CS425 MP2 Report Wenhao Su (wenhaos3) Yichen Yang (yy18)

Distributed Group Membership Design:

In this project, each server maintains a membership list, where each entry in the membership list has three elements: the ip address of this member, the latest timestamp received from this member and the latest updated time of receiving the message from this member.

All the members in the group forms a virtual ring, with the position in the ring decided by the alphabetical size of their ip address string. Every member will send heartbeat to its next three successors, and monitor the heartbeat from its previous three predecessors.

Initially all servers are not in the group, and they must send a message with overhead "JOIN" including its ip address to the indicator (including the indicator itself). Then the indicator would return a message with overhead "JOIN-SUCCESS" and the entire membership list of the indicator back to the sender of "JOIN". The indicator will also send a message with overhead "ANNOUNCE" to its successors to notify them that a new machine has joined the group.

Every 0.5 second, a server will send a heartbeat message with its latest timestamp to its successors. If a machine voluntarily leaves the group, or a machine founds that its predecessor has not send him heartbeat for more than 4 seconds, it will send a "LEAVE"/"FAILURE" message to its successors and traverse among the living machines in the group. Upon receive a new this kind of message, the server will remove the leaved or failure machine from their membership list.

The use of MP1:

We need to use log files to debug the whole system since we cannot use IDE or other debuggers to debug it. What we can do is to write down different logs to see which part goes wrong. Therefore, it is very useful to use MP1 to get the log files and read it in one machine to see where goes wrong.

measurements:

i. background bandwidth usage

heartbeat rate = 0.5s / one

every heartbeat package size = 32 Bytes

every time each member will send 3 packages

so the background bandwidth is:

$$total\ bandwidth = \left(\frac{1}{0.5}\right) * 6 * 32 * 3 = 1152\ Bytes/s$$

ii. average bandwidth usage

joins:

join will cause two kinds of messages: join and announcement

for a join, it will cause 32 Bytes + 192 Bytes to let indicator know and joiner get the membership list

for announcement, the message is 32 Bytes, and every node will send 3 packages to the successor:

so total:

$$average\ bandwidth = 32 + 192 + 32 * 3 * 6 = 800\ Bytes/join$$

leaves:

for a leave, each package is 32 Bytes, and every node will send 3 packages to the successor:

so total:

$$average\ bandwidth = 32 * 3 * 6 = 576\ Bytes/leave$$

fails:

for a fail, each package is 32 Bytes, and every node will send 3 packages to the successor:

so total:

$$average\ bandwidth = 32 * 3 * 6 = 576\ Bytes/fail$$

iii. False positive rate of your membership service

Here we define the false positive rate as the following:

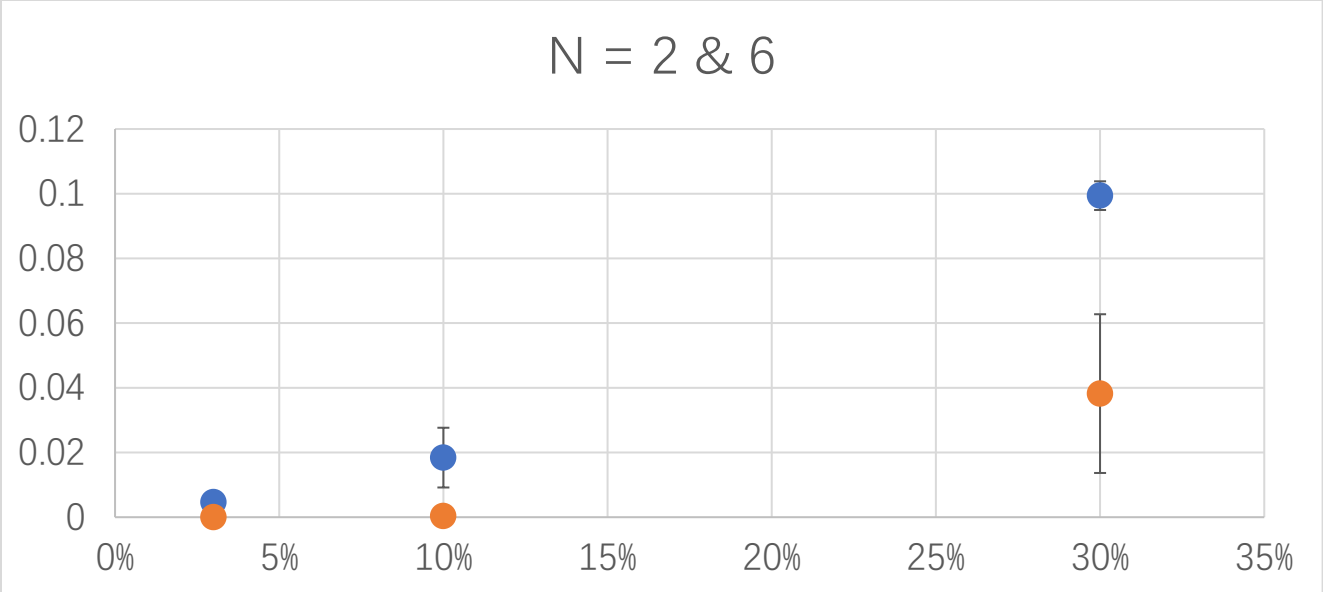
$$\frac{f}{n+f}$$

where f is the failure detection number and n is the total heartbeat number.

Here is the result:

N	Loss	average false positive rate	standard deviations	95% confidence intervals
6	30%	0.09939893	0.00443692	0.00388906
6	10%	0.01842033	0.0092337	0.0068403
6	3%	0.00464622	0.00159592	0.00127698
2	30%	0.03820655	0.02453357	0.02150422
2	10%	0.00036757	0.00034242	0.00030013
2	3%	0	0	0

Here is the plot:



where the blue dots represent the N = 6 and orange dots represent N = 2.

From the plot we can see that the trend of both N = 6 and N = 2 is: when the Loss Rate of UDP increases, the average false positive rate will also increase. This follows our expectation since with higher Loss Rate of UDP, we will expect to get more heartbeats missing. Therefore, we may miss too much heartbeat and treat a live member as failed.

For N = 6 and N = 2, we can see the trend of when compare between these two number of members, N = 6 will have larger false positive rate and N = 2 will have smaller one. This also follows our expectation since for N = 2, they will only focus on listening one other machine, so the rate of treating a live member as failure will be k^t, where k is the loss rate and t is how much round of unreceiving heartbeat will lead to failure. However, for N = 6, it will more possible to reach a failure, once a failure is detected, it will be sent to the successors. When the successor receives a failure, it will also report a failure. Therefore, it will cause higher false positive rate. In addition, with more communications for one member, it will have higher rate for missing.