Distributed Systems Exercise 3 Design Document

Minqi Ma (JHED: mma17), Jerry Chen (JHED: gchen41)

Data Structures

- *bool finished[num_processes]*: for each machine i, if current machine has received the last packet from machine i
- *int num_delivered*: the number of messages that the current machine has successfully delivered
- *int num_sent*: the number of messages that the current machine gives to SPREAD

Message formats

- unsigned int process_index
- unsigned int message_index
- unsigned int random_number
- int payload[1300 / sizeof(int)]

Note: every regular message also has a mess_type

- mess_type = TAG_DATA: it is a normal message with data information
- mess_type = TAG_END: it is the last packet sent by the process

Design

1. Upon start of the program, send membership message to join the group.
2. Once received a membership message, check if num_groups equals num_processes. If so, all processes have joined and we can start sending messages. The current process first sends INIT_SEND_SIZE packets, so SPREAD has enough to work with.
3. When receiving a regular message
    a. If mess_type is TAG_DATA
    Deliver the information to output file.
    Increment num_delivered if the packet is from the current machine.
    Every time when num_delivered is increased by SEND_SIZE, it means that SPREAD has successfully delivered this amount of current process's packets to all processes. So we send SEND_SIZE number of packets to SPREAD.
    b. If mess_type is TAG_END
    Mark the machine as finished in the *finished* array.
    If all processes are finished, exit the program.

Tuning parameters:

- INIT_SEND_SIZE = 65
- SEND_SIZE = 1

Performance

	1	2	3	4	5	AVERAGE
Performance(s)	28.1	28.41	26.13	27.86	38.02	29.704

Discussion

1. If we send too many packets initially, SPREAD will overflow, which makes the performance worse.

2. We would like to control two parameters so that it can give SPREAD enough work to do (depending on INIT_SEND_SIZE), and send more packets if SPREAD has less message “on the way” (depending on SEND_SIZE).
3. Sometimes we got inconsistent results even with the same parameter. We are not sure why it happens. It can vary from 25-50s.