Report for Experimental Evaluation of FloodSet and OptFloodSet Algorithm

Zhiyong Liu 201298442

COMP212 Distributed System,

Assignment 2

***Abstract*—FloodSet and OptFlood are two different kinds of agreement algorithm in distributed system, which have some differences. The experimental data of these two algorithms are shown in this report, and the performance, such as communication complexity is also analyzed in this report.**

I. INTRODUCTION

The purpose of this report is to show the experimental data of FloodSet and OptFlood algorithm simulated through the java program. First, it will describe the correctness of the simulator. Then the performance of these two algorithms are shown and analyzed in this report.

This report is organized as follow. Section II describes the simulator of these two algorithms. In Section III, it shows the experimental results using the different test data. Section IV analyses the performance of these two algorithms and compares with them. Section V is the conclusion.

II. OVEW OF THE SIMULATOR

In this part, the report will describe structure of the simulator. It will also describe how the FloodSet and OptFloodSet algorithm are implement in the simulator. Then, the correctness of implementations of these two algorithms will be proved.

A. Structure of Simulator

The simulator starts with asking user to input the algorithm selected, FloodSet or OptSet. Then second input to be asked is the processor number. The third parameter which user should input is f, which refers to the maximum number of failed processors during the execution. After simulating the selected algorithm, the output information is as follow:

Number of Processors: processorAmount

Number of Decided Processors: processorAmount

Number of Failed Processors: processorAmount

Number of Round: roundAmount

Number of Message: messageAmount

Satisfy Agreement: Boolean

Agreement Decision: String

These output information is used in the analysis of performance of these two algorithms. The simulator has 6 classes, which are described as follow:

- **Main:** It is the entrance of the simulator. It implements the algorithm, number of processors, maximum number of failed processors setting function. Then it creates a simulator according to the selected algorithm and run it.

- **Processor:** It is the class of Processor, which is used to simulator the processor. It has two kinds of running methods according to the selected algorithm.

- **Simulator:** This class is used to simulate the FloodSet and OptFloodSet simulator. It first initializes the

simulator configuration, which implements the network structure. Then it runs the simulator according to the selected algorithm. After selecting the algorithm, it prints the results.

- **Round:** This class is used to simulate the round in both algorithm. Each round the processor will do some actions.

- **Message:** It is used to simulate the message in the distributed network. It has an attribute called value which is an arraylist contains strings.

B. Correctness

In this part, it will prove that the simulation of FloodSet algorithm and OptFloodSet algorithm is correct. It means that eventually agreement is satisfied after executing the algorithm.

Prove in Theory:

For both FloodSet algorithm and OptFloodSet algorithm, it will execute at most f + 1 rounds, which f is the maximum number of failed processors. In the round f + 1, if *Wi* of processor only have one value, it will decide this value. Otherwise, it will decide the default value. So, these two algorithms satisfy the Termination. All processors in the network will know the value of other nodes before round f + 1, so in the round f + 1, all the nodes will have the same *Wi,* it means they will make the same decision, which satisfy the validity and agreement.

Prove in Experimentation:

The simulator output some information which can prove that the simulation is correct, just as follow:

```
Simulator results:
Number of Processors: 5
Number of Decided Processors: 4
Number of Failed Processors: 1
Number of Rounds: 3
Number of Messages: 50
Satisfy Agreement: true
Agreement Decision: Abort

Fail processor:
Pro_3

Processor decision:
Pro_1:Abort Pro_2:Abort Pro_3:null Pro_4:Abort Pro_5:Abort
```

Fig. 1. The output information in one siumlation

The output information has an option “Satisfy Agreement”, it is implemented by a function which checks all the decisions of processors after executing the algorithm. If the result is true, which will prove the agreement is satisfied. Besides, the last line will output the agreement decision.

The output information can also out the information about the fail processors and all processors decisions, which can be to check the result of the algorithm executed. For example, in Fig. 1, the failed processor is Pro_3, after executing the algorithm, all processors have the same decisions exclude Pro_3.

From the experiments which the size varying from 2 to 1000 for FloodSet algorithm and OptFloodSet algorithm. The results show that in all the experiments, all the no-faulty processors will have the same decisions after executing the algorithm, which proves that the simulation of FloodSet algorithm and OptFloodSet algorithm is correct.

III. EVALUATION

In this part, it will give an evaluation on the experiments. It will show the results in different test cases and analyze the results. The value of set X which processor can decide has two values, which are commit and abort.

The initial value is assigned randomly, and the default value is abort for each processor.

A. *Test*

First, we show that the simulator can work in the case of large size of network as below:

```
Time used: 27.757 s

Simulator results:
Number of Processors: 1000
Number of Decided Processors: 610
Number of Failed Processors: 390
Number of Rounds: 501
Number of Messages: 405848609
Satisfy Agreement: true
Agreement Decision: Abort
```

Fig. 2. Successful simulation in FloodSet of large size

```
Time used: 0.183 s

Simulator results:
Number of Processors: 1000
Number of Decided Processors: 541
Number of Failed Processors: 459
Number of Rounds: 501
Number of Messages: 1993543
Satisfy Agreement: true
Agreement Decision: Abort
```

Fig. 3. Successful simulation in OptFloodSet of large size

B. *FloodSet Algorithm*

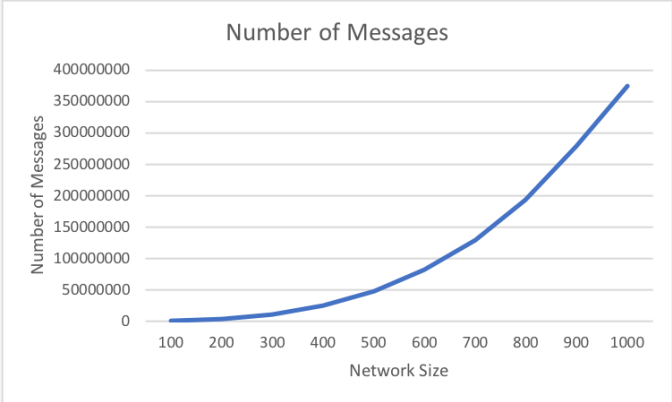

Fig. 4. The Number of Messages with Fixed Failures $\frac{n}{2}$

Finding 1: For the FloodSet algorithm, in the situation of fixed failures $\frac{n}{2}$, with the increase of network size, the number of messages increases more and more rapidly.

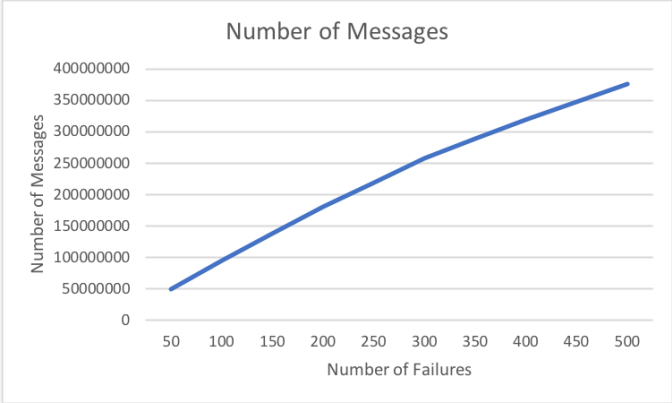

Fig. 5. The Number of Messages with Fixed Network Size

Finding 2: For the FloodSet algorithm, in the situation of fixed network size, with the increase of number of failures, the number of messages increases more and more slowly.

C. OptFloodSet Algorithm

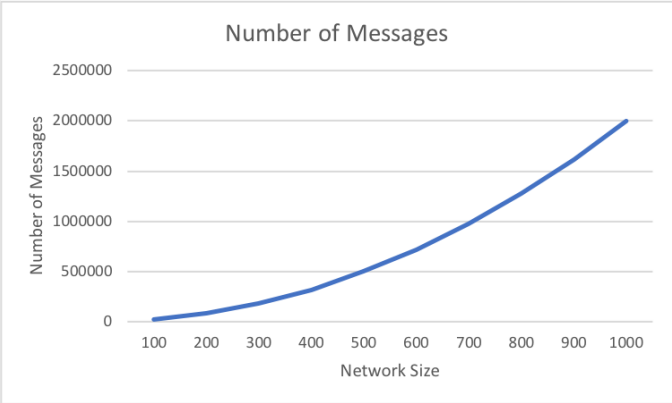

Fig. 6. The Number of Messages with Fixed Failures $\frac{n}{2}$

Finding 3: For the OptFloodSet algorithm, in the situation of fixed failures $\frac{n}{2}$, with the increase of network size, the number of messages increases more and more rapidly.

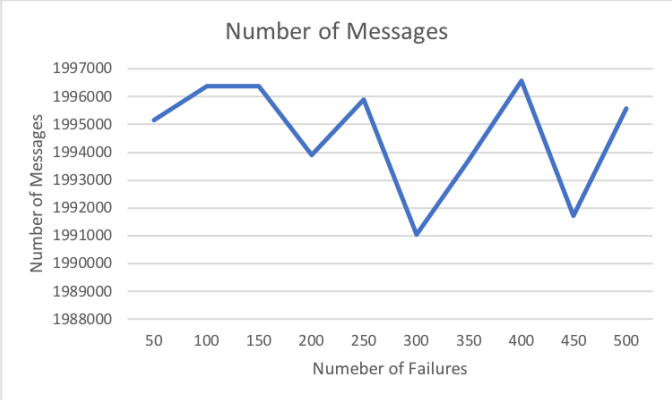

Fig. 7. The Number of Messages with Fixed Network Size

Finding 4: For the OptFloodSet algorithm, in the situation of fixed network size, with the increase of number of failures, the number of messages is fluctuant, which tends to be fixed value. It can be concluded the complexity is constant.

D. Comparison

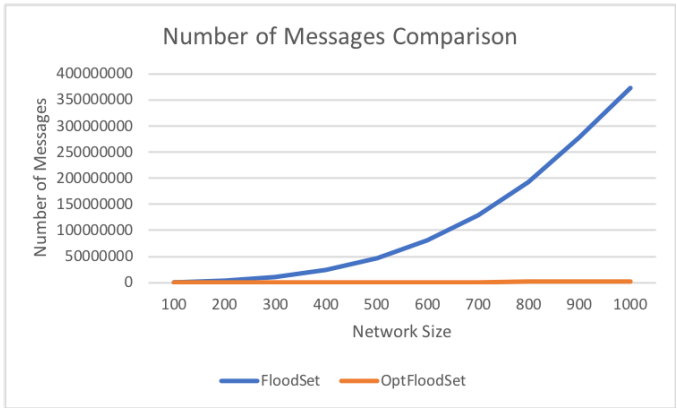

Fig. 8. The Number of Messages Comparison with Fixed Failures $\frac{n}{2}$

Finding 5: In the situation of fixed failures $\frac{n}{2}$, the number of messages in both algorthm are growing with the increase of network size. However, we can conclude that the number of messages in FloodSet is larger than OptFloodSet. Besides, the growth rate in FloodSet is larger than OptFloodSet, which means the complexity of OptFloodSet is less than FloodSet.

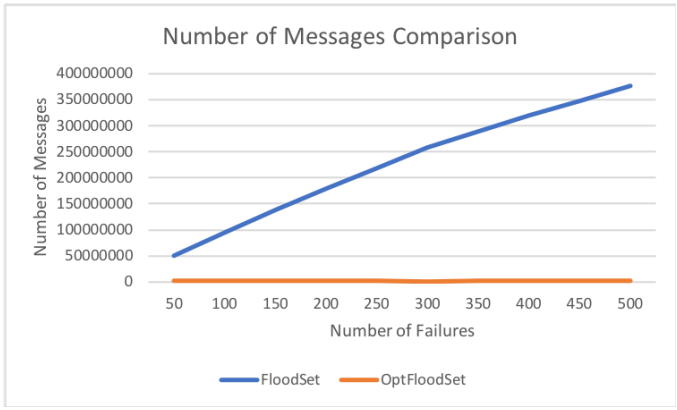

Fig. 9. The Number of Messages Comparison with Fixed Network Size.

Finding 6: In the situation of fixed network size, the number of messages in FloodSet algortihm is growing with the increase of number of failures. However, in OptFloodSet algorithm, the number of messages is stable.

It can also be concluded that the complexity of OptFloodSet is less than FloodSet.

IV. Conclusion

This report describes the simulator of FloodSet algorithm and OptFloodSet algorithm and show the results of experiments through the simulator based on different test data.

According to the results, both algorithm implemented in siumulator are correct and satisfy the agreement.

In the situation of fixed failures, the number of messages in both algorithms are growing with the increase of network size. However, the number of messages in FloodSet is larger than OptFloodSet in the situation of same network size, which means the complexity of OptFloodSet is less than FloodSet.

In the situation of fixed network size, the number of messages in FloodSet is growing with the increase of number of failures. However, in OptFloodSet, the number of messages locate in a stable value. Besides, we also concluded that the complexity of OptFloodSet is less than FloodSet.

In conclusion, OptFloodSet algorithm performs better than FloodSet in both case.