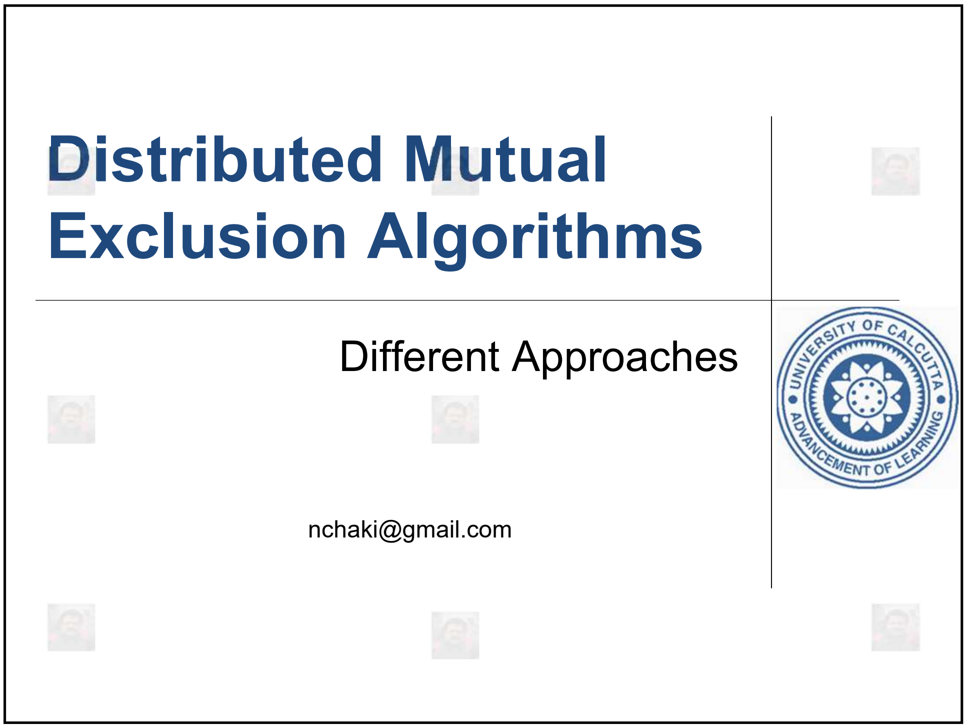
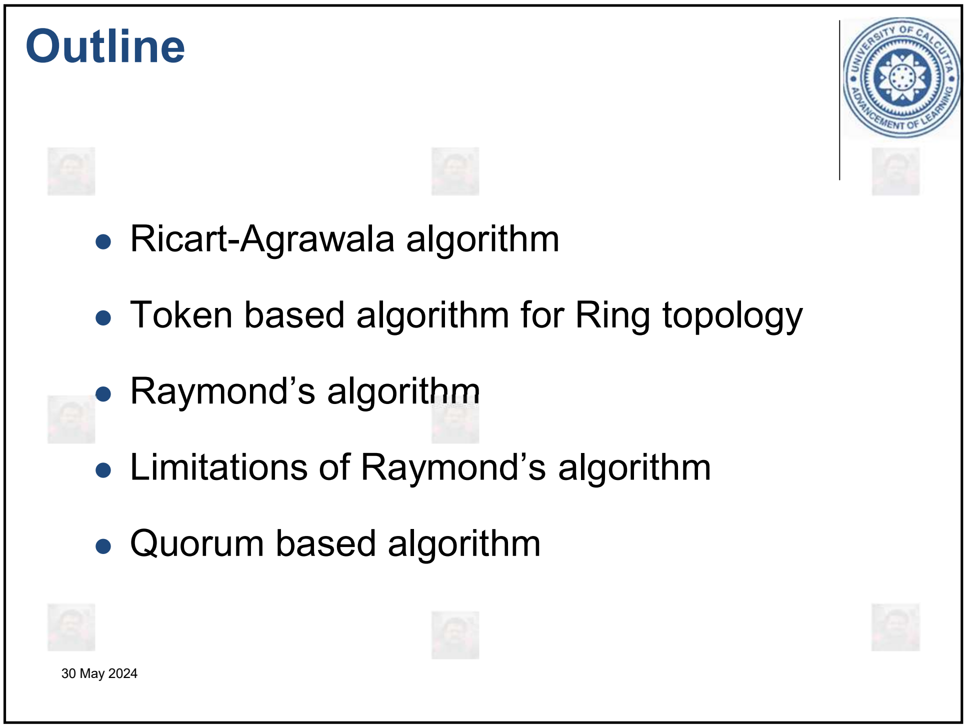


Distributed Mutual Exclusion Algorithms

Different Approaches

nchaki@gmail.com

1



Outline

- Ricart-Agrawala algorithm
- Token based algorithm for Ring topology
- Raymond’s algorithm
- Limitations of Raymond’s algorithm
- Quorum based algorithm

30 May 2024

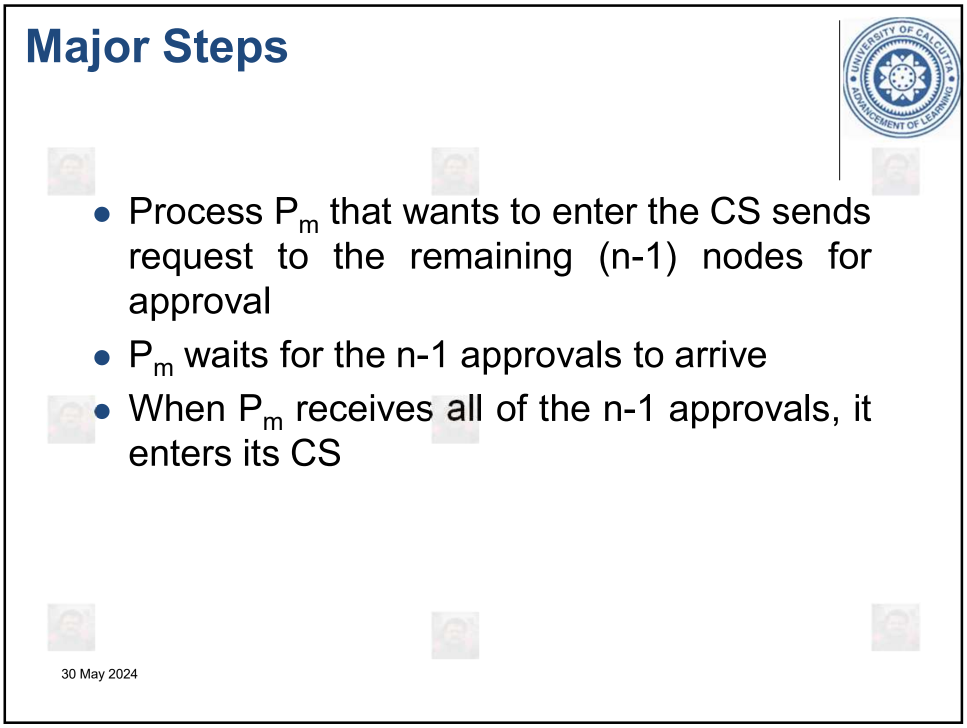
2

Outline

- Ricart-Agrawala algorithm
- Token based algorithm for Ring topology
- Raymond’s algorithm
- Limitations of Raymond’s algorithm
- Quorum based algorithm

Basic Ideas

- The requesting process has to get the approval of all processes to enter its CS
- A processes currently in its own CS, cannot give approval to another candidate
- What would be the response of another candidate process to a request

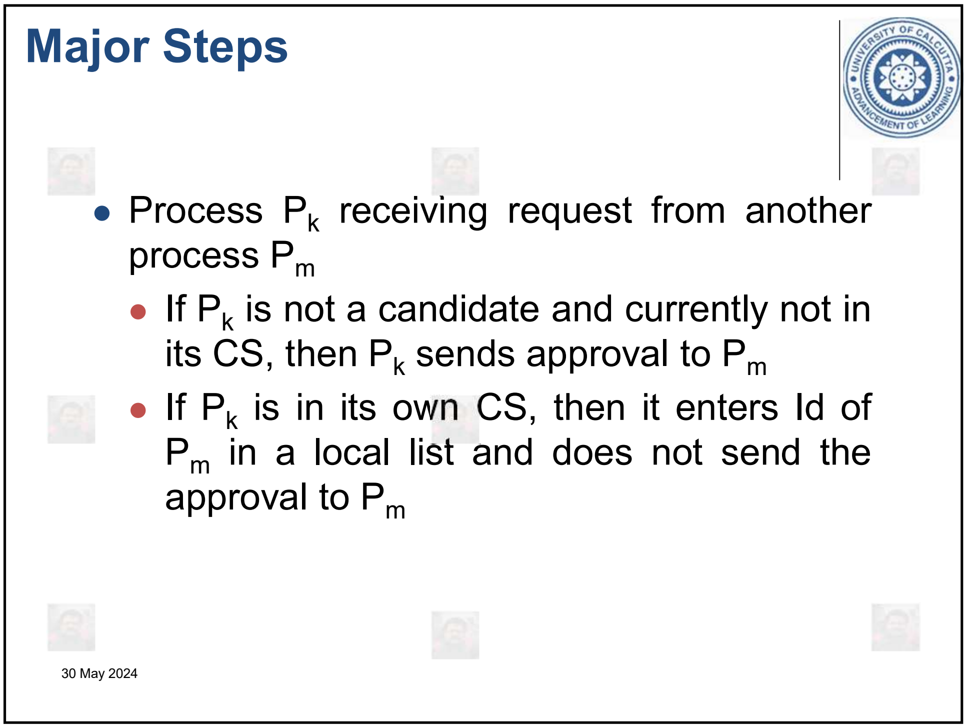


Major Steps

- Process P_m that wants to enter the CS sends request to the remaining (n-1) nodes for approval
- P_m waits for the n-1 approvals to arrive
- When P_m receives all of the n-1 approvals, it enters its CS

30 May 2024

5

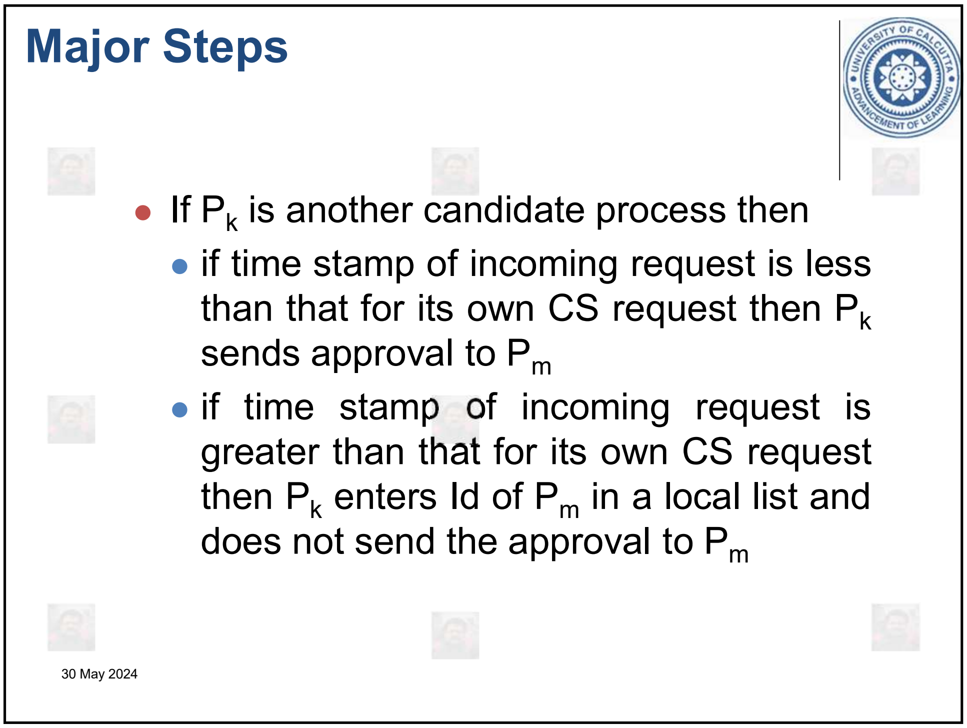


Major Steps

- Process P_k receiving request from another process P_m
  - If P_k is not a candidate and currently not in its CS, then P_k sends approval to P_m
  - If P_k is in its own CS, then it enters Id of P_m in a local list and does not send the approval to P_m

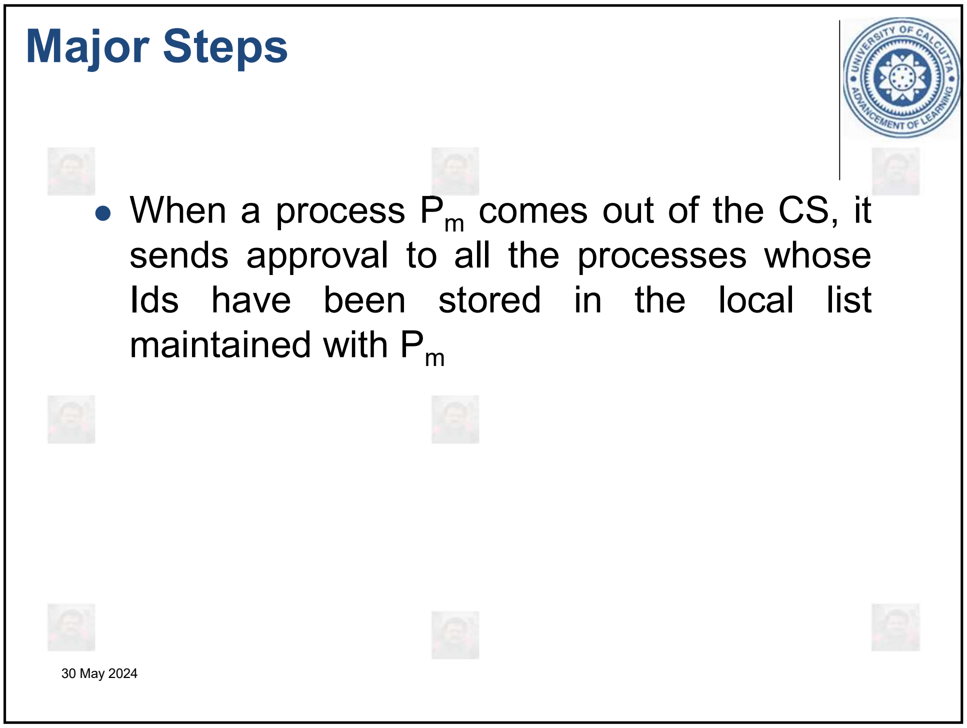
30 May 2024

6



Major Steps

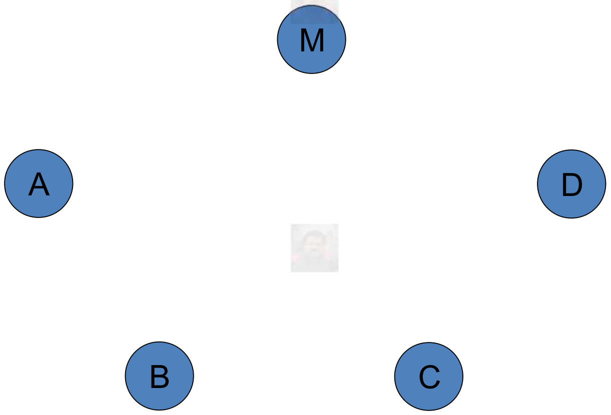
- If P_k is another candidate process then
  - if time stamp of incoming request is less than that for its own CS request then P_k sends approval to P_m
  - if time stamp of incoming request is greater than that for its own CS request then P_k enters Id of P_m in a local list and does not send the approval to P_m



Major Steps

- When a process P_m comes out of the CS, it sends approval to all the processes whose Ids have been stored in the local list maintained with P_m

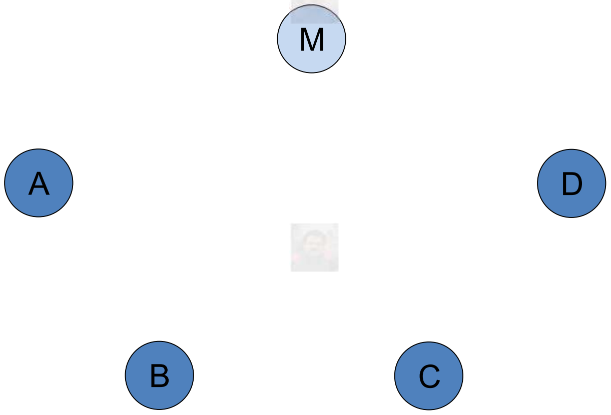
Ricart-Agrawala DME algorithm



30 May 2024

9

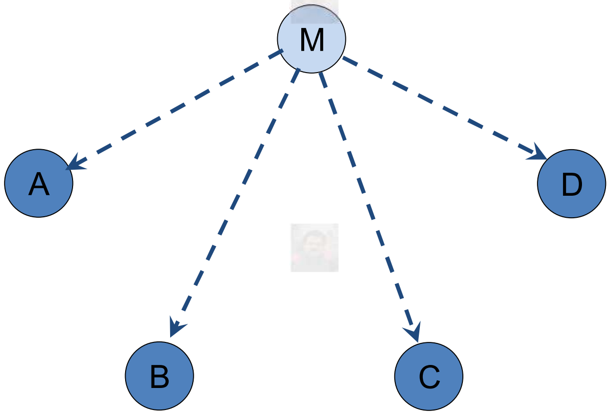
Ricart-Agrawala DME algorithm



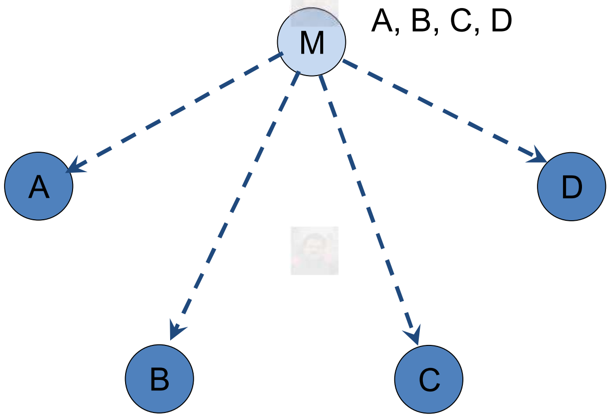
30 May 2024

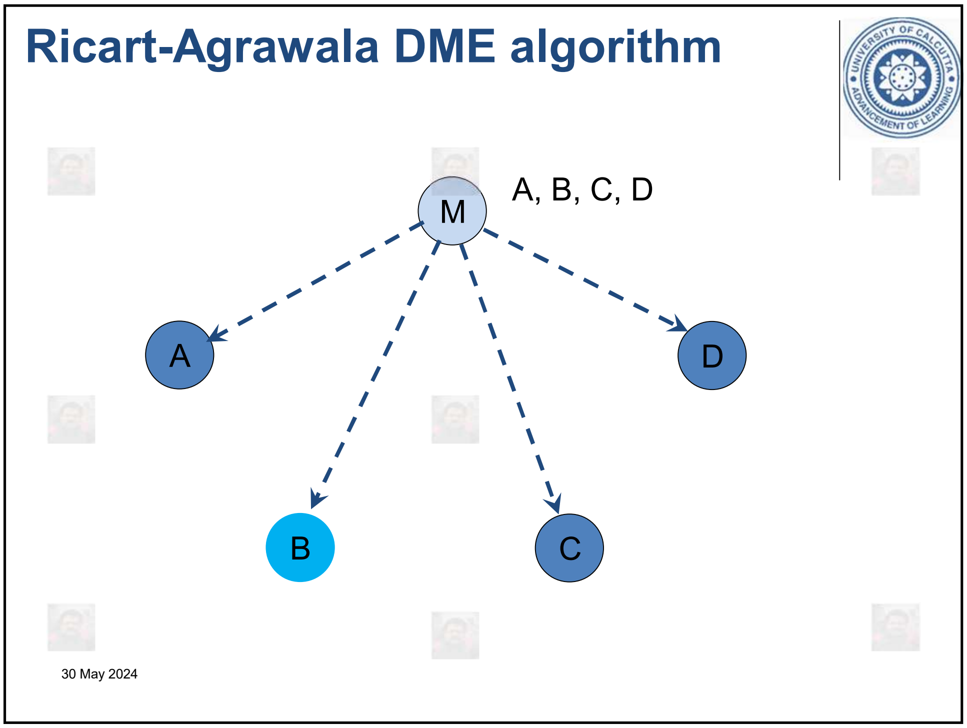
10

Ricart-Agrawala DME algorithm



Ricart-Agrawala DME algorithm

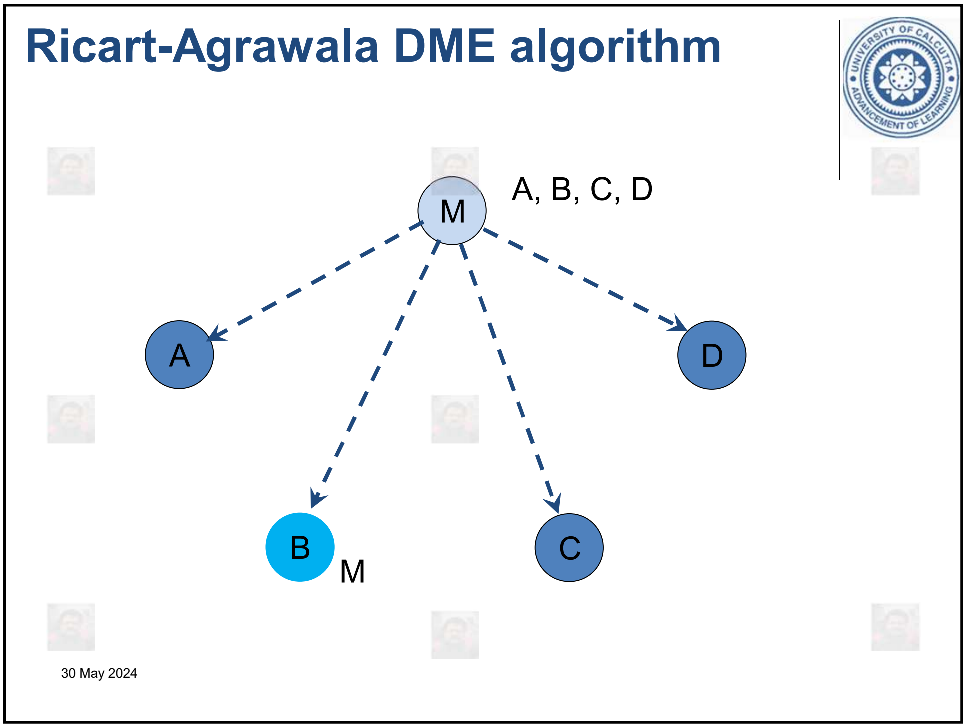




Ricart-Agrawala DME algorithm

30 May 2024

13

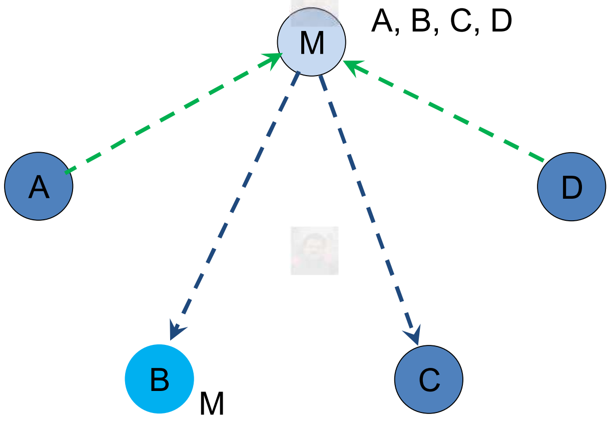


Ricart-Agrawala DME algorithm

30 May 2024

14

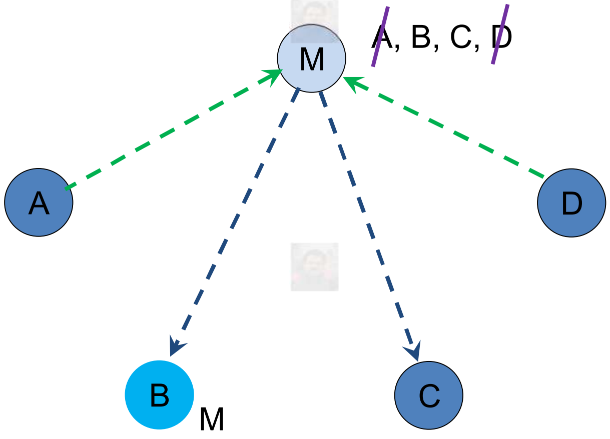
Ricart-Agrawala DME algorithm



30 May 2024

15

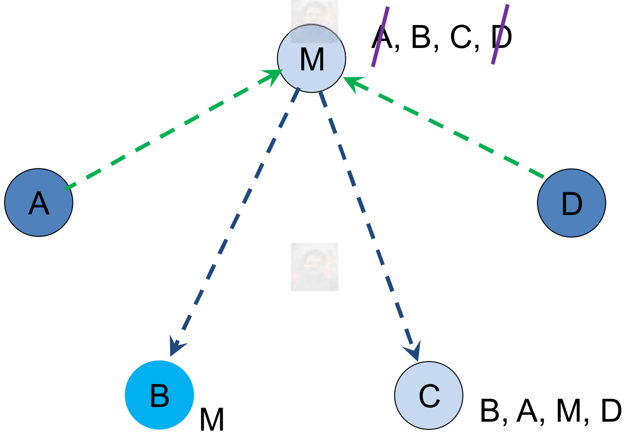
Ricart-Agrawala DME algorithm



30 May 2024

16

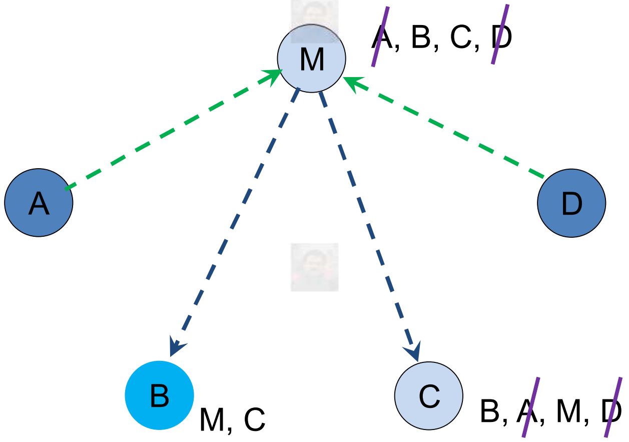
Ricart-Agrawala DME algorithm



30 May 2024

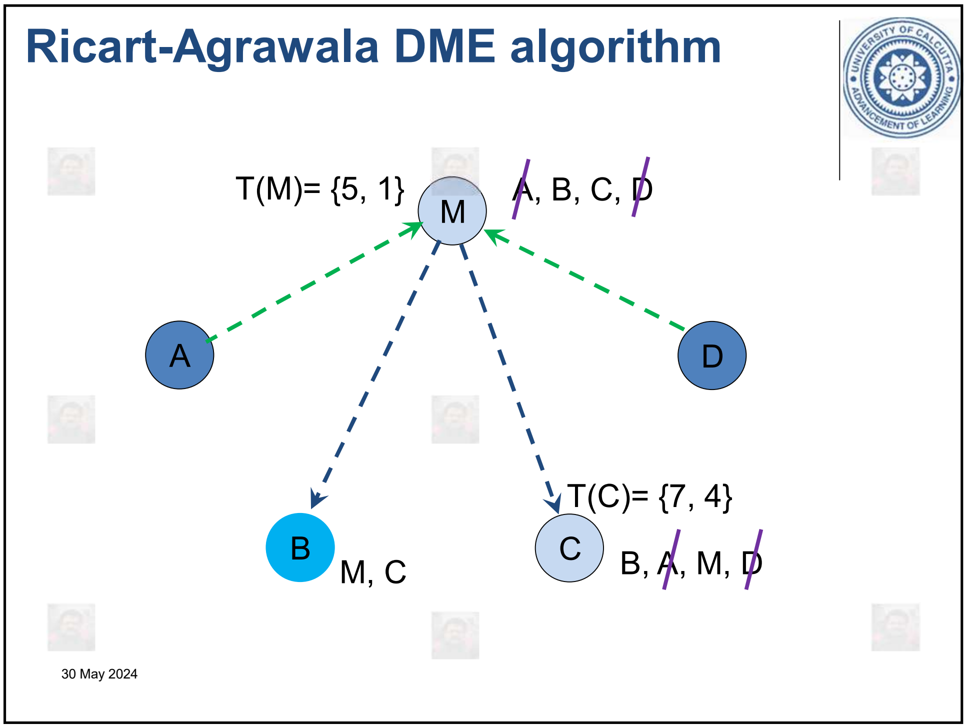
17

Ricart-Agrawala DME algorithm



30 May 2024

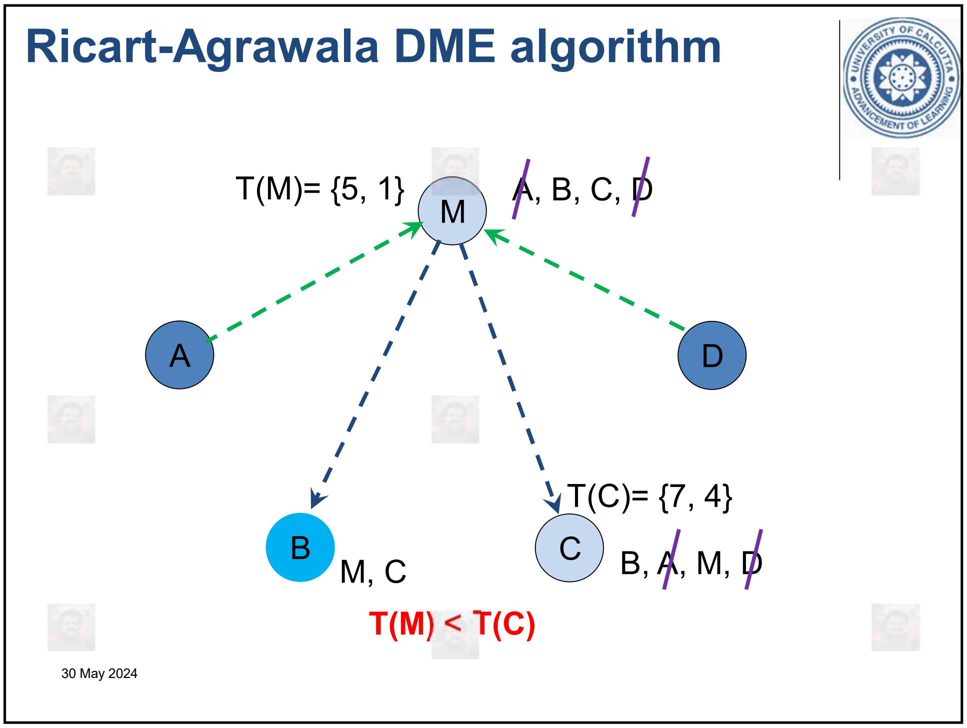
18



Ricart-Agrawala DME algorithm

30 May 2024

19



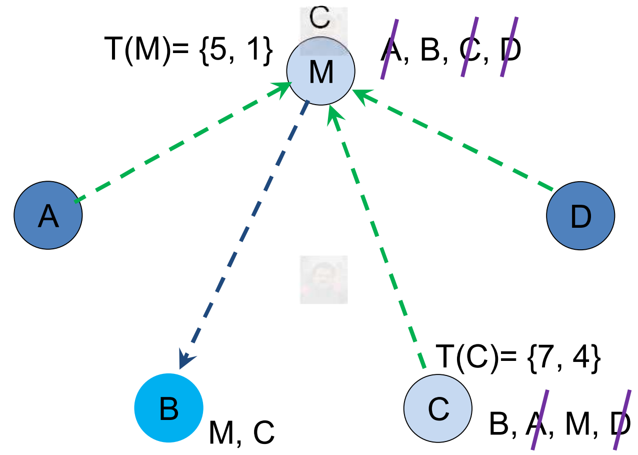
Ricart-Agrawala DME algorithm

T(M) < T(C)

30 May 2024

20

Ricart-Agrawala DME algorithm



C

T(M)= {5, 1}

M

~~A~~, B, ~~C~~, ~~D~~

A

D

B

M, C

T(C)= {7, 4}

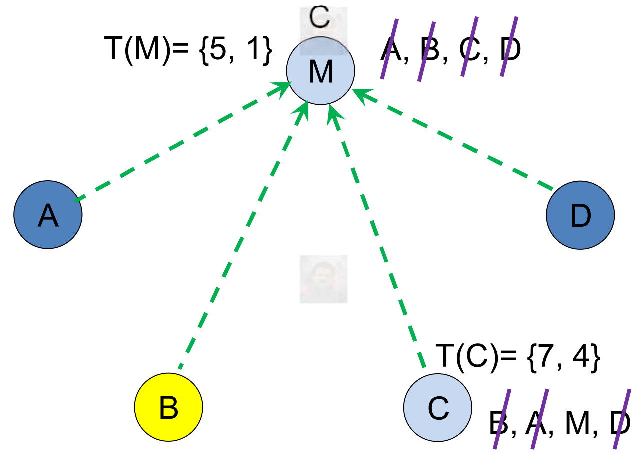
C

B, ~~A~~, M, ~~D~~

30 May 2024

21

Ricart-Agrawala DME algorithm



C

T(M)= {5, 1}

M

~~A~~, ~~B~~, ~~C~~, ~~D~~

A

D

B

T(C)= {7, 4}

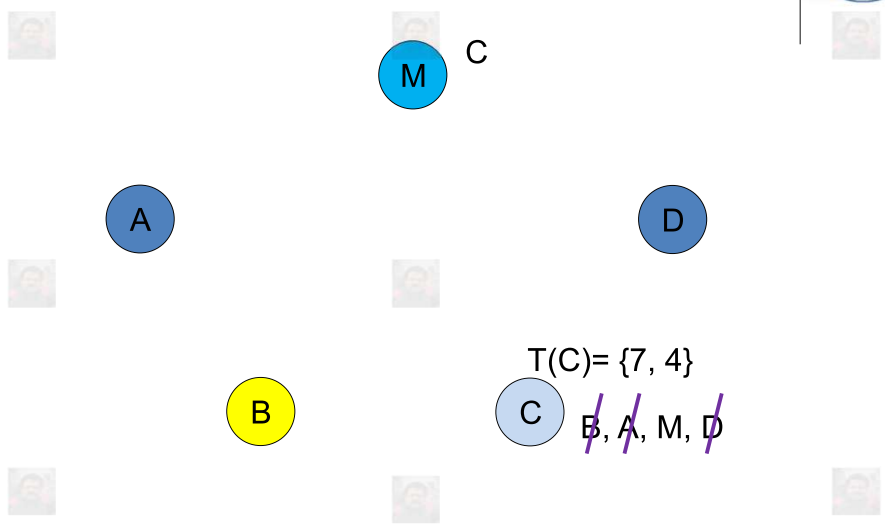
C

~~B~~, ~~A~~, M, ~~D~~

30 May 2024

22

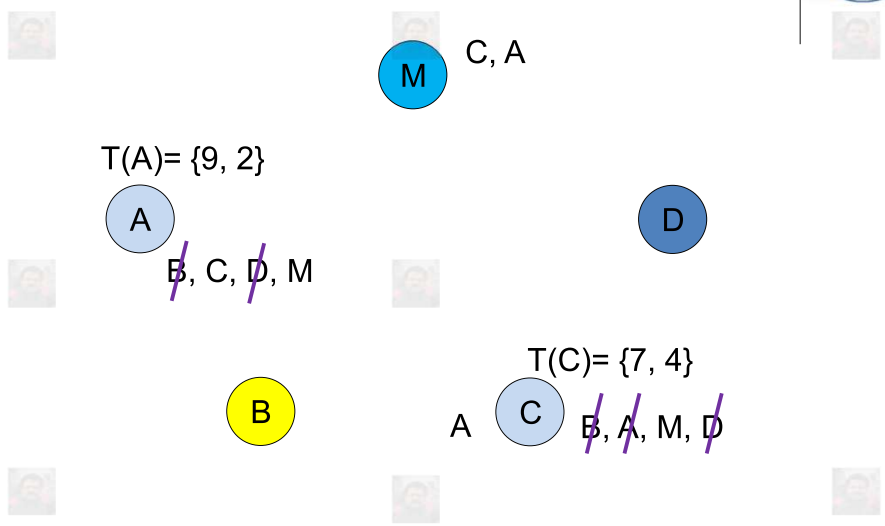
Ricart-Agrawala DME algorithm



30 May 2024

23

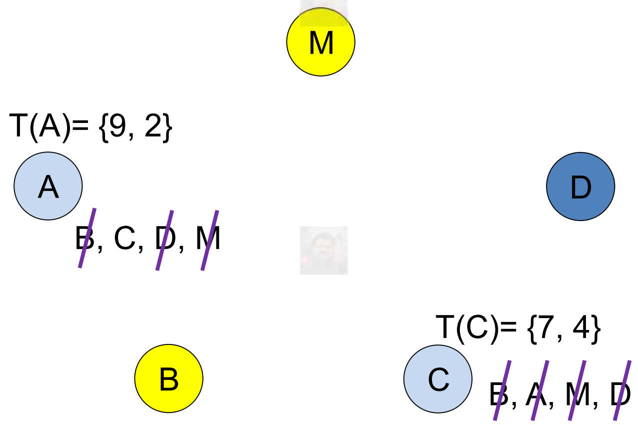
Ricart-Agrawala DME algorithm



30 May 2024

24

Ricart-Agrawala DME algorithm



T(A)= {9, 2}

~~B~~, C, ~~D~~, ~~M~~

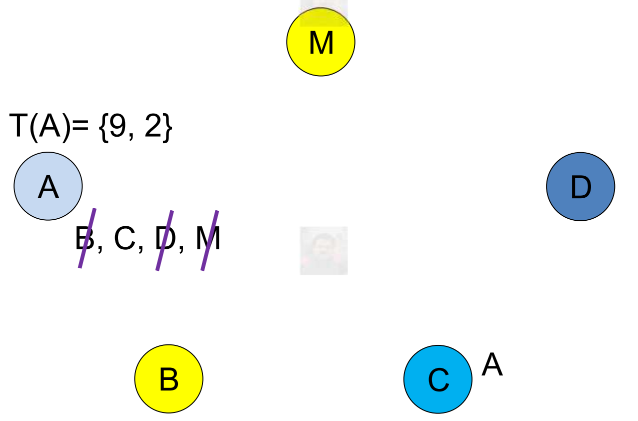
T(C)= {7, 4}

~~B~~, ~~A~~, ~~M~~, ~~D~~

30 May 2024

25

Ricart-Agrawala DME algorithm



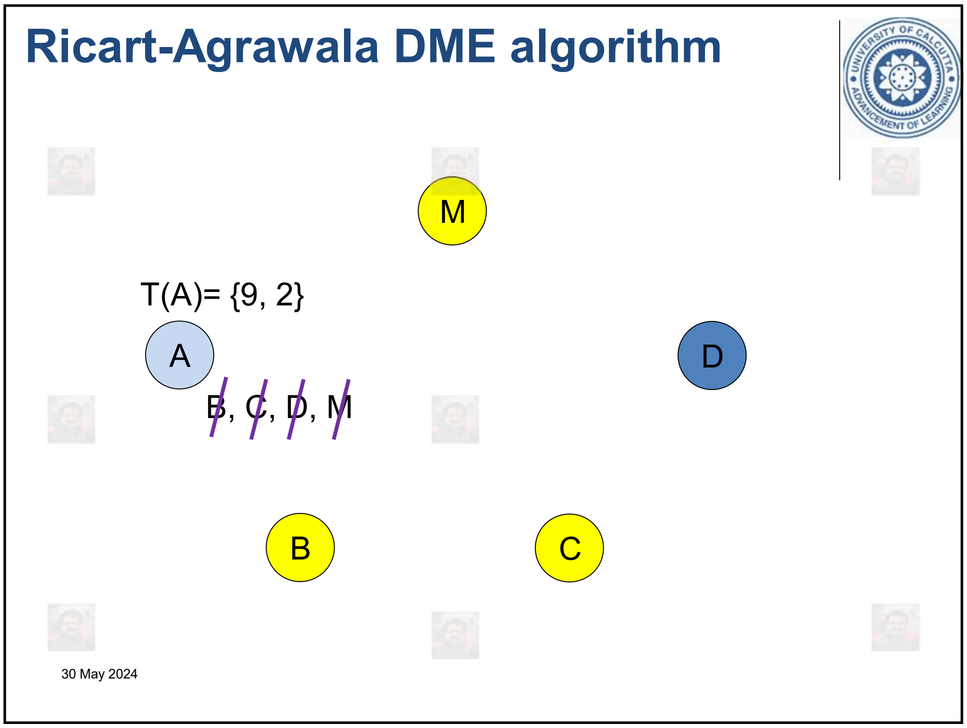
T(A)= {9, 2}

~~B~~, C, ~~D~~, ~~M~~

A

30 May 2024

26



Ricart-Agrawala DME algorithm

M

T(A)= {9, 2}

A

D

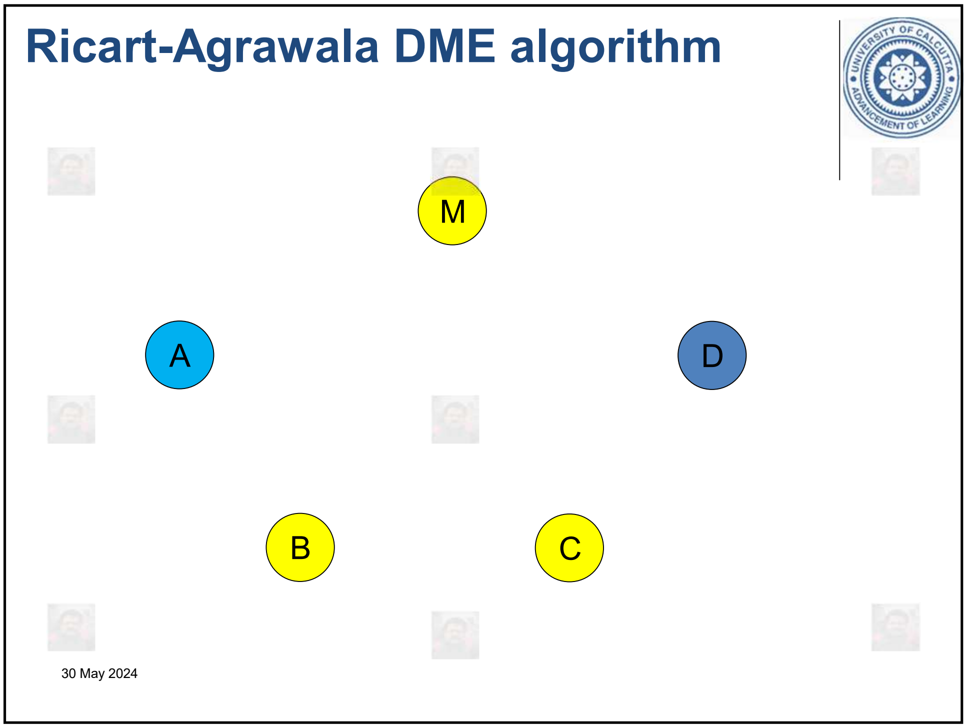
~~B~~, ~~C~~, ~~D~~, ~~M~~

B

C

30 May 2024

27



Ricart-Agrawala DME algorithm

M

A

D

B

C

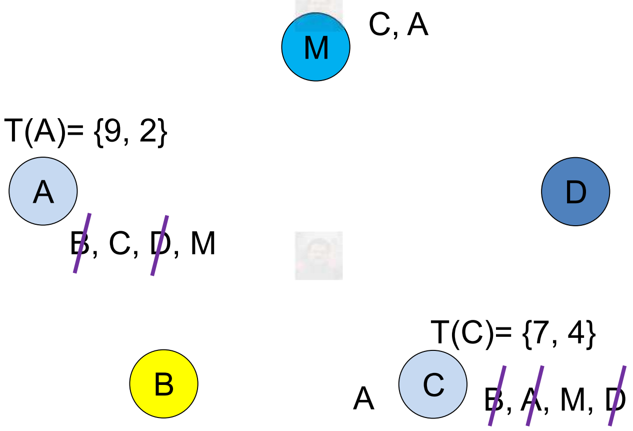
30 May 2024

28

Ricart-Agrawala DME algorithm

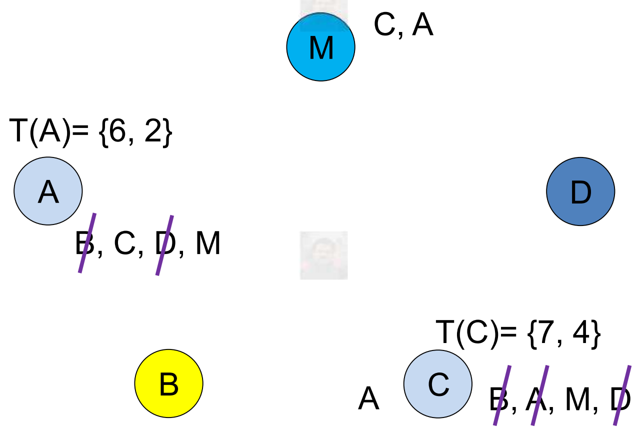
- Being a Symmetric algorithm, decisions are taken collectively and based on mutual consent
- 2*(N-1) messages are exchanged for each access to CS
- In case of a network size of N, for M requests there will be 2M*(N-1) control message exchanges ≃ O(N*M)

Ricart-Agrawala DME algorithm



T(C) < T(A)

What happens if T(A) < T(C)?



M C, A

T(A)= {6, 2}

A

~~B~~, C, ~~D~~, M

D

T(C)= {7, 4}

B

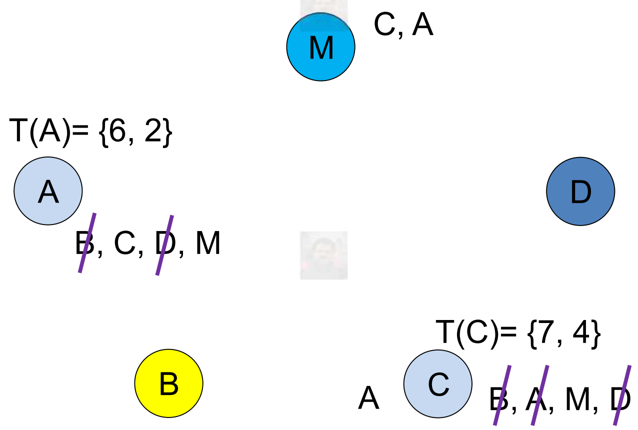
A C ~~B~~, ~~A~~, M, ~~D~~

T(A) < T(C)

30 May 2024

31

What happens if T(A) < T(C)?



M C, A

T(A)= {6, 2}

A

~~B~~, C, ~~D~~, M

D

T(C)= {7, 4}

B

A C ~~B~~, ~~A~~, M, ~~D~~

Is this possible?

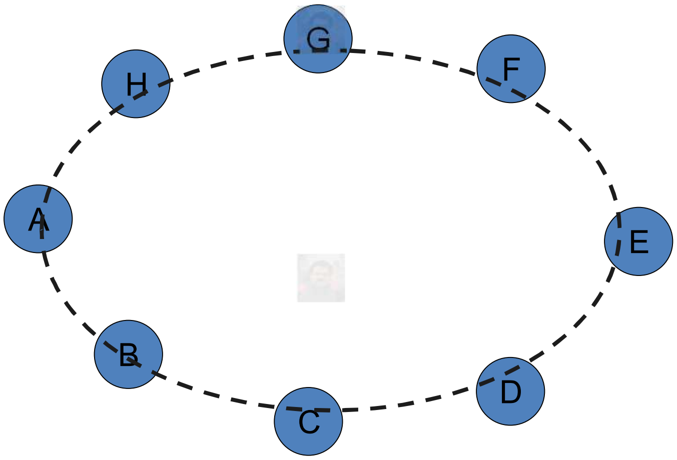
30 May 2024

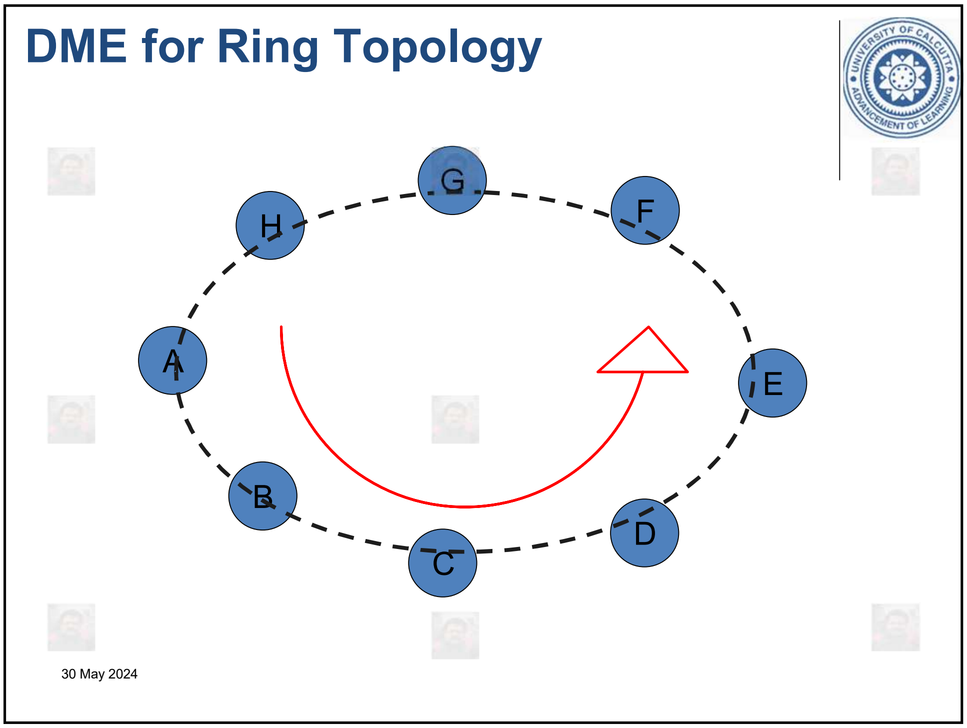
32

Outline

- Ricart-Agrawala algorithm
- Token based algorithm for Ring topology
- Raymond’s algorithm
- Limitations of Raymond’s algorithm
- Quorum based algorithm

DME for Ring Topology

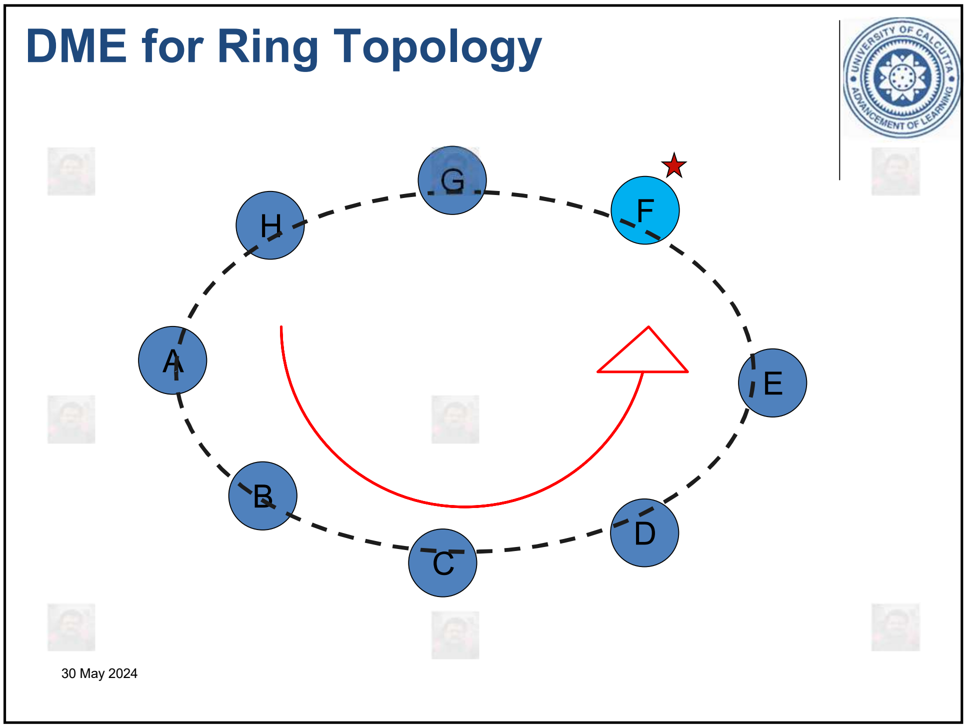




DME for Ring Topology

30 May 2024

35

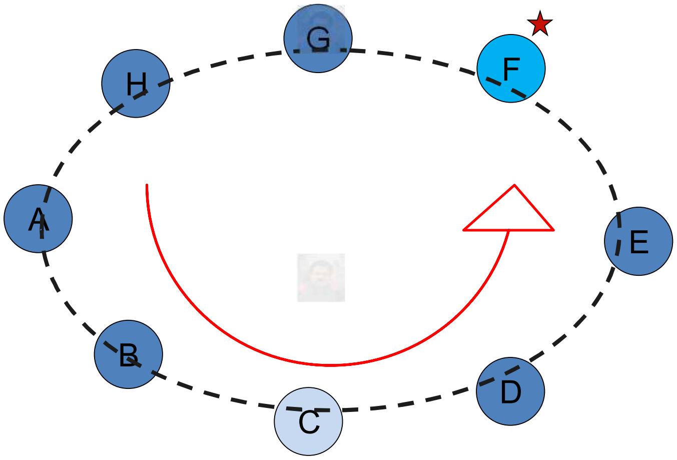


DME for Ring Topology

30 May 2024

36

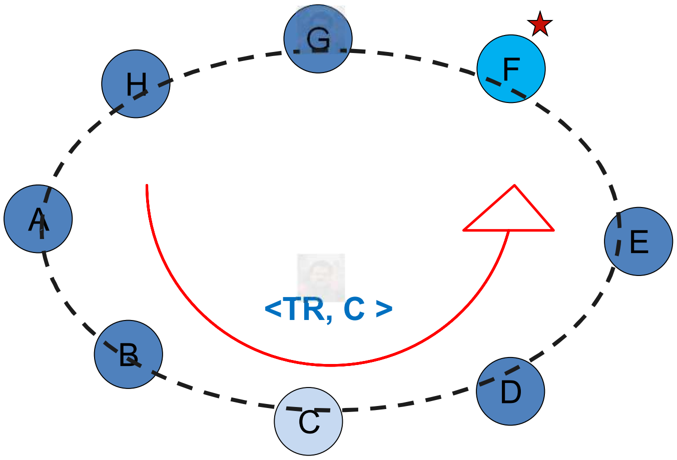
DME for Ring Topology



30 May 2024

37

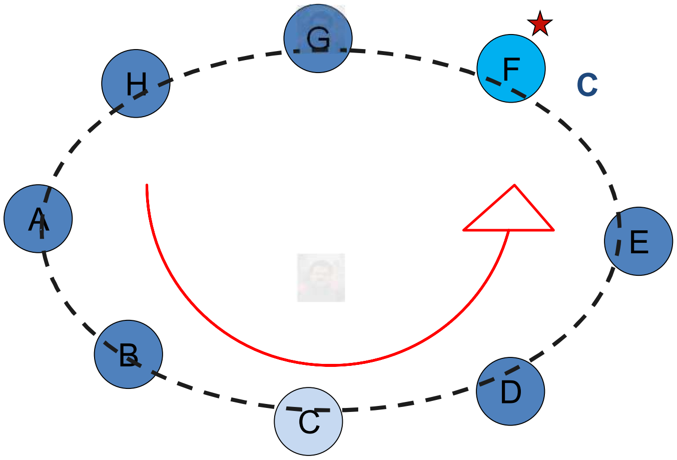
DME for Ring Topology



30 May 2024

38

DME for Ring Topology

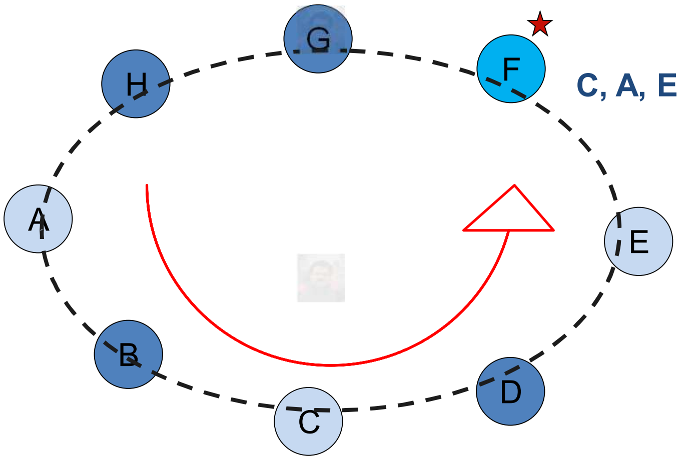


C

30 May 2024

39

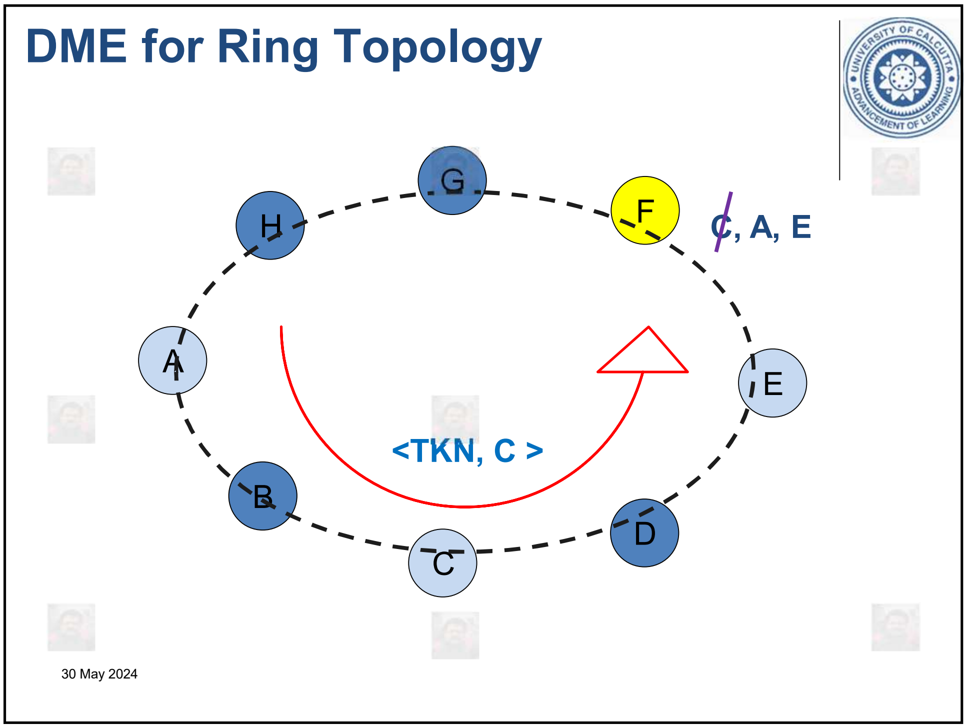
DME for Ring Topology



C, A, E

30 May 2024

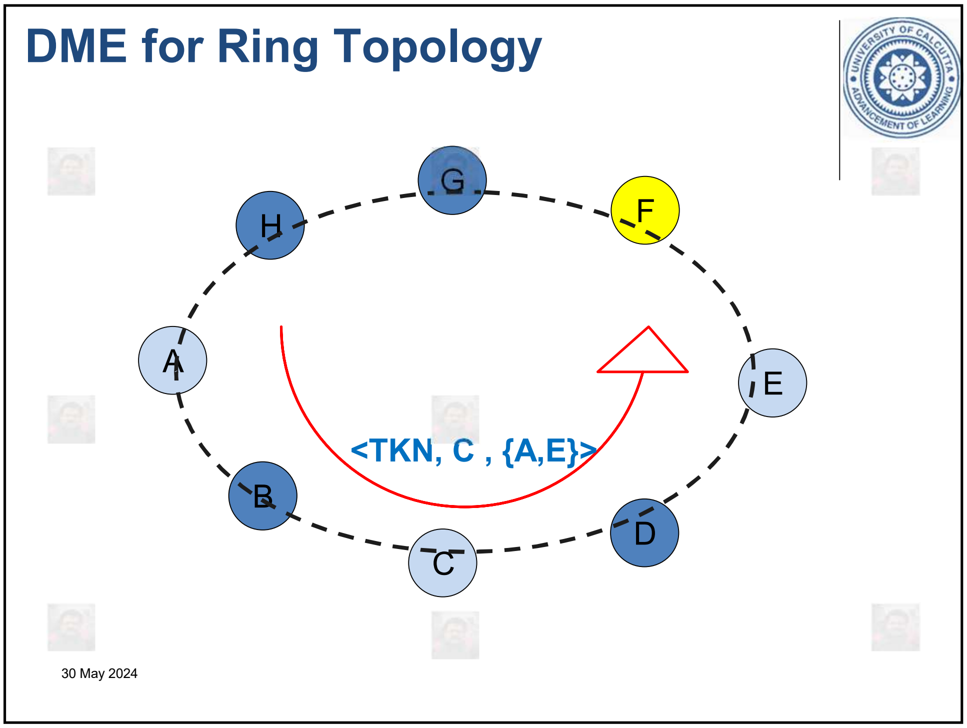
40



DME for Ring Topology

30 May 2024

41

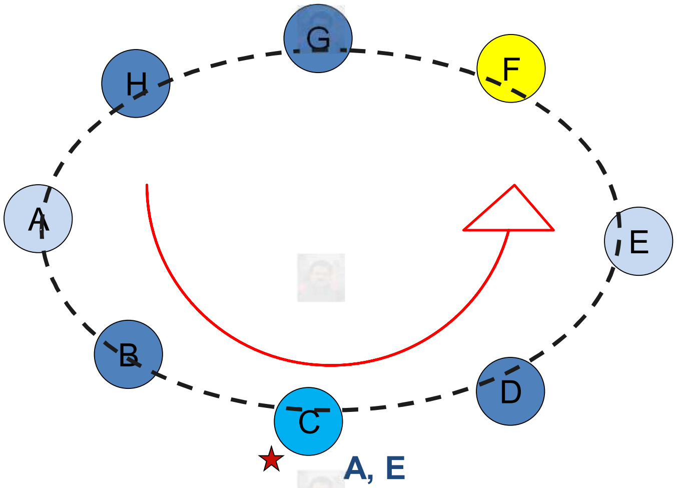


DME for Ring Topology

30 May 2024

42

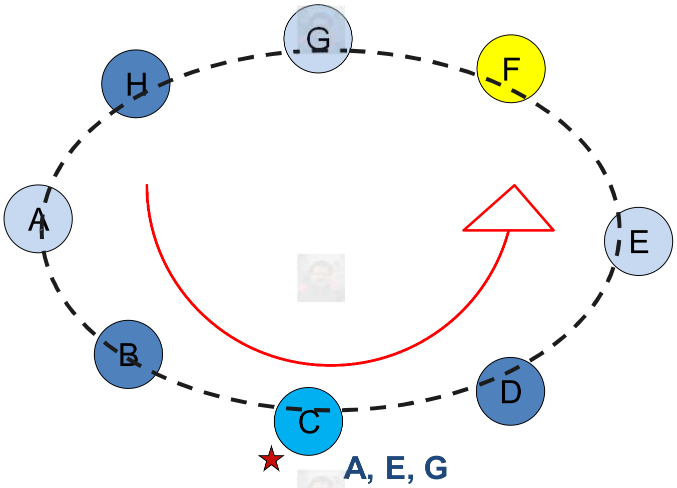
DME for Ring Topology



30 May 2024

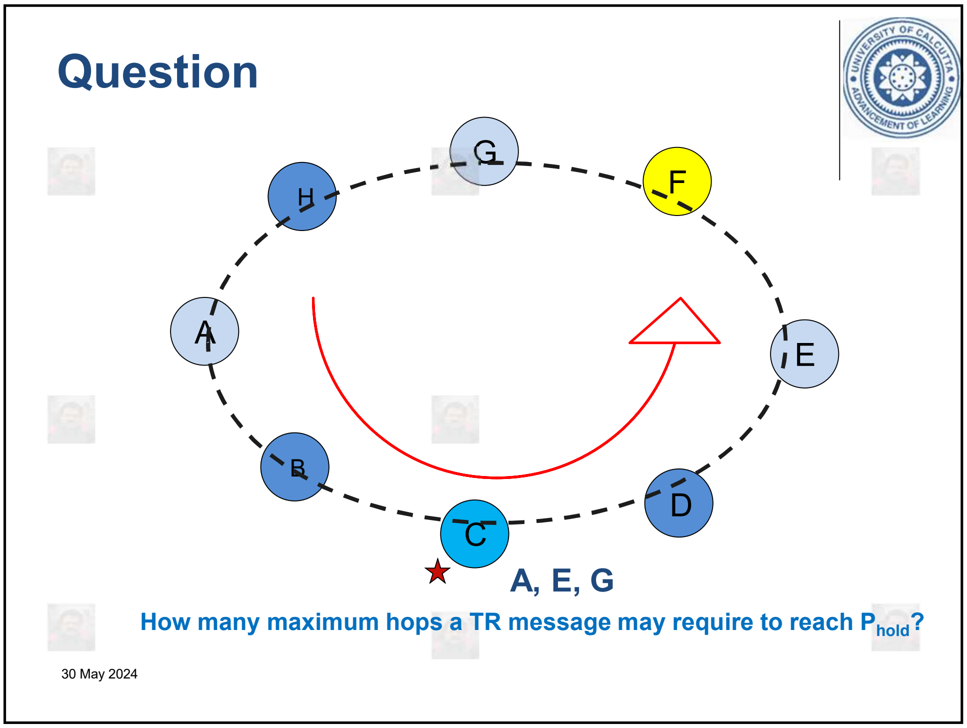
43

DME for Ring Topology



30 May 2024

44



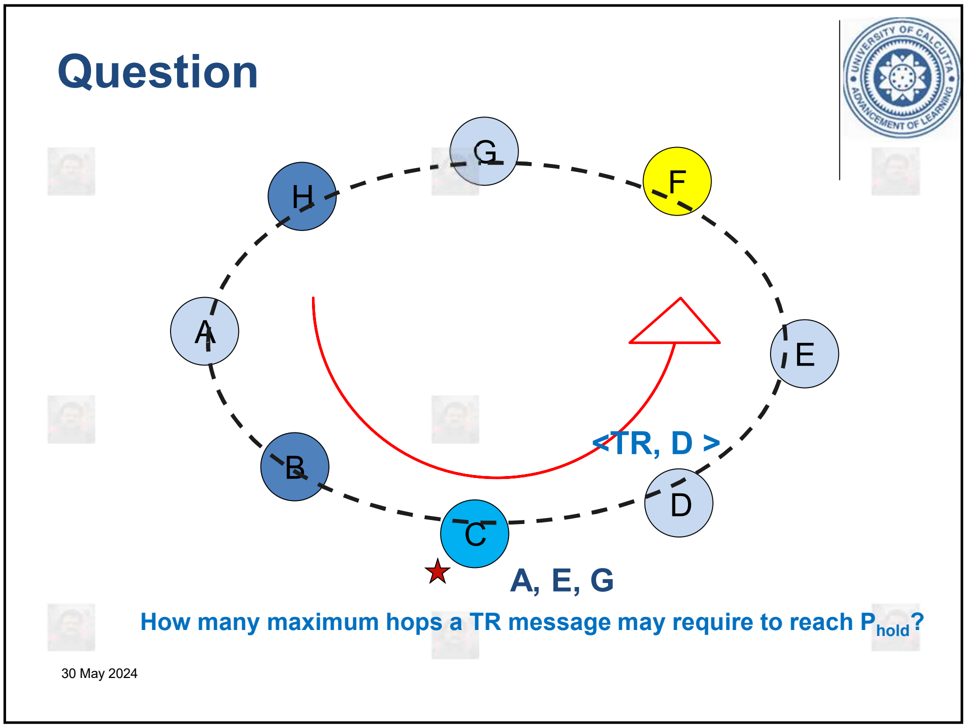
Question

A, E, G

How many maximum hops a TR message may require to reach P_{hold}?

30 May 2024

45



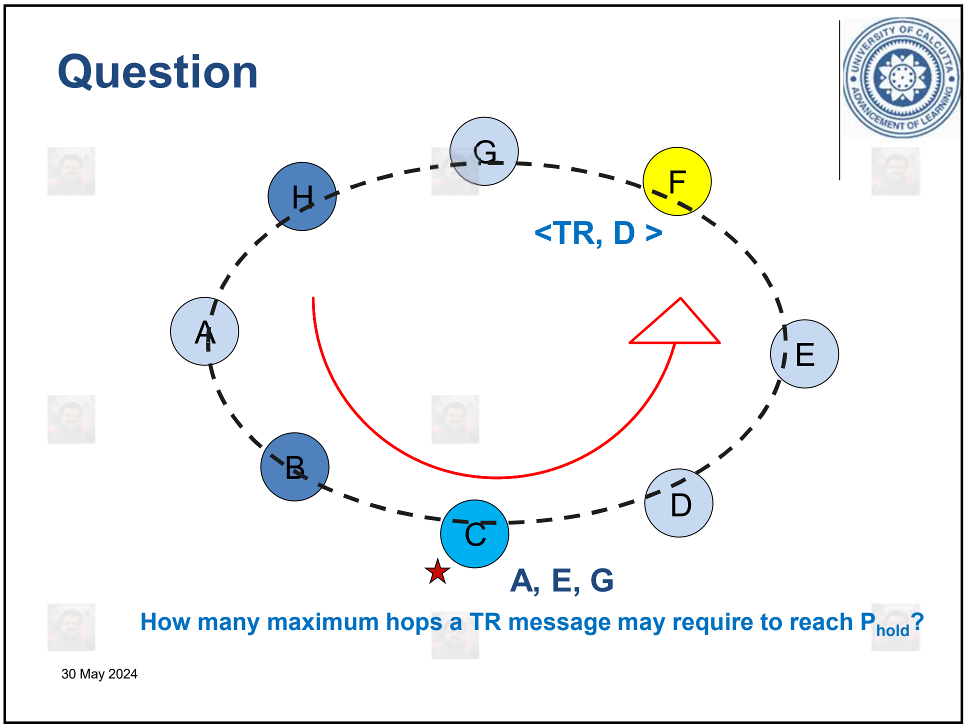
Question

A, E, G

How many maximum hops a TR message may require to reach P_{hold}?

30 May 2024

46

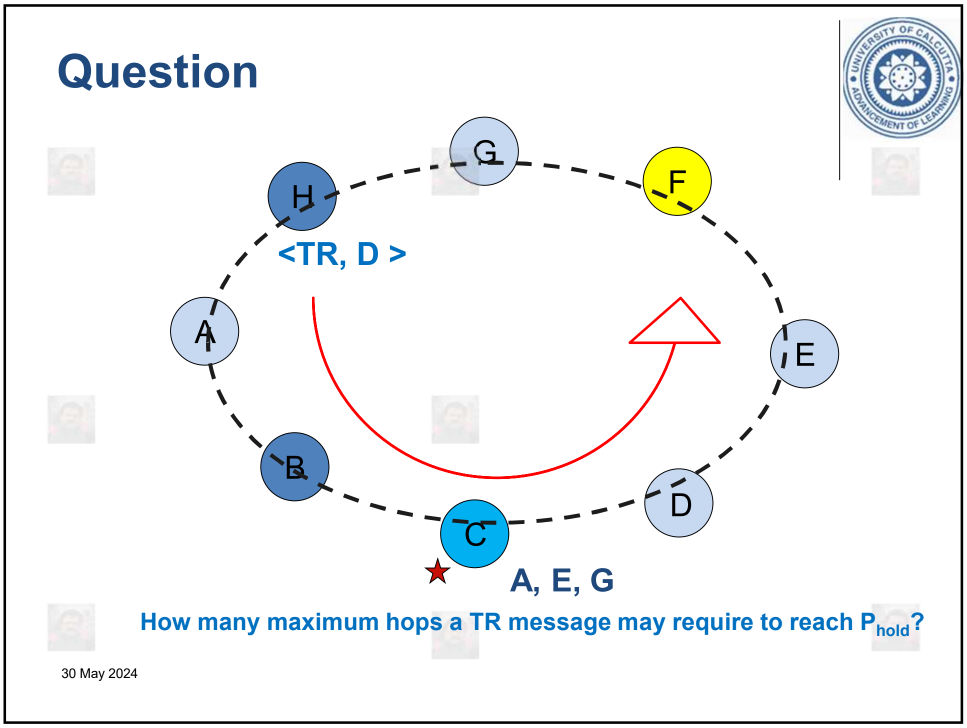


Question

How many maximum hops a TR message may require to reach P_{hold}?

30 May 2024

47

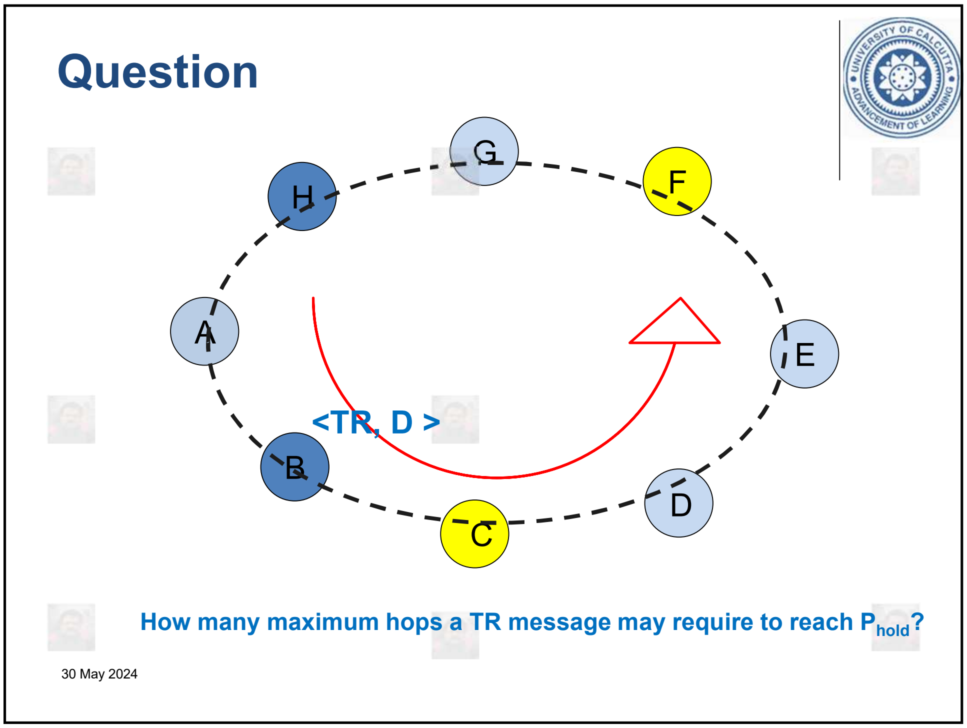


Question

How many maximum hops a TR message may require to reach P_{hold}?

30 May 2024

48

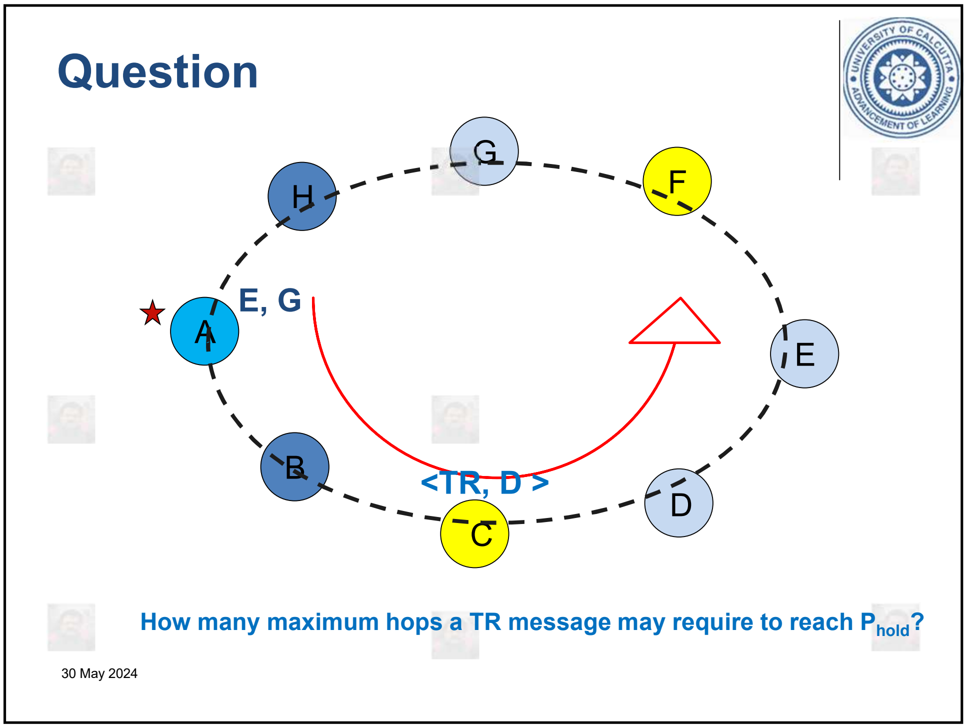


Question

How many maximum hops a TR message may require to reach P_{hold}?

30 May 2024

49



Question

How many maximum hops a TR message may require to reach P_{hold}?

30 May 2024

50

Outline

- Ricart-Agrawala algorithm
- Token based algorithm for Ring topology
- Raymond’s algorithm
- Limitations of Raymond’s algorithm
- Quorum based algorithm

30 May 2024

51

Basic Premises

- Raymond’s algorithm uses an inverted spanning tree to reduce the number of control messages per CS access.
- The algorithm assumes that the underlying network guarantees message delivery.
- All nodes of the network are completely reliable.

30 May 2024

52

Basic Premises

- A node needs to hold information about and communicate only to its 1-hop neighboring nodes.
- In stead of token, Raymond used the term PRIVILEGE
- Only one node can be in possession of the PRIVILEGE (next onwards we'll refer it as Token) at any time

53

Basic Premises

- When there are no nodes requesting for the token, it remains in possession of the node that last used it.
- We would refer the token holding node at any point of time as P_{hold}

54

Major Steps in Raymond’s Algorithm

- For a process P_m that wants to enter the CS, the following steps are done:
  - ID of the requesting node, m in this case, is entered in a local queue of site for P_m
  - A token request is sent to the parent node of P_m in the tree, say to node P_k

Major Steps in Raymond’s Algorithm

- For any intermediate node P_k, on receipt of a token request from one of its decedents, P_m in this case, , the following steps are done:
  - ID of the requesting node P_m is entered in the local queue of site for P_k
  - A new token request for node P_k is sent to its parent, if no such request is already pending

Major Steps in Raymond’s Algorithm

- Instead of an intermediate node P_k, if the recipient for a token request is P_{hold} itself, then only the following step is executed:
  - ID of the requesting node P_m is entered in the local queue of site for P_{hold}

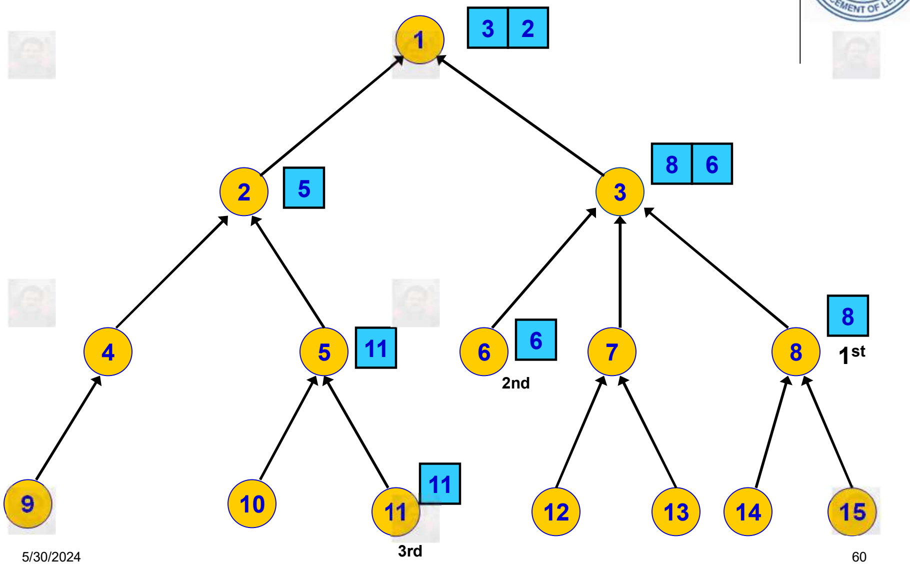
Major Steps in Raymond’s Algorithm

- When P_{hold} comes out of the CS, the following steps are performed:
  - An ID from its local queue is deleted, say X
  - The edge between P_{hold} and P_x is reversed in the spanning tree
  - The token is sent to P_x
  - If the local queue of P_{hold} is not empty then a token request is sent to P_x
  - P_x becomes the new P_{hold}

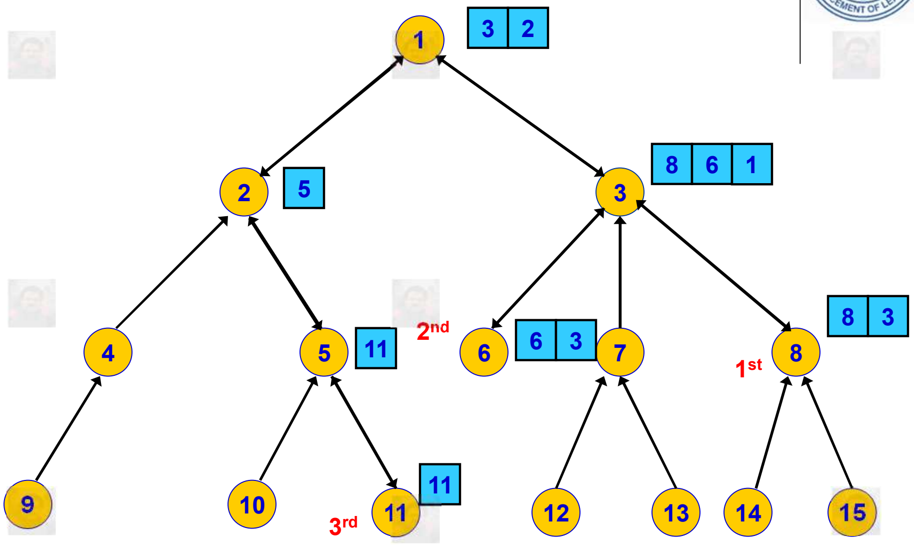
Major Steps in Raymond's Algorithm

- When node P_x gets token from parent node, the following steps are performed:
  - An ID from the local queue is deleted, say Z
  - If this ID is same as that for P_x itself (i.e., if X=Z as in this case) then
    - Node $P_X = P_Z$ enters the CS
  - If $X \neq Z$ then
    - the token will be passed to node P_Z
    - If the local queue at P_x is not empty then a token request is sent to P_Z

Raymond's Algorithm



Raymond’s Algorithm



5/30/2024

61

Safety in Raymond’s Algorithm

- Whenever a node receives a token (PRIVILEGE), it becomes privileged.
- Similarly, whenever a node releases the token, it becomes unprivileged.
- No node is privileged between the instants when one node releases token and the next node acquires it.
- Thus, there is at most one privileged node at any point of time in the system that can enter CS.

Thanks for your kind attention

Questions??

63