Lab 2

Multicast Programming

Compilation :

sh install_library.sh
make

Execution :

Unicast without FEC：

./unicast n send **<ip>** 6666 test_input.png
./unicast n recv **<ip>** 6666

Unicast using FEC：

./unicast f send **<ip>** 6666 test_input.png
./unicast f recv **<ip>** 6666

unicast 共有 5 個執行參數：
./unicast <n 或 f> <send 或 recv> <ip> <port> <檔名>
第 1 個代表要不要使用 FEC： n 為不使用 / f 為使用。
第 2 個代表 server 或 client：send 為 server / recv 為 client
第 3 個代表 IP address
第 4 個代表 port
第 5 個代表要傳送的檔名，當第 2 個參數為 send 時才需輸入

Multicast without FEC：

./server n **<ip>** test_input.png 15
./client n **<ip>**

Multicast using FEC：

./server f **<ip>** test_input.png 15
./client f **<ip>**

server 共有 4 個執行參數：
./unicast <n 或 f> <ip> <檔名> <秒數>
第 1 個代表要不要使用 FEC： n 為不使用 / f 為使用。

第 2 個代表 IP address
第 3 個代表要傳送的檔名
第 4 個代表要觸發播送事件的等待秒數

client 共有 2 個執行參數：
./unicast <n 或 f> <ip>
第 1 個代表要不要使用 FEC： **n** 為不使用 / **f** 為使用
第 2 個代表 IP address

說明：

FEC 的部分有使用 library，所以在 compile 之前須先執行 install_library.sh 安裝之後再 make。

multicast 的 server 執行參數中，可輸入 argv[4]，決定 server 開啟幾秒後要觸發播送事件，若沒有輸入的話是預設 15 秒。由於要求封包要加入序號，所以我在封包的最前面設定 11 位擺序號。又因為 buffer 一次是讀 1024 Bytes，但檔案的大小除以 1024 不可能剛好除盡，所以最後一個封包的大小可能會跟前面都不一樣，如果 client 端不知道有效資料長度，最後一次一樣寫 1024 Bytes 進去，就會造成檔案大小變得比原本大，多出一些空的資料，為了解決這個問題，所以我在封包序號後面又設定 5 位擺封包的有效資料長度。因此整個封包的大小是 11+5+1024=1040，然後用 library 的 fec_get_enc_msg_length()得知 encode 後會變多長，並把長度先傳給 client，讓他知道要開多大的 buffer 去接。把序號和有效資料長度都擺好後，再把從檔案讀到的 1024 Bytes 擺進去，然後整段一起用 fec_encode()編碼之後再傳給 client。client 收到之後先用 fec_decode()解碼，然後把最前面擺的序號和有效資料長度都讀出來，利用序號算這一次和上一次之間掉了多少封包，並決定這次要寫多少 Bytes 進檔案。Server 端傳完所有封包之後，會再傳最後一個封包的序號以及檔案的總大小給 client，讓 client 端能夠計算封包遺失率及資料遺失率。

實驗結果與觀察心得：

【unicast 傳送 1.3MB 的檔案】
無 FEC ↓

```
os2018@os2018:~/Network Homework/Multicast-Programming$ ./unicast n recv 10.0.2.15 6666
The multimedia is saved in ./Received/image.jpg
Lost packet = 213
Packet lost rate = 0.167453
Received data size = 1084133
Lost data size = 218112
Data lost rate = 0.167489
```

一樣無 FEC，但是 server 端多加了一個 10000 次的 for 迴圈 delay ↓

```
os2018@os2018:~/Network Homework/Multicast-Programming$ ./unicast n recv 10.0.2.15 6666
The multimedia is saved in ./Received/image.jpg
Lost packet = 0
Packet lost rate = 0.000000
Received data size = 1302245
Lost data size = 0
Data lost rate = 0.000000
```

有 FEC ↓

```
os2018@os2018:~/Network Homework/Multicast-Programming$ ./unicast f recv 10.0.2.15 6666
The multimedia is saved in ./Received/image.jpg
Lost packet = 40
Packet lost rate = 0.031447
Received data size = 1261285
Lost data size = 40960
Data lost rate = 0.031453
```

一樣有 FEC，但是 server 端多加了一個 10000 次的 for 迴圈 delay ↓

```
os2018@os2018:~/Network Homework/Multicast-Programming$ ./unicast f recv 10.0.2.15 6666
The multimedia is saved in ./Received/image.jpg
Lost packet = 0
Packet lost rate = 0.000000
Received data size = 1302245
Lost data size = 0
Data lost rate = 0.000000
```

一開始用 unicast 傳的時候發現封包掉的很嚴重，但嘗試在 server 傳下一個封包前多加一個 10000 次的 for 迴圈延遲下一次傳的時間後，封包就完全沒有掉了。而沒有加 for 迴圈時，FEC 的遺失率會比無 FEC 時的遺失率少，猜想可能是因為 FEC 時 server 端多做了 encode 的動作，所以會耗費比較多時間，傳送封包的頻率就沒有那麼快，client 比較不會來不及中間來不及接到就直接跳到下一個。

【unicast 傳送 8.2MB 的檔案】

無 FEC ↓

```
os2018@os2018:~/Network Homework/Multicast-Programming$ ./unicast n recv 10.0.2.15 6666
The multimedia is saved in ./Received/test_input.JPG
Lost packet = 314
Packet lost rate = 0.039118
Received data size = 7897755
Lost data size = 321536
Data lost rate = 0.039120
```

無 FEC 但加 for 迴圈 delay ↓

```
os2018@os2018:~/Network Homework/Multicast-Programming$ ./unicast n recv 10.0.2.15 6666
The multimedia is saved in ./Received/test_input.JPG
Lost packet = 0
Packet lost rate = 0.000000
Received data size = 8219291
Lost data size = 0
Data lost rate = 0.000000
```

有 FEC ↓

```
os2018@os2018:~/Network Homework/Multicast-Programming$ ./unicast f recv 10.0.2.15 6666
The multimedia is saved in ./Received/test_input.JPG
Lost packet = 159
Packet lost rate = 0.019808
Received data size = 8056475
Lost data size = 162816
Data lost rate = 0.019809
```

有 FEC，但加 for 迴圈 delay↓

```
os2018@os2018:~/Network Homework/Multicast-Programming$ ./unicast f recv 10.0.2.15 6666
The multimedia is saved in ./Received/test_input.JPG
Lost packet = 0
Packet lost rate = 0.000000
Received data size = 8219291
Lost data size = 0
Data lost rate = 0.000000
```

傳送更大的 8.2MB 檔案情況也和 1.3MB 時差不多，有 FEC 時遺失率較無 FEC 時小，加了 for 迴圈之後兩者都不會遺失。

【unicast 傳送 113.4MB 的檔案】

無 FEC↓

```
os2018@os2018:~/Network Homework/Multicast-Programming$ ./unicast n recv 10.0.2.15 6666
The multimedia is saved in ./Received/video2.mp4
Lost packet = 845
Packet lost rate = 0.007630
Received data size = 112533660
Lost data size = 865280
Data lost rate = 0.007630
```

無 FEC 但加 for 迴圈 delay↓

```
os2018@os2018:~/Network Homework/Multicast-Programming$ ./unicast n recv 10.0.2.15 6666
The multimedia is saved in ./Received/video2.mp4
Lost packet = 0
Packet lost rate = 0.000000
Received data size = 113398940
Lost data size = 0
Data lost rate = 0.000000
```

有 FEC↓

```
os2018@os2018:~/Network Homework/Multicast-Programming$ ./unicast f recv 10.0.2.15 6666
The multimedia is saved in ./Received/video2.mp4
Lost packet = 471
Packet lost rate = 0.004253
Received data size = 112916636
Lost data size = 482304
Data lost rate = 0.004253
```

有 FEC 但加 for 迴圈 delay↓

```
os2018@os2018:~/Network Homework/Multicast-Programming$ ./unicast f recv 10.0.2.15 6666
The multimedia is saved in ./Received/video2.mp4
Lost packet = 27
Packet lost rate = 0.000244
Received data size = 113371292
Lost data size = 27648
Data lost rate = 0.000244
```

再傳送更大的 113.4MB 檔案，沒加 for 迴圈前，有 FEC 時遺失率還是較無 FEC 時小，但加了 for 迴圈之後無 FEC 的情況沒掉封包，但有 FEC 時卻掉了，但是我覺得能是網路狀態影響的，因為之後再測每次結果都不太一樣，有時候沒加 FEC 的也會掉封包，加 FEC 的有時候也不會掉。

【multicast 傳送 1.3MB 的檔案給 3 個 client】

無 FEC：

client_1 ↓

```
os2018@os2018:~/Network Homework/Multicast-Programming$ ./client n 10.0.2.15
Opening datagram socket....OK.
Setting SO_REUSEADDR...OK.
Binding datagram socket...OK.
Adding multicast group...OK.
The multimedia is saved in ./Received/image.jpg
Lost packet = 0
Packet lost rate= 0.000000
Received data size = 1302245
Lost data size = 0
Data lost rate = 0.000000
```

client_2 ↓

```
os2018@os2018:~/Network Homework/Multicast-Programming/client_2$ ./client2 n 10.
0.2.15
Opening datagram socket....OK.
Setting SO_REUSEADDR...OK.
Binding datagram socket...OK.
Adding multicast group...OK.
The multimedia is saved in ./Received/image.jpg
Lost packet = 0
Packet lost rate= 0.000000
Received data size = 1302245
Lost data size = 0
Data lost rate = 0.000000
```

client_3 ↓

```
os2018@os2018:~/Network Homework/Multicast-Programming/client_3$ ./client3 n 10.
0.2.15
Opening datagram socket....OK.
Setting SO_REUSEADDR...OK.
Binding datagram socket...OK.
Adding multicast group...OK.
The multimedia is saved in ./Received/image.jpg
Lost packet = 0
Packet lost rate= 0.000000
Received data size = 1302245
Lost data size = 0
Data lost rate = 0.000000
```

有 FEC：

client_1 ↓

```
os2018@os2018:~/Network Homework/Multicast-Programming$ ./client f 10.0.2.15
Opening datagram socket....OK.
Setting SO_REUSEADDR...OK.
Binding datagram socket...OK.
Adding multicast group...OK.
The multimedia is saved in ./Received/image.jpg
Lost packet = 0
Packet lost rate= 0.000000
Received data size = 1302245
Lost data size = 0
Data lost rate = 0.000000
```

client_2 ↓

```
os2018@os2018:~/Network Homework/Multicast-Programming/client_2$ ./client2 f 10.
0.2.15
Opening datagram socket....OK.
Setting SO_REUSEADDR...OK.
Binding datagram socket...OK.
Adding multicast group...OK.
The multimedia is saved in ./Received/image.jpg
Lost packet = 0
Packet lost rate= 0.000000
Received data size = 1302245
Lost data size = 0
Data lost rate = 0.000000
```

client_3 ↓

```
os2018@os2018:~/Network Homework/Multicast-Programming/client_3$ ./client3 f 10.
0.2.15
Opening datagram socket....OK.
Setting SO_REUSEADDR...OK.
Binding datagram socket...OK.
Adding multicast group...OK.
The multimedia is saved in ./Received/image.jpg
Lost packet = 0
Packet lost rate= 0.000000
Received data size = 1302245
Lost data size = 0
Data lost rate = 0.000000
```

傳送 1.3MB 的小檔案時，FEC 前後沒有什麼差別，每個 client 都不會掉封包

【multicast 傳送 8.2MB 的檔案】

無 FEC：

client_1 ↓

```
os2018@os2018:~/Network Homework/Multicast-Programming$ ./client n 10.0.2.15
Opening datagram socket....OK.
Setting SO_REUSEADDR...OK.
Binding datagram socket...OK.
Adding multicast group...OK.
The multimedia is saved in ./Received/test_input.JPG
Lost packet = 0
Packet lost rate= 0.000000
Received data size = 8219291
Lost data size = 0
Data lost rate = 0.000000
```

client_2 ↓

```
os2018@os2018:~/Network Homework/Multicast-Programming/client_2$ ./client2 n 10.
0.2.15
Opening datagram socket....OK.
Setting SO_REUSEADDR...OK.
Binding datagram socket...OK.
Adding multicast group...OK.
The multimedia is saved in ./Received/test_input.JPG
Lost packet = 0
Packet lost rate= 0.000000
Received data size = 8219291
Lost data size = 0
Data lost rate = 0.000000
```

client_3 ↓

```
os2018@os2018:~/Network Homework/Multicast-Programming/client_3$ ./client3 n 10.
0.2.15
Opening datagram socket....OK.
Setting SO_REUSEADDR...OK.
Binding datagram socket...OK.
Adding multicast group...OK.
The multimedia is saved in ./Received/test_input.JPG
Lost packet = 0
Packet lost rate= 0.000000
Received data size = 8219291
Lost data size = 0
Data lost rate = 0.000000
```

有 FEC：

client_1 ↓

```
os2018@os2018:~/Network Homework/Multicast-Programming$ ./client f 10.0.2.15
Opening datagram socket....OK.
Setting SO_REUSEADDR...OK.
Binding datagram socket...OK.
Adding multicast group...OK.
The multimedia is saved in ./Received/test_input.JPG
Lost packet = 3
Packet lost rate= 0.000374
Received data size = 8216219
Lost data size = 3072
Data lost rate = 0.000374
```

client_2 ↓

```
os2018@os2018:~/Network Homework/Multicast-Programming/client_2$ ./client2 f 10.
0.2.15
Opening datagram socket....OK.
Setting SO_REUSEADDR...OK.
Binding datagram socket...OK.
Adding multicast group...OK.
The multimedia is saved in ./Received/test_input.JPG
Lost packet = 0
Packet lost rate= 0.000000
Received data size = 8219291
Lost data size = 0
Data lost rate = 0.000000
```

client_3 ↓

```
os2018@os2018:~/Network Homework/Multicast-Programming/client_3$ ./client3 f 10.
0.2.15
Opening datagram socket....OK.
Setting SO_REUSEADDR...OK.
Binding datagram socket...OK.
Adding multicast group...OK.
The multimedia is saved in ./Received/test_input.JPG
Lost packet = 0
Packet lost rate= 0.000000
Received data size = 8219291
Lost data size = 0
Data lost rate = 0.000000
```

檔案稍微變大後，無 FEC 時 3 個 client 都沒有掉封包，但 FEC 時，clien_1 掉了 3 個封包，其他則沒掉。

【multicast 傳送 113.4MB 的檔案】

無 FEC：

client_1 ↓

```
os2018@os2018:~/Network Homework/Multicast-Programming$ ./client n 10.0.2.15
Opening datagram socket....OK.
Setting SO_REUSEADDR...OK.
Binding datagram socket...OK.
Adding multicast group...OK.
The multimedia is saved in ./Received/video2.mp4
Lost packet = 567
Packet lost rate= 0.005120
Received data size = 112818332
Lost data size = 580608
Data lost rate = 0.005120
```

client_2 ↓

```
os2018@os2018:~/Network Homework/Multicast-Programming/client_2$ ./client2 n 10.
0.2.15
Opening datagram socket....OK.
Setting SO_REUSEADDR...OK.
Binding datagram socket...OK.
Adding multicast group...OK.
The multimedia is saved in ./Received/video2.mp4
Lost packet = 207
Packet lost rate= 0.001869
Received data size = 113186972
Lost data size = 211968
Data lost rate = 0.001869
```

client_3 ↓

```
os2018@os2018:~/Network Homework/Multicast-Programming/client_3$ ./client3 n 10.
0.2.15
Opening datagram socket....OK.
Setting SO_REUSEADDR...OK.
Binding datagram socket...OK.
Adding multicast group...OK.
The multimedia is saved in ./Received/video2.mp4
Lost packet = 741
Packet lost rate= 0.006691
Received data size = 112640156
Lost data size = 758784
Data lost rate = 0.006691
```

有 FEC：

client_1 ↓

```
os2018@os2018:~/Network Homework/Multicast-Programming$ ./client f 10.0.2.15
Opening datagram socket....OK.
Setting SO_REUSEADDR...OK.
Binding datagram socket...OK.
Adding multicast group...OK.
The multimedia is saved in ./Received/video2.mp4
Lost packet = 648
Packet lost rate= 0.005851
Received data size = 112735388
Lost data size = 663552
Data lost rate = 0.005851
```

client_2 ↓

```
os2018@os2018:~/Network Homework/Multicast-Programming/client_2$ ./client2 f 10.
0.2.15
Opening datagram socket....OK.
Setting SO_REUSEADDR...OK.
Binding datagram socket...OK.
Adding multicast group...OK.
The multimedia is saved in ./Received/video2.mp4
Lost packet = 625
Packet lost rate= 0.005644
Received data size = 112758940
Lost data size = 640000
Data lost rate = 0.005644
```

client_3 ↓

```
os2018@os2018:~/Network Homework/Multicast-Programming/client_3$ ./client3 f 10.
0.2.15
Opening datagram socket....OK.
Setting SO_REUSEADDR...OK.
Binding datagram socket...OK.
Adding multicast group...OK.
The multimedia is saved in ./Received/video2.mp4
Lost packet = 876
Packet lost rate= 0.007910
Received data size = 112501916
Lost data size = 897024
Data lost rate = 0.007910
```

再傳了非常大的 113.4MB 檔案，FEC 前後 3 個 client 都會掉封包，我算了平均遺失率，FEC 前是 0.00456，FEC 後則是 0.00647，有 FEC 的封包遺失率高於沒有 FEC 時

經過多次測試和觀察後，發現以 multicast 來說，檔案越小，就越不容易遺失，基本上 10MB 以內的檔案掉封包的機率非常小，很少遇到。但是檔案如果到 100 多 MB 的時候就一定會掉封包，檔案越大遺失率越高。但是如果以 unicast 沒有加 for 迴圈的時候來看，卻是檔案越大遺失率越小。 而加了 for 迴圈延遲傳送的頻率之後可以大大減少封包遺失率，但相對時間就得花比較長。另外還發現一樣用 multicast 傳 8.2MB 的檔案，只開一個 client 時完全沒掉過封包，但是開了 3 個 client 時，3 個之中就會有一個掉封包。而 FEC 前後測試出來的結果很不一定，有時候 FEC 前會比較高，有時候又變 FEC 後比較高，猜想應該是網路狀態影響的。此外，還有一個現象是在傳檔案時，因為要等他傳完有時候會邊開 youtube 影片看，發現如果邊看影片邊傳的話遺失率也會提高，本來平常都不會遺失的狀況，邊開著影片時就會掉封包。