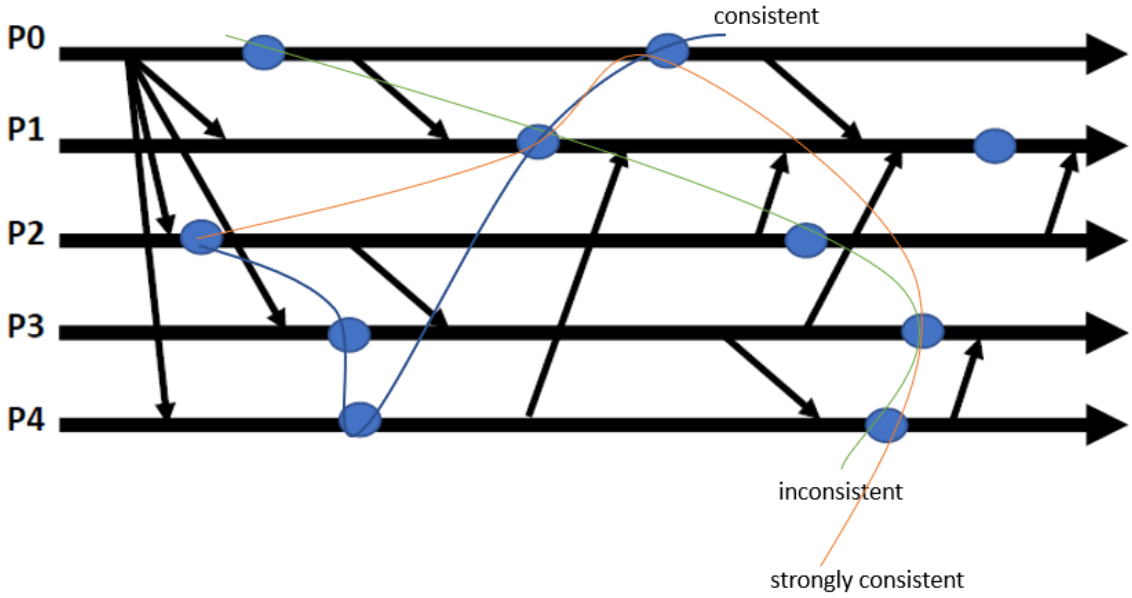
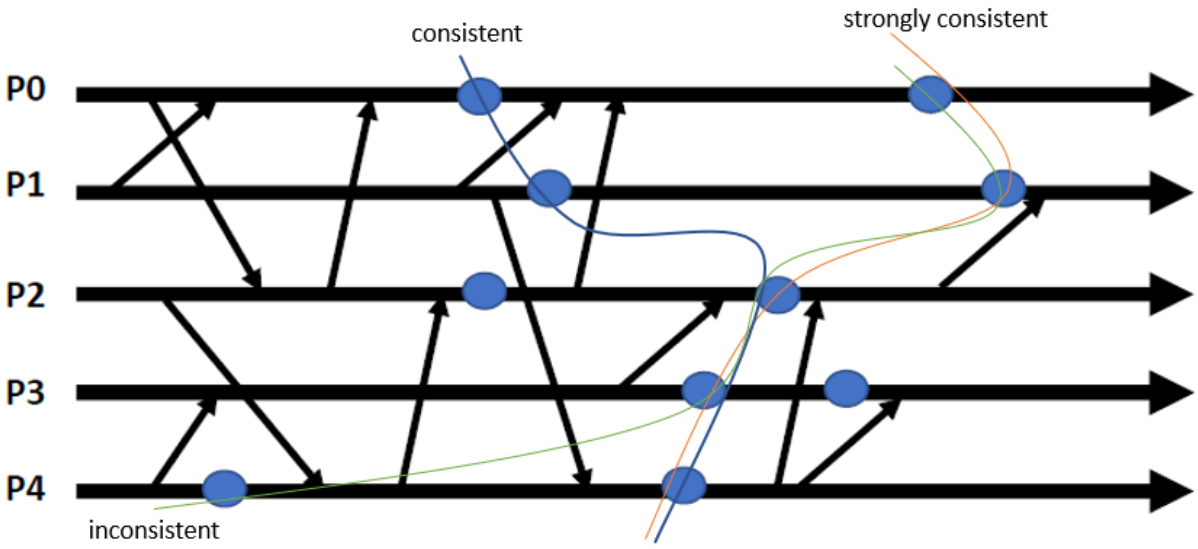
Eric Paulz
ECE 4930-004

HW2

1. Draw the location of checkpoints to create one strongly consistent cut, one consistent cut, and one inconsistent cut. Clearly label each cut.



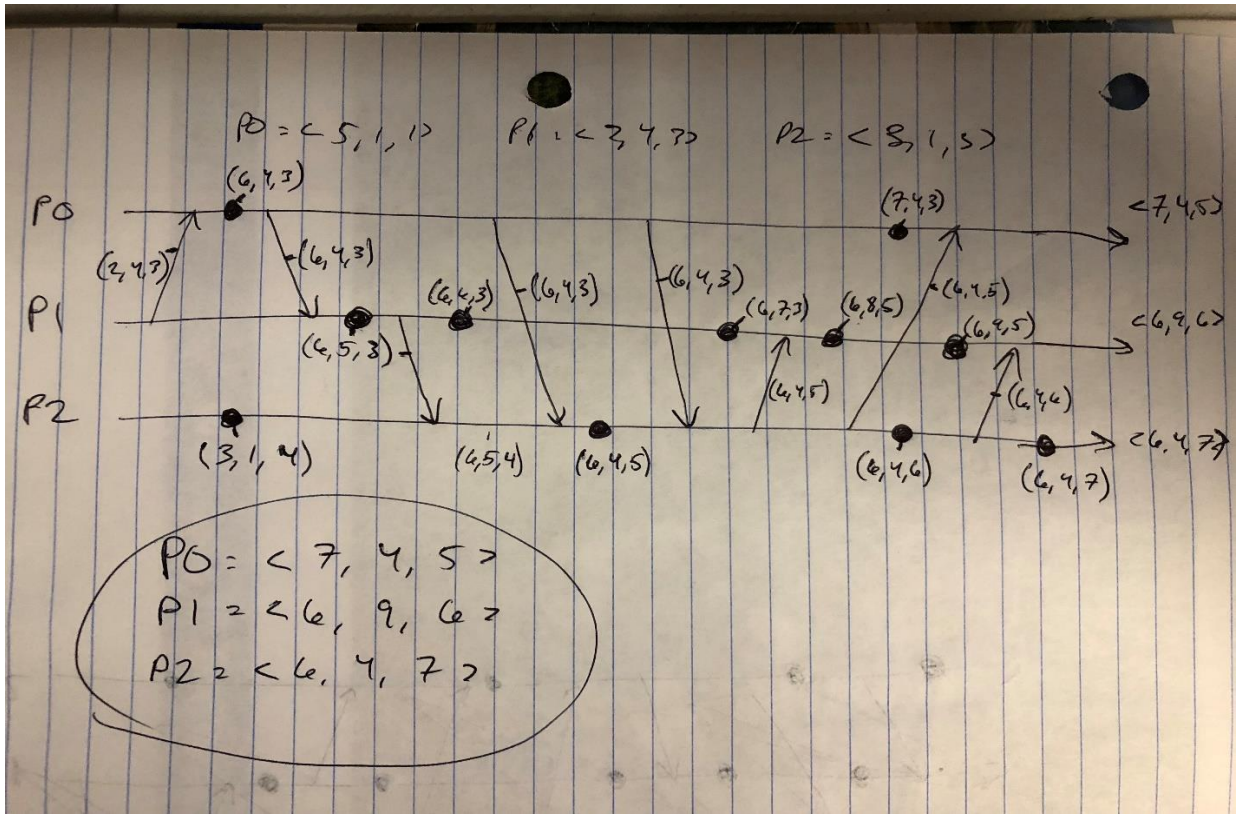


2. What is the *domino effect*, and why is it an important concern when checkpointing? Clearly state what causes it and discuss a solution to mitigate it.

   The domino effect is a problem that arises when uncoordinated checkpointing is utilized. Specifically, it occurs when the rollback of some process forces rollback of other processes beyond the most recent checkpoint. This compounding rollback is thus likened to the falling of a row of dominos. This means that the system must keep older checkpoints in order to get a consistent cut.

   Since uncoordinated checkpointing eliminates marker messages, there has to be some other way to recover older checkpoint information in case a process fails and has to restart without affecting other processes. To achieve this, we can use message logging. Message logging provides a way to keep all processes in sync without requiring all-to-all marker messages, and therefore helps to mitigate the domino effect issue.

3. Given the initial vector timestamps for the three processes below, what is the final vector timestamp for each process? Countable events include: message sends, message receives, and local events (circles). Show your work for partial credit.



4. Sending marker message is expensive as it is essentially an all-to-all operation that does not scale. However, using marker messages allows the processes to coordinate to obtain a consistent checkpoint. Create and describe a checkpointing scheme that reduces the overall number of marker message. Note: your scheme must still use marker messages to initiate the checkpointing operation. Does your scheme produce inconsistent, consistent, or strongly consistent checkpoints?