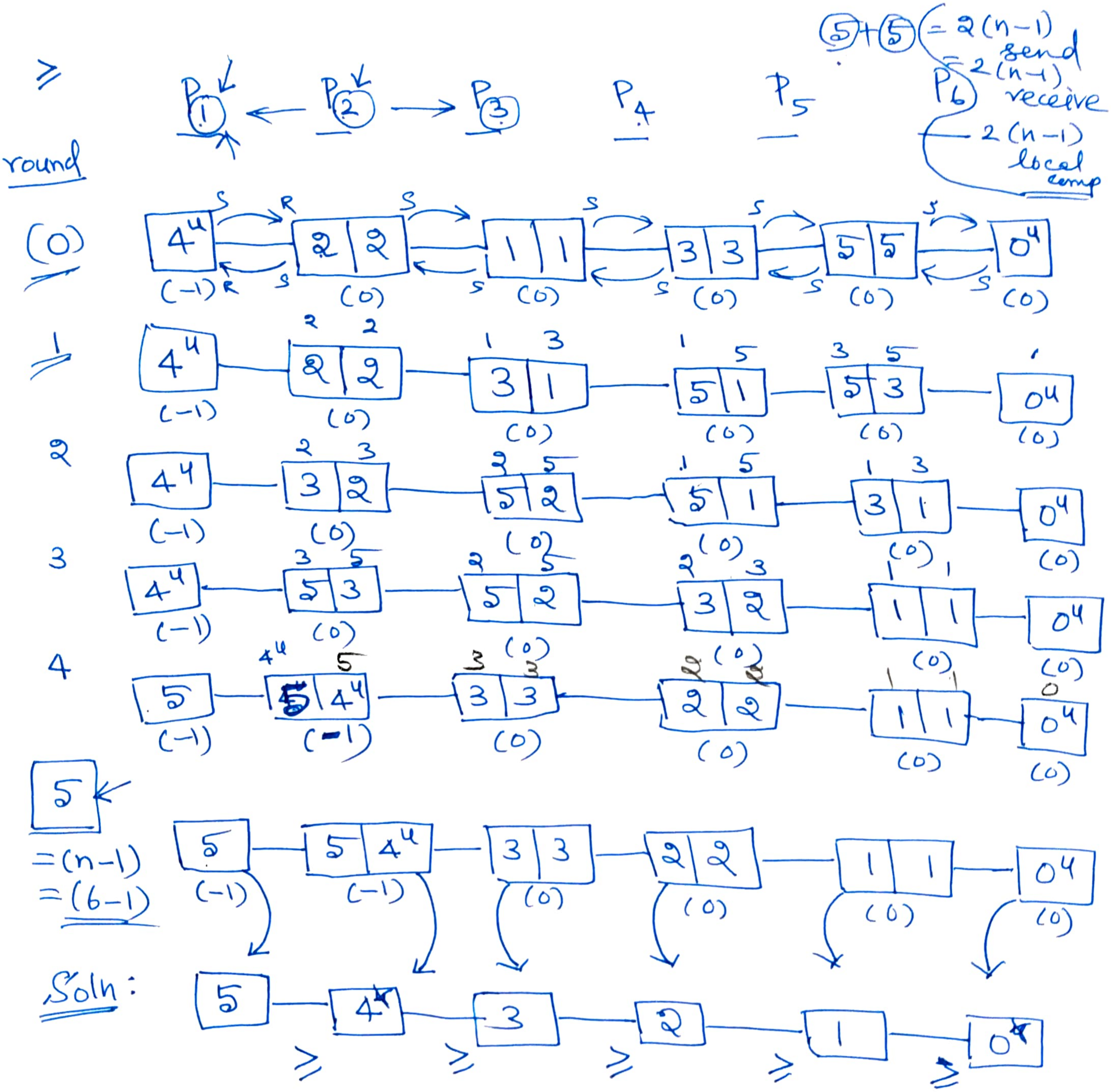
$\text{Area} = \text{area} - 1$, if the element marked as unique (u) moves from P_i to P_{i+1} (left to right)
receiving P_i

$= \text{area} + 1$, if the marked element moves from P_{i+1} to P_i (right to left)
sending P_i's area



steps: (≤)	P_1	P_2	P_3	P_4	P_5
Initial(0):	3^u (−1)	2 | 2 (0)	5 | 5 (0)	4 | 4 (0)	1^u (0)
1	2 (−1)	2 | 3^u (−1)	4 | 5 (0)	1^u | 5 (0)	4 (1)
2	2 (−1)	2 | 3^u (−1)	1^u | 4 (0)	4 | 5 (1)	5 (1)
3	2 (−1)	1^u | 2 (−1)	3^u | 4 (−1)+1 =(0)	4 | 5 (1)	5 (1)
4 =(n−1)	1^u (−1)	2 | 2 −1+1=(0)	3^u | 4 (0)	4 | 5 (1)	5 (1)

Soln: $1^u \leq 2 \leq 3^u \leq 4 \leq 5$



n rounds → $(n-1)$ rounds (time-optimal)

Three types of inputs:

1) Sorted seq
2) reversely sorted Seq
3) random Sequence.

AdV

1) $(n-1)$ rounds algo (time-optimal)
2) Processes IDs are not necessarily known.