CSCI 251: Concepts of Parallel and Distributed Systems

Alvin Lin

November 1st, 2017

Coordination and Agreement

Topics:

- Election Algorithm
- Multicast
- Agreement

Bully Algorithm

A process is elected as the leader/coordinator based on its number/label. Any process can start the election process.

Multicast

The message m is sent to a group g of processes. The basic multicast involves a process performing a multicast to all connected processes. Those processes will propagate the message to all of its connected processes. This is inefficient since a network of processes can send and acknowledge duplicate messages. Reliable multicast is a variation that ensures some other basic properties are satisfied.

- Integrity: A correct process delivers m exactly once.
- Validity: If a correct process multicasts m, then it will eventually deliver m.
- Agreement: If a correct process delivers m, then every correct process in the group g will also deliver m.

Piggyback Acknowledgements

Along with (m, g), a process lets other processes know the sequence number of the multicast message for a process $\langle (m, g), S_g^p \rangle$. A message $\langle q, R_g^q \rangle$ is sent to each process such that $q \in g, q \neq p$. R_g^q is nothing more than the sequence number of the last multicast message from q that was delivered at p. For a receiving process r, R_g^p is the last multicast message from p that r has delivered, and R_g^q is the last multicast message from q that r has delivered.

- $S = R_g^p + 1$, deliver m
- $S \leq R_g^p$, ignore m since it already has been delivered
- $S > R_g^p + 1$, process r has missed a previous message, so the current message is placed in a holdback queue and negative acknowledgement is sent to p

Ordering of Multicast Message

FIFO: if a process sends (m, g) before (m', g), then $d(m)$ happens before $d(m')$.
Causal Ordering: multicast message (m, g) is the cause of another multicast message (m', g) and therefore $d(m)$ happens before $d(m')$.

Reminders

Check MyCourses for details on Project 2.
Professor Mohan Kumar:
`mjkvcs@rit.edu`
`https://cs.rit.edu/~mjk`

Rahul Dashora (TA):
`rd5476@mail.rit.edu`

You can find all my notes at `http://omgimanerd.tech/notes`. If you have any questions, comments, or concerns, please contact me at alvin@omgimanerd.tech