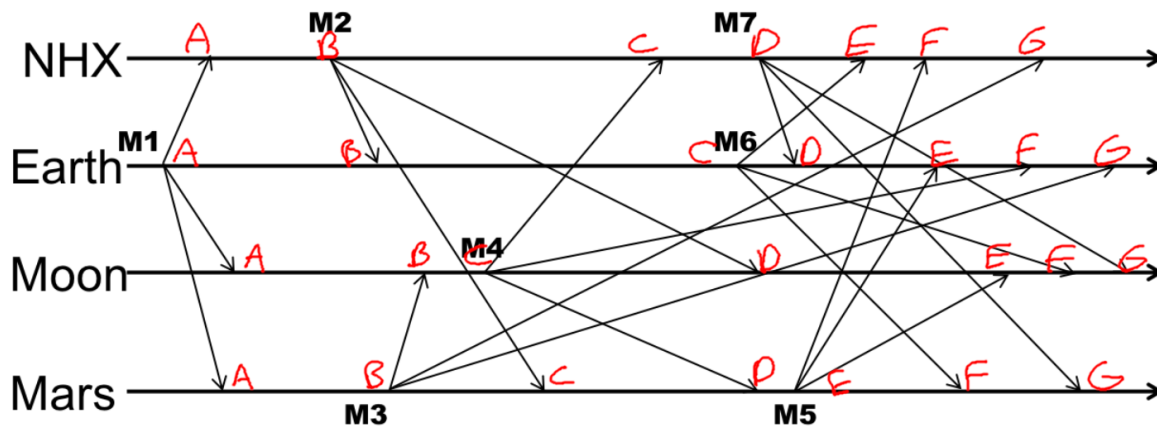
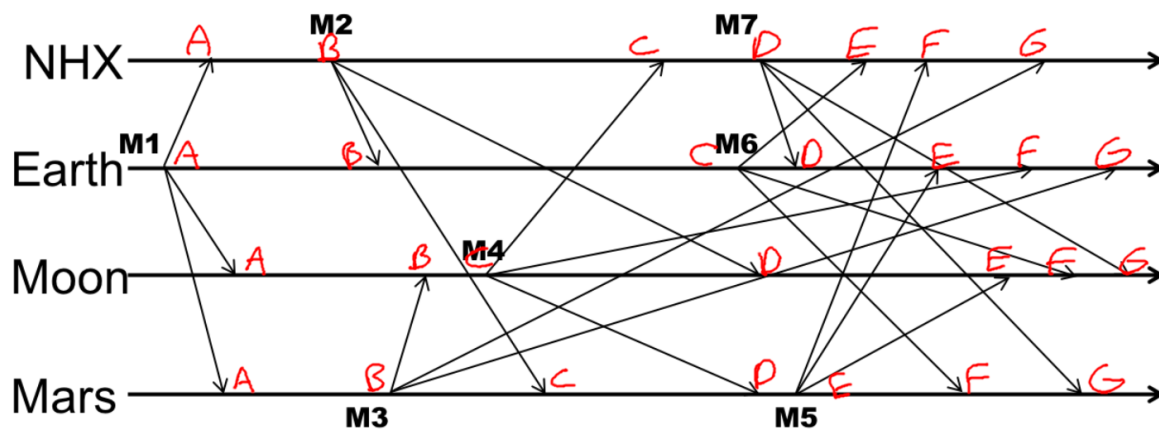


**Problem 2:**


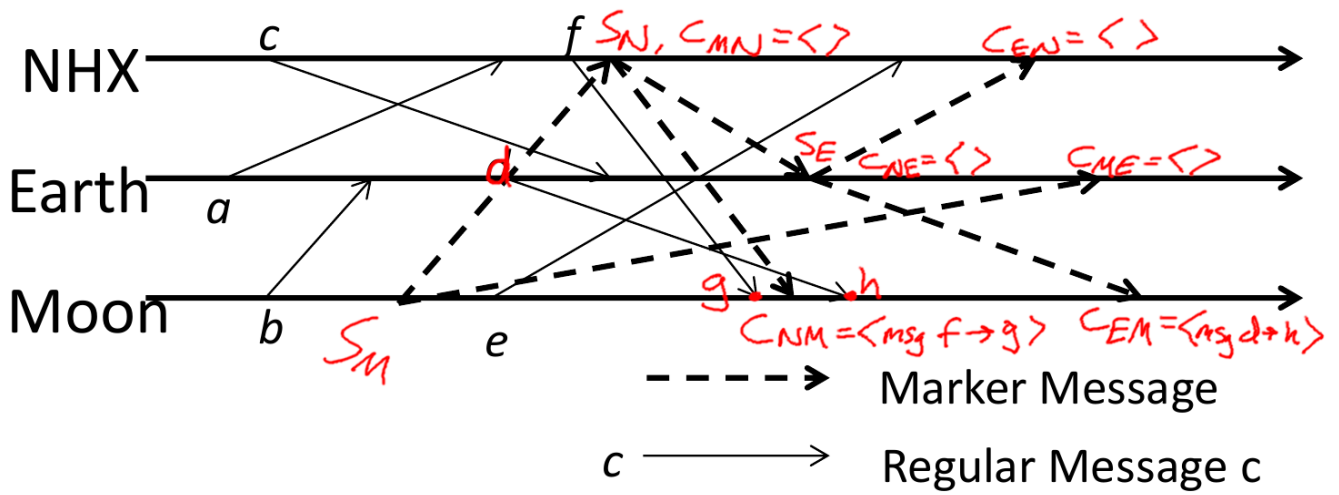
	NHX	Earth	Moon	Mars
A	0,1,0,0	0,1,0,0	0,1,0,0	0,1,0,0
B	1,1,0,0	1,1,0,0	0,1,0,1	0,1,0,1
C	1,1,1,0	1,2,0,0	0,1,1,1	1,1,0,1
D	2,1,1,0	2,2,0,0	1,1,1,1	1,1,1,1
E	2,2,1,0	2,2,0,0 B<Seq 4:2>	1,1,1,2	1,1,1,2
F	2,2,1,0 B<Seq 4:2>	2,2,1,0	1,2,1,2	1,2,1,2
G	2,2,1,2 D<Seq 4:2>	2,2,1,2 D<Seq 4:2>	2,2,1,2	2,2,1,2

Note: B → Buffer, D → Delivery

**Problem 3:**

	NHX	Earth	Moon	Mars
A	0,1,0,0	0,1,0,0	0,1,0,0	0,1,0,0
B	1,1,0,0	1,1,0,0	0,1,0,1	0,1,0,1
C	1,1,0,0 B<M 3: 0,1,1,1>	1,2,0,0	0,1,1,1	1,1,0,1
D	2,1,0,0	2,2,0,0	1,1,1,1	1,1,1,1
E	2,2,0,0	2,2,0,0 B<M 4: 1,1,1,2>	1,1,1,2	1,1,1,2
F	2,2,0,0 B<M 4: 1,1,1,2>	2,2,0,0 B<M 3: 0,1,1,1>	1,2,1,2	1,2,1,2
G	2,2,0,1 D<M 3: 0,1,1,1> → 2,2,1,1 D<M 4: 1,1,1,2> → 2,2,1,2	2,2,0,1 D<M 3: 0,1,1,1> → 2,2,1,1 D<M 4: 1,1,1,2> → 2,2,1,2	2,2,1,2	2,2,1,2

**Problem 6:**



**Problem 4:**

$M_{11} \rightarrow M_{12} \rightarrow M_{21} \rightarrow M_{31} \rightarrow M_{32} \rightarrow M_{33} \rightarrow M_{22} \rightarrow M_{23} \rightarrow M_{13}$

or

$M_{11} \rightarrow M_{12} \rightarrow M_{21} \rightarrow M_{31} \rightarrow M_{32} \rightarrow M_{33} \rightarrow M_{22} \rightarrow M_{13} \rightarrow M_{23}$

**Problem 1:**

Consider minimum possible number of machines for a system,  $N = 6$  (first integer greater than 5). The maximum number of failures for this system will be at most  $f = N = 6$ . If multicasts in rounds 3 and 4 never get delivered, then it's as if those rounds never happened. In other words, there are really only 4 rounds of multicasts (round 1, 2, 5, 6).

Thus, it is possible for a process  $p_i$  to have a value  $v$  that  $p_j$  does not have as there would be  $f - 2 = 4$  failures, which does not go against our assumption of max failures of  $f = 6$ . Therefore all runs of this algorithm will not all still be correct.

**Problem 5:**

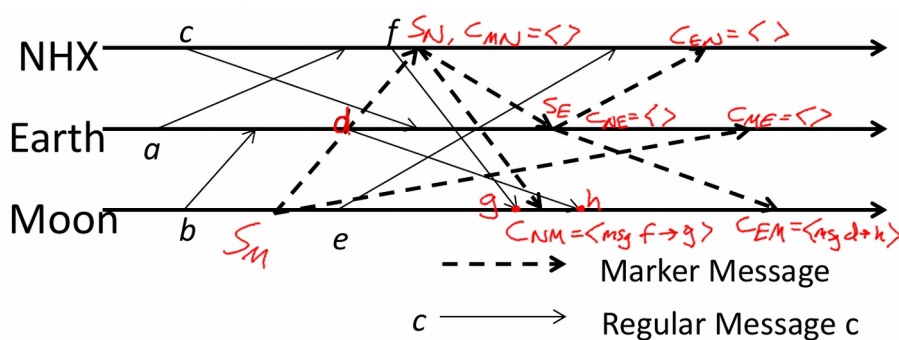
- i. This new version is still safe, and possesses eventual liveness. While there can be more than 1 leader, the Bill phase still requires a majority and hence prevents inconsistencies. Since fewer votes are required to become a leader and there could be more than 1 leader working the Bill phase. However, each process can only vote once, so the votes could be split among the multiple leaders. Because of this, this version will be slower than the original version as it could prove challenging for a leader to get a majority of votes on a Bill. If each process is allowed to submit a vote to each leader, then it would be faster than the original version as it takes less to determine a leader and if there's more than one leader then the Bill and Law phases could go quicker with two processes coordinating them.
- ii. This new version is not safe and possesses a much weaker level of liveness (there is always a chance that they could all agree eventually). Since a majority is no longer required in the Bill phase, inconsistencies can arise. If it doesn't get lucky in the beginning, this will be a lot slower to complete than the original version as it tries to sort out the inconsistencies.
- iii. This new version is no longer safe and possesses a much weaker level of liveness (there is always a chance that they could all agree eventually). There can be more than one leader and there's a chance that each leader determines two different values since the Bill phase no longer requires the majority. As such inconsistencies can arise. If inconsistencies arise in the beginning, this version will be much slower at reaching consensus. If it gets incredibly lucky and doesn't encounter any inconsistencies, it could be faster than the original version since there are two leaders and they require less votes to get things done.

**Problem 9:**

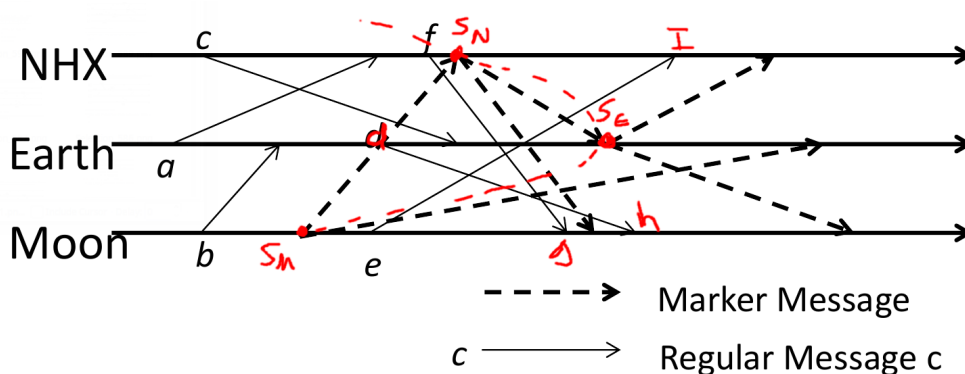
- a. This is **incorrect**, multicast M32 needs to be delivered before the new view V12 is delivered. What happens in the view stays in the view.
- b. This is **correct**. While p1 multicasts M32, it is never delivered to any process. It is as if M32 was never sent in the first place and so virtual synchrony is satisfied.
- c. This is **incorrect**. P2 needs to deliver M32 before the new view, V12, is delivered. What happens in the view stays in the view.
- d. This is **incorrect**. P1 and P2 need to deliver their own messages to themselves or not deliver theirs to other processes at all.
- e. This is **incorrect**. Either P1 should not deliver M32 to itself, or P2 and P3 need to deliver M32 before their new views are delivered.
- f. This is **incorrect**. M32 should only be delivered to p1, p2, and p3 within their current view prior to the new view V12 being delivered.

**Problem 10:**

Consider the example from problem 6:



Here is the same timeline but with a few more labels and the cut made by  $S_N$ ,  $S_M$ , and  $S_E$  drawn with a dotted red line:



i. With the algorithm presented in problem 10, Msg  $e \rightarrow I$  would be included in the channel  $C_{MN}$  in the cut made by  $S_N$ ,  $S_M$ , and  $S_E$ . But this violates the consistent cut property as  $S_M \rightarrow e$  and  $S_N \rightarrow I$  and  $e \& I$  are both not in the cut made by  $S_M$ ,  $S_N$ , and  $S_E$ . **As such, the algorithm presented in problem 10 is not correct.**

ii. The algorithm is very similar to Chandy-Lamport's algorithm. The differences are each process records all incoming channels upon first Marker message and it doesn't stop recording until the last marker message is received. This causes messages sent after the snapshot initiation to be recorded, which is incorrect. To correct the algorithm, make changes to make it the same as Chandy-Lamport's algorithm. Such as, make processes that receive their first Marker message to only start recording incoming channels that it did not receive the marker from and to close the channels and stop recording once the respective channel receives its second marker message for that corresponding channel.

iii. Colonizing Mars is impractical, irresponsible, and wasteful. We can barely manage our current dwindling resources to support our current world population (people are still suffering from hunger, climate change, etc). Why should we waste already dwindling resources on colonizing a frozen wasteland that is nearly 100% inhospitable to life? I say let's instead invest our current resources towards better detection and space travel technologies that will enable us to detect and then travel to actual habitable planets elsewhere in the galaxy. Forget mars, let's go back to Earth and start researching FTL (faster than light) travel while also solve the current problems we face on Earth, such as climate change.