LECTURE 5

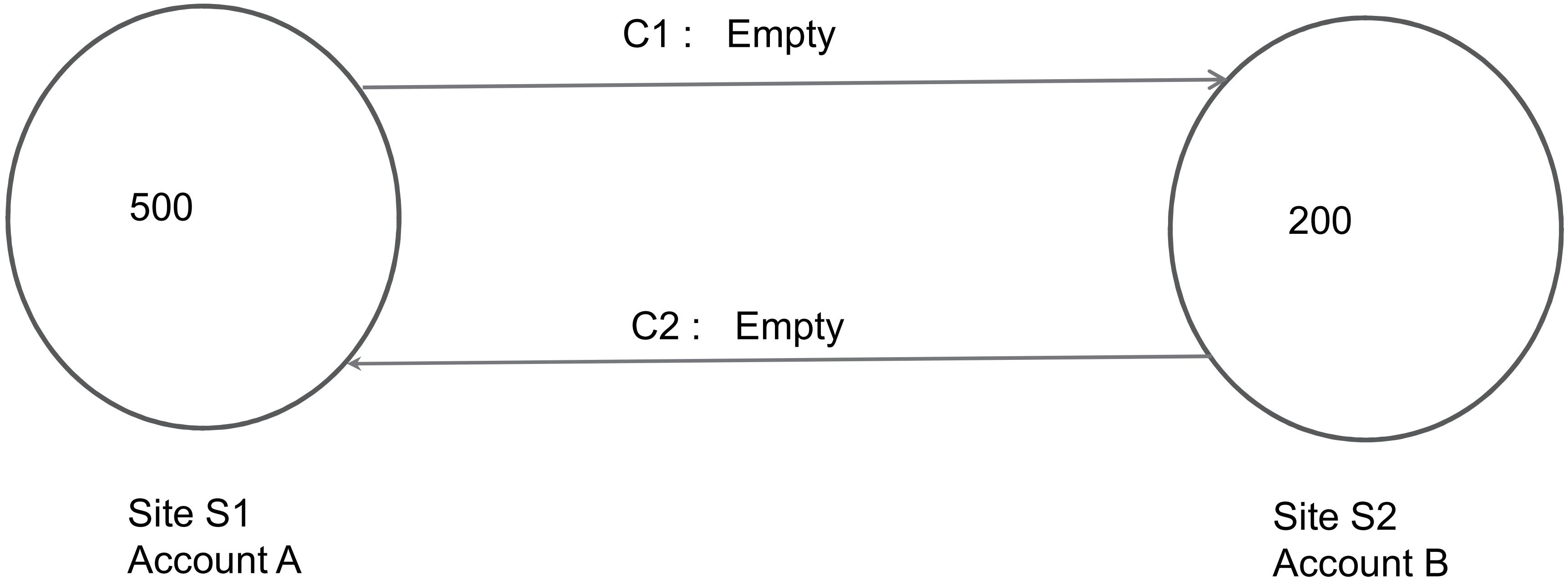
GLOBAL STATE AND CHECKPOINTS

Prof. D. S. Yadav
Department of Computer Science
IET Lucknow

INTRODUCTION

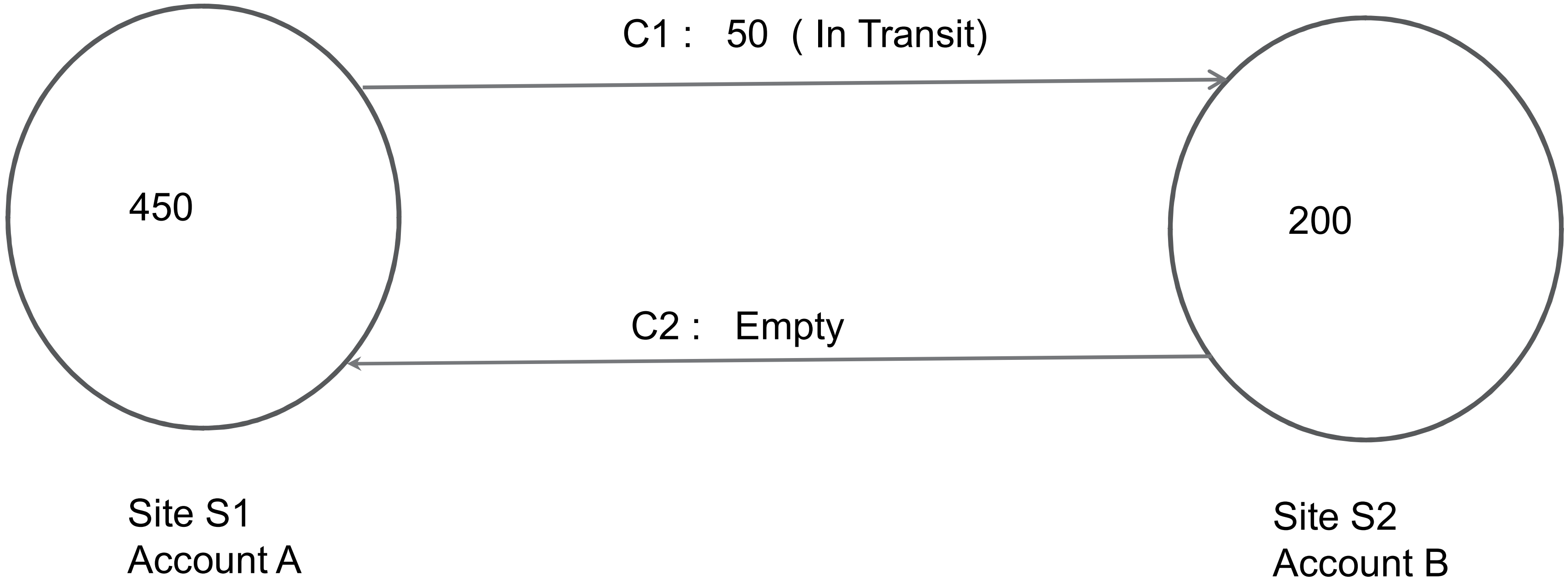
- Recording the global state of a distributed system on-the-fly is an important paradigm.
- The lack of globally shared memory, global clock and unpredictable message delays in a distributed system make this problem non-trivial.
- We first defines consistent global states and discusses issues to be addressed to compute consistent distributed snapshots.
- Then several algorithms to determine on-the-fly such snapshots are presented for several types of networks.

BANKING EXAMPLE : FUND TRANSFER



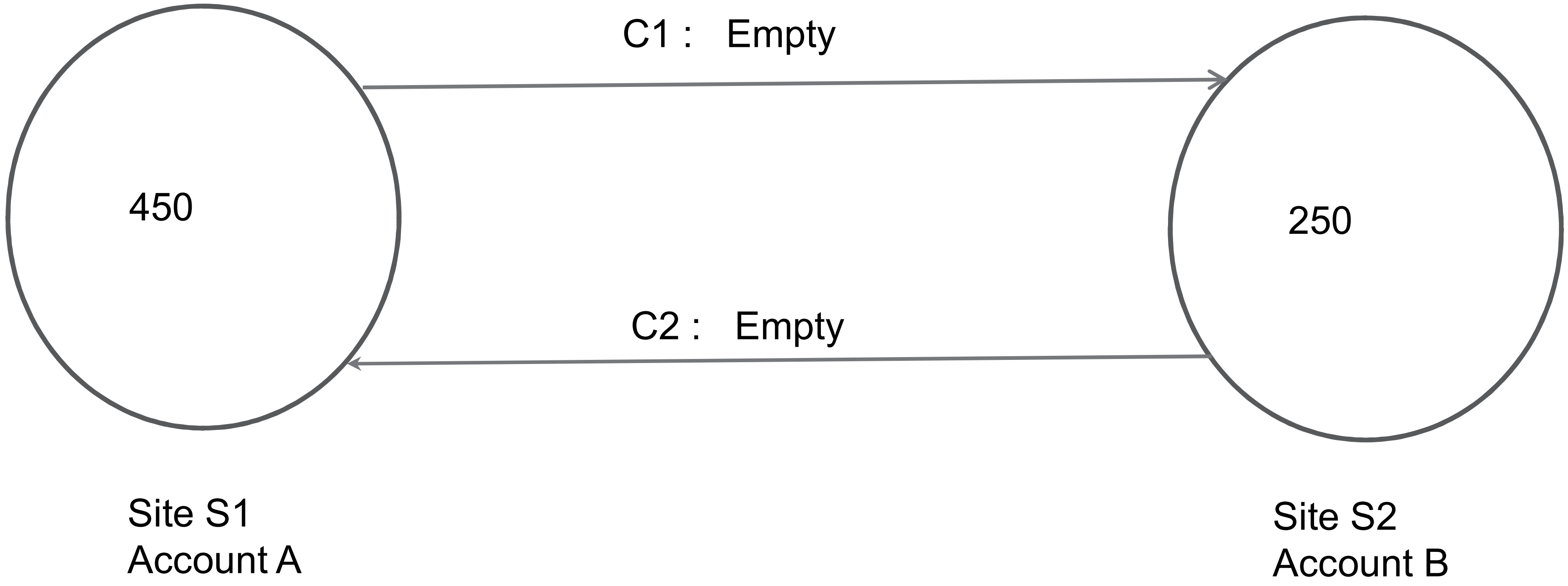
Global State A

BANKING EXAMPLE : FUND TRANSFER



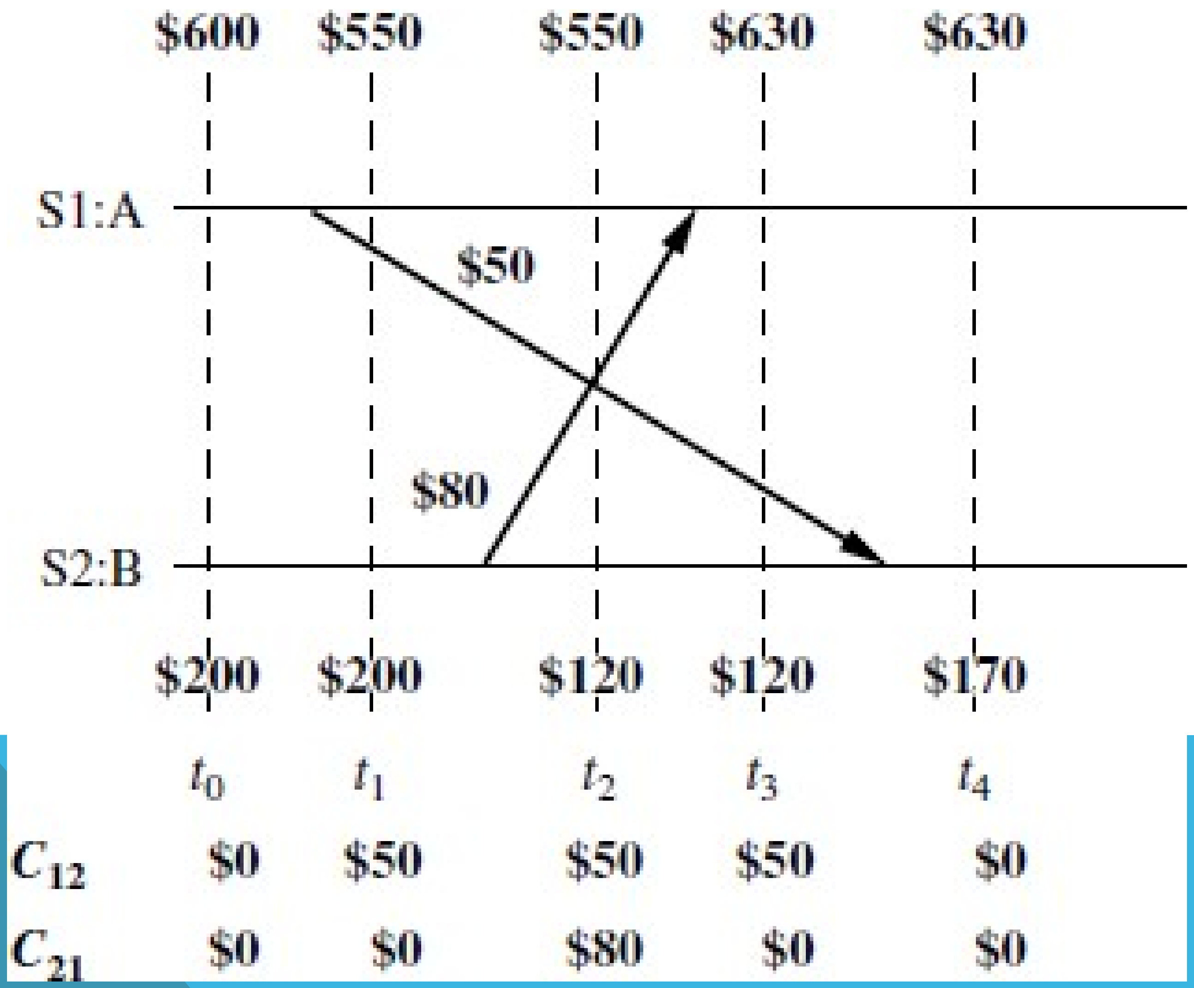
Global State B

BANKING EXAMPLE : FUND TRANSFER

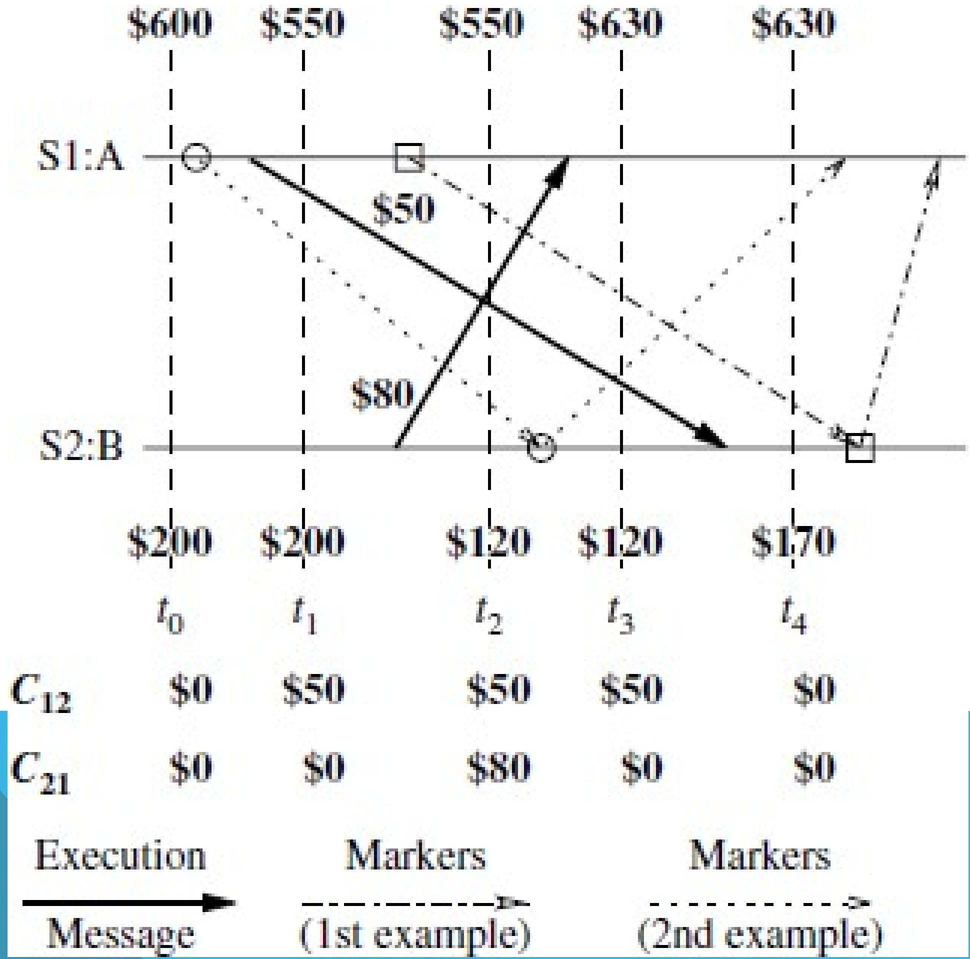


Global State C

SNAPSHOT ALGORITHM: BANKING EXAMPLE



SNAPSHOT ALGORITHM: BANKING EXAMPLE



SYSTEM MODEL

- The system consists of a collection of n processes $p_1, p_2, \ldots, p_n$ that are connected by channels.
- There are no globally shared memory and physical global clock and processes communicate by passing messages through communication channels.
- C_{ij} denotes the channel from process p_i to process p_j and its state is denoted by SC_{ij}.
- The actions performed by a process are modeled as three types of events: Internal events,the message send event and the message receive event.
- For a message m_{ij} that is sent by process p_i to process p_j, let $send\ (m_{ij})$ and $rec\ (m_{ij})$ denote its send and receive events.

SYSTEM MODEL

- At any instant, the state of process p_i, denoted by LS_i, is a result of the sequence of all the events executed by p_i till that instant.
- For an event e and a process state LS_i, $e \in LS_i$ iff e belongs to the sequence of events that have taken process p_i to state LS_i.
- For an event e and a process state LS_i, $e \notin LS_i$ iff e does not belong to the sequence of events that have taken process p_i to state LS_i.
- For a channel C_{ij}, the following set of messages can be defined based on the local states of the processes p_i and p_j

Transit: $transit(LS_i, LS_j) = \{m_{ij} \mid send(m_{ij}) \in LS_i \wedge rec(m_{ij}) \notin LS_j\}$

GLOBAL STATE COLLECTION

- **Applications:**
  - Checking "stable" properties, checkpoint & recovery

- **Issues:**
  - Need to capture both node and channel states
  - system cannot be stopped
  - no global clock

NOTATIONS

Some notations:

- LS_i: Local state of process I
- send(m_{ij}) : Send event of message m_{ij} from process i to process j
- rec(m_{ij}) : Similar, receive instead of send
- time(x) : Time at which state x was recorded
- time (send(m)) : Time at which send(m) occurred

DEFINITIONS

- $send(m_{ij}) \in LS_i$ iff $time(send(m_{ij})) < time(LS_i)$
- $rec(m_{ij}) \in LS_j$ iff $time(rec(m_{ij})) < time(LS_j)$
- $transit(LS_i, LS_j)$

  $= \{ m_{ij} \mid send(m_{ij}) \in LS_i \text{ and } rec(m_{ij}) \notin LS_j \}$
- $inconsistent(LS_i, LS_j)$

  $= \{ m_{ij} \mid send(m_{ij}) \notin LS_i \text{ and } rec(m_{ij}) \in LS_j \}$

DEFINITIONS

- Global state: collection of local states

  $$GS = \{LS1, LS2, \ldots, LSn\}$$

- GS is consistent iff

  for all i, j, $1 \le i, j \le n$,

  $$inconsistent(LSi, LSj) = \Phi$$

- GS is transitless iff

  for all i, j, $1 \le i, j \le n$,

  $$transit(LSi, LSj) = \Phi$$

- GS is strongly consistent if it is consistent and transitless.

MODELS OF COMMUNICATION

- Recall, there are three models of communication: FIFO, non-FIFO, and CO.
- In FIFO model, each channel acts as a first-in first-out message queue and thus, message ordering is preserved by a channel.
- In non-FIFO model, a channel acts like a set in which the sender process adds messages and the receiver process removes messages from it in a random order.
- A system that supports causal delivery of messages satisfies the following property:

"**For any two messages m_{ij} and m_{kj},**

if $send(m_{ij}) \rightarrow send(m_{kj})$, then $rec(m_{ij}) \rightarrow rec(m_{kj})$".

CONSISTENT GLOBAL STATE

- The global state of a distributed system is a collection of the local states of the processes and the channels.
- Notationally, global state *GS* is defined as,

  $GS = \{ \cup_i LS_i , \cup_{i,j} SC_{ij} \}$

- A global state *GS* is a *consistent global state* iff it satisfies the following two conditions :

  C1: $send(m_{ij}) \in LS_i \Rightarrow m_{ij} \in SC_{ij} \oplus rec(m_{ij}) \in LS_j$.

  ($\oplus$ is Ex-OR operator.)

  C2: $send(m_{ij}) \notin LS_i \Rightarrow m_{ij} \notin SC_{ij} \wedge rec(m_{ij}) \notin LS_j$.

INTERPRETATION IN TERMS OF CUTS

- A cut in a space-time diagram is a line joining an arbitrary point on each process line that slices the space-time diagram into a PAST and a FUTURE.
- A consistent global state corresponds to a cut in which every message received in the PAST of the cut was sent in the PAST of that cut.
- Such a cut is known as a *consistent cut.*

- For example, consider the space-time diagram for the computation illustrated in Figure 1.
- Cut C1 is inconsistent because message m1 is flowing from the FUTURE to the PAST.
- Cut C2 is consistent and message m4 must be captured in the state of channel C_{21}.

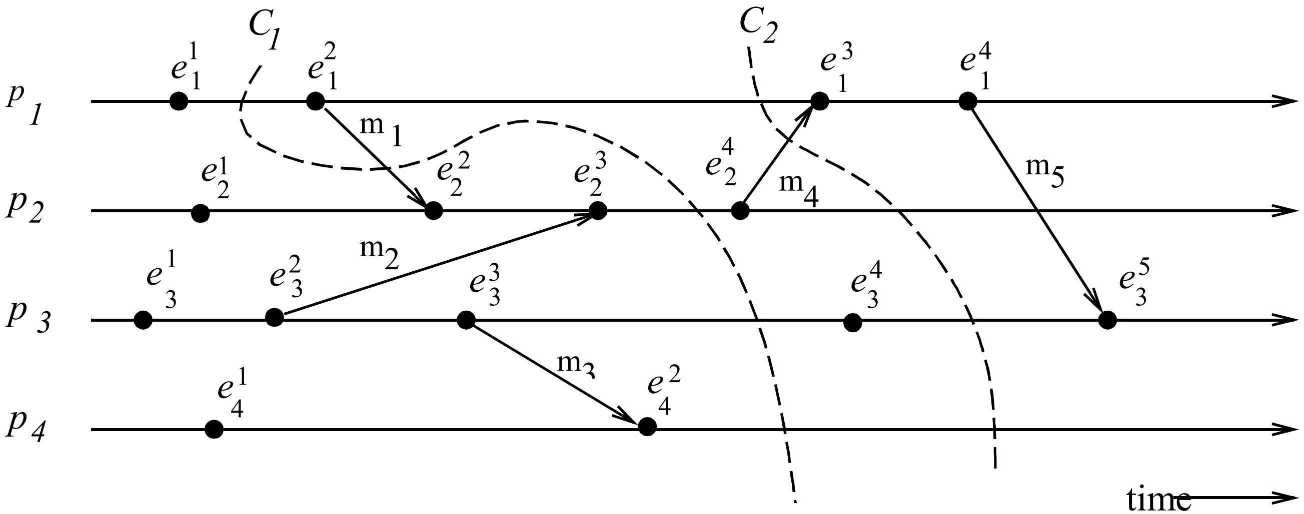


Figure 1: An Interpretation in Terms of a Cut.