Total Order Multicast Proof

Siddhant Dilip Godshalwar

March 1, 2023

1 Introduction

A communication technique called total order multicast is employed in distributed systems to guarantee that messages are sent to various receivers in a certain order. A message is multicast via this protocol to a number of receivers, and each recipient is assured to get the message in the same sequence..

2 Proof:

The total order property, using the happens-before relationship, is that for any two messages m1 and m2, either m1 happened before m2 (m1 ->m2), or m2 happened before m1 (m2 ->m1), or m1 and m2 are concurrent (m1 || m2). If m1 ->m2, then every process that delivers m2 must have delivered m1 or any other message that happened before m1. If m2 ->m1, then every process that delivers m1 must have given m2 or any other message that happened before m2. If m1 || m2, then there is no constraint on the order in which the messages are delivered.

- 1 Let's suppose that messages M1 and M2 exist, but that M2 is delivered prior to M1. This means that each process that sends message m2 must also send message m1 or any other message that occurred before to message m1. The procedure that provided (m2,s2) before receiving (m1,s1), where s1 s2, must have occurred if m2 is given before m1. Due to the rigorous rising order in which sequence numbers are allocated to messages, if m2 has a higher sequence number than m1, it should only be sent after m1.

- 2 This goes against the algorithm's core presumption that messages should be ordered according to first-in, first-out (FIFO). According to the FIFO property, if a process

```
Upon broadcasting a message m:
    Increment a sequence number s.
    Attach the sequence number to the message and multicast (m,s) to all processes.

Upon receiving a message (m,s) from process p:
    Deliver (m,s) only if s is the next expected sequence number for m.
    Otherwise, store (m,s) in a buffer until all messages with a lower sequence number have been delivered.
    When (m,s) can be delivered, check the buffer to see if any other messages can be delivered in sequence order.
```

Figure 1: Pseudocode for the proof

sends message m1 before m2, m1 was broadcast before m2. In other words, information is conveyed in the sequence in which it was transmitted. As a result, the FIFO characteristic is broken by the expectation that m2 is delivered before m1.

- 3 We may infer that such a scenario is implausible since the assumption that m2 being delivered before m1 would violate the FIFO principle, a key tenet of the algorithm. This means that for any two messages, m1 and m2, one of the following is true: m1 happened before m2 (m1 ->m2), m2 happened before m1 (m2 ->m1), or m1 and m2 are concurrent (m1 || m2).