

# Report of Team Simulation Project

Kevin Cyu[2]

kevinbird61@gmail.com

***Nation Cheng Kung University***

Nuntanut Bhooanusas(ChuanJie Fu)

nuttt.nunutanut@gmail.com

***Nation Cheng Kung University***

Zhuo JingZhao(Taku)

taku0592@gmail.com

***Nation Cheng Kung University***

December 15, 2017

## **Abstract**

This document is generated by docoGen. We'll introduce our effort in this simulation project later.

## 0.1 Introduction

About our simulation project

- Course of network simulation in NCKU[1]
- For education only.

### 0.1.1 Why we using Docogen as document generator?

Standalone document generator

- Edit without insert configuration into your source code.
- And author is Kevin Cyu[2]

## 0.2 FlowChart

That's how our program work

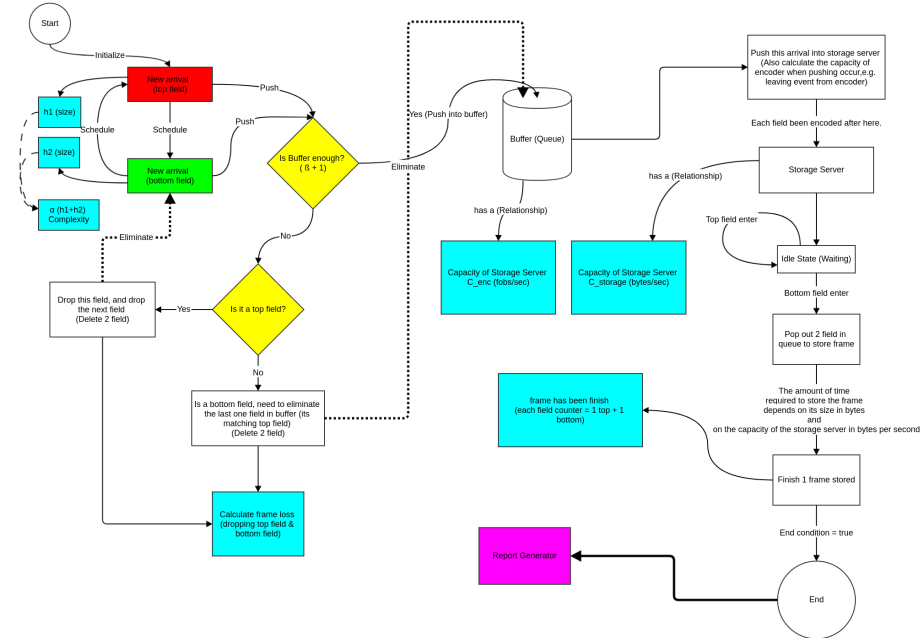


Figure 1: Entire system

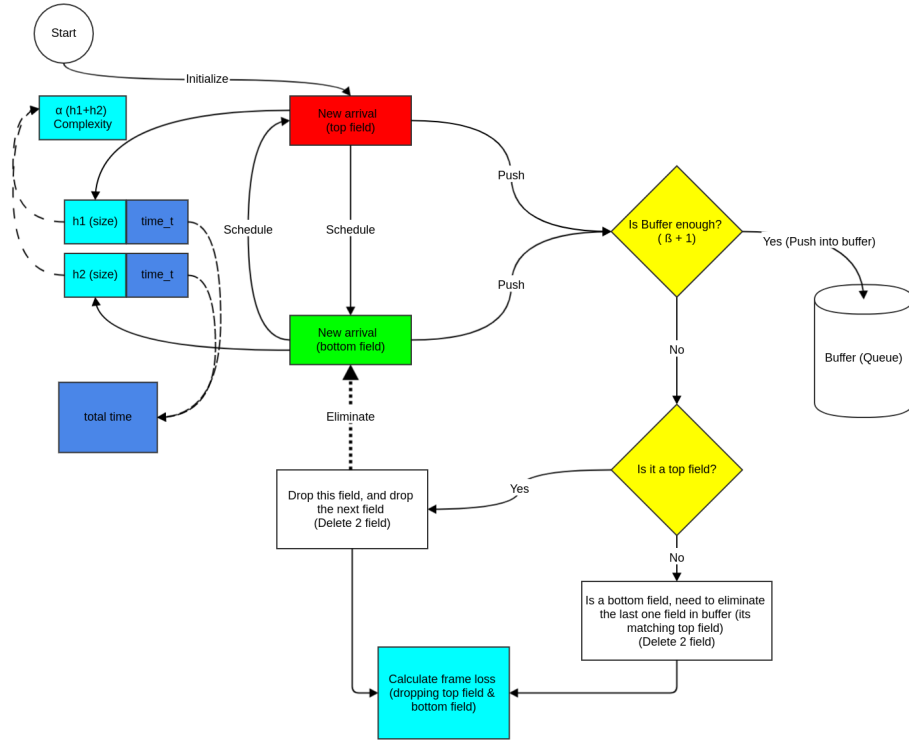


Figure 2: Initial routine + Encoder

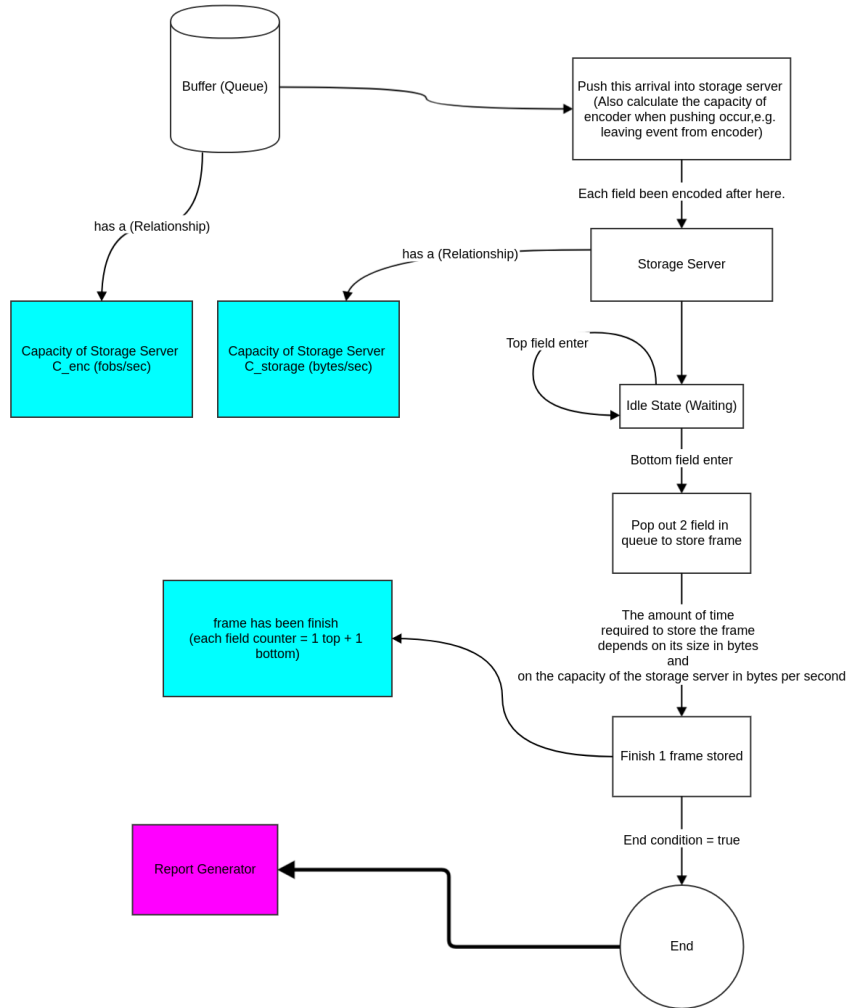


Figure 3: Encoder + Storage server

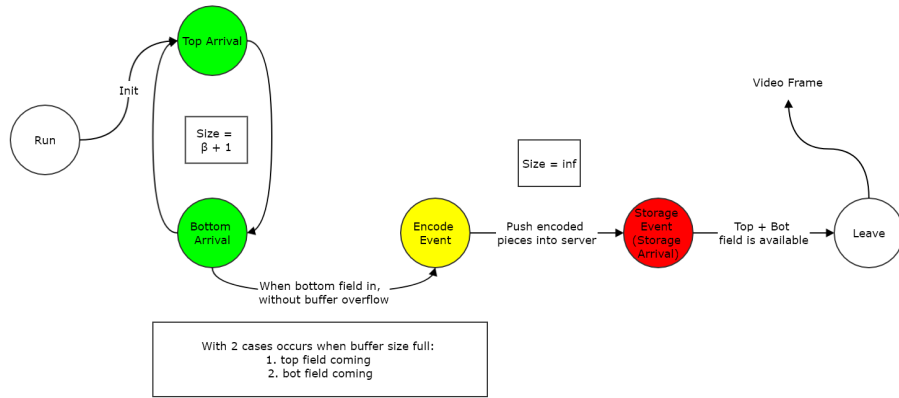


Figure 4: Simulation graph of project

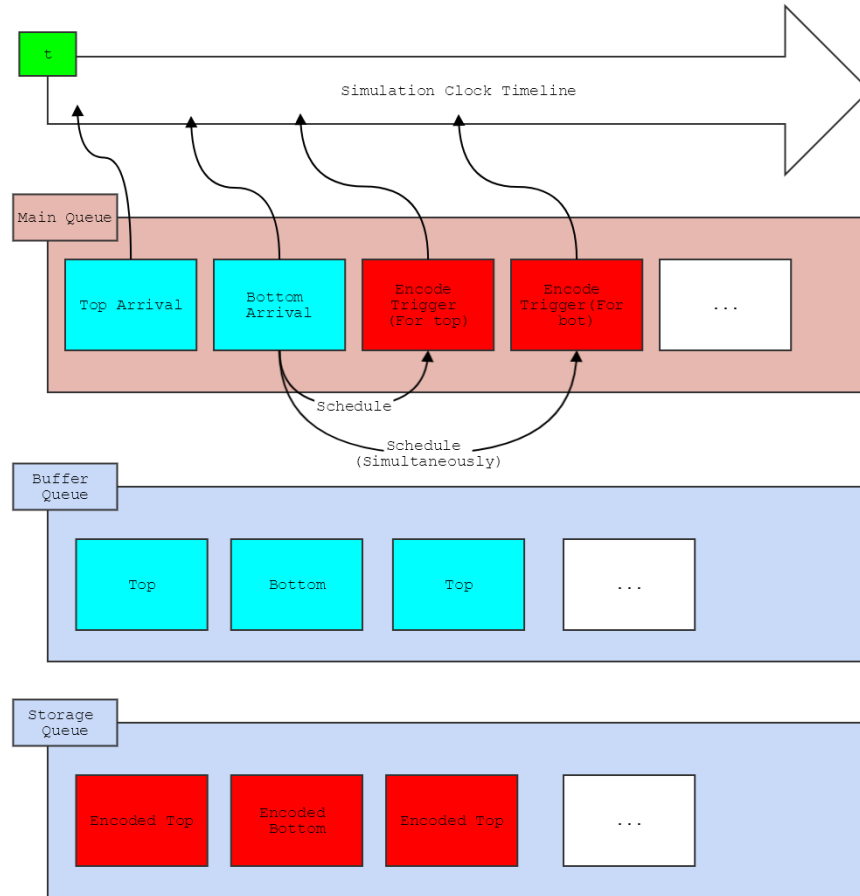


Figure 5: Memory usage(Queue)

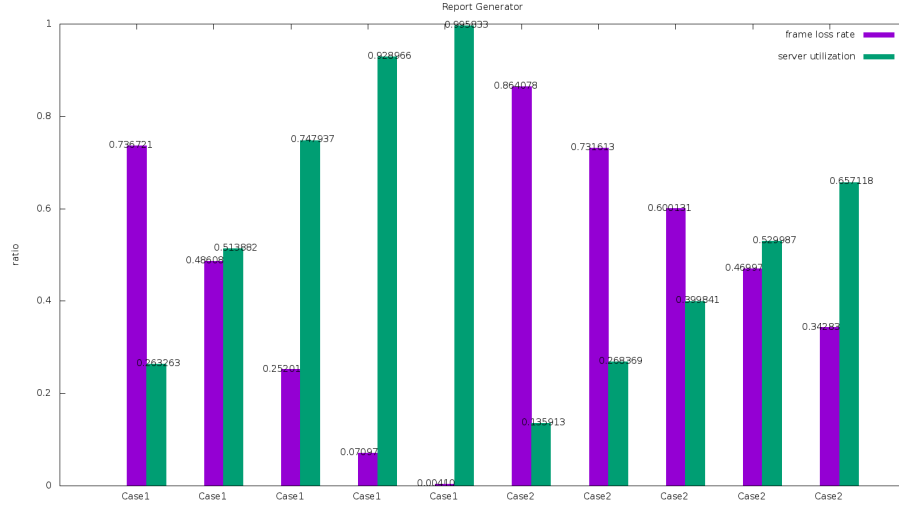


Figure 6: Entire Simulation System Report

### 0.3 Entire System

#### Explanation

- This simulation is based on the video frame encoding server to build its model.
- All the event can be pushed into main event queue, which the simulation clock will base on it and progress.
- And the other queue: buffer and storage server can directly use the top event and bottom event. These event have a special data structure, that can adapt size with each event together(Discuss Later).

#### Several Routine

1. Initial Routine(Arrival)
  - (a) Top field arrival
  - (b) Bot field arrival
2. Encoding Process
  - (a) Encoding Event(Top/Bot)



### 3. Storage Process

(a) Leave Event(Top+Bot)

## 0.4 Data Structure

Each Event

```
1 #ifndef STATE_VAR
2 #define STATE_VAR
3
4 // determine inter-arrival time
5 typedef float inter_t;
6
7 // frame fraction.
8 // 0: top, 1: bottom
9 typedef struct frame_piece {
10     int type;
11     float size;
12     inter_t timestamp;
13     // for link-list
14     struct frame_piece *parent;
15     struct frame_piece *child;
16 } frame_frac;
17
18 #endif
```

Listing 1: special event structure

Type of event:

- 0: Top arrival
- 1: Bot arrival
- 2: Encoded Top arrival
- 3: Encoded Bot arrival
- 4: Leaving arrival
- 99: [Deprecated] Dummy event( use for debug )

Sizing and timing problem

- At initial routine, the size will randomly generated by each arrival.

- And then when each field comes to encoder, encoding process will base on their size to schedule the finish time.
- When encoded bot field arrive in buffer, then it will represent a pair-wise field is save, and schedule a Leaving event.
- Also, this leaving event will base on the summation size of encoded top and bot fields.

# Bibliography

- [1] NCKU, Nation Cheng Kung University, Tainan, Taiwan
- [2] Kevin Cyu, <https://github.com/kevinbird61>