

網路系統總整與實作 Lab #6

Lab6_Handoff Simulation over ns3

0716236 劉耀文

Part I – Visualize Result

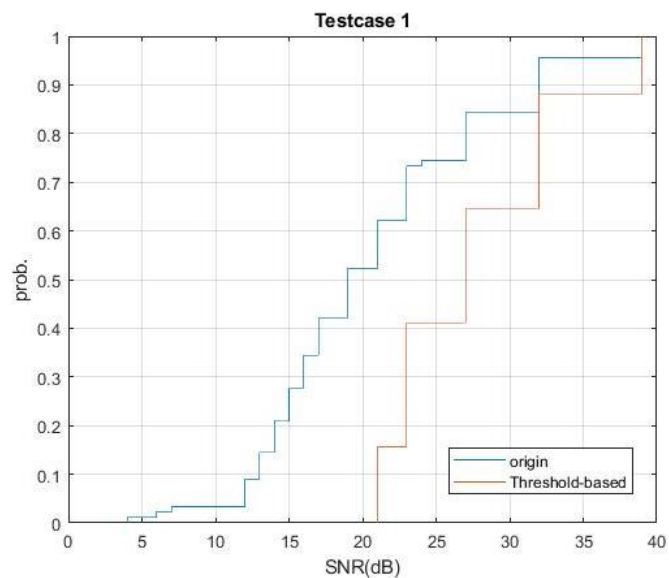


Figure 1. Data packet SNR of testcase 1

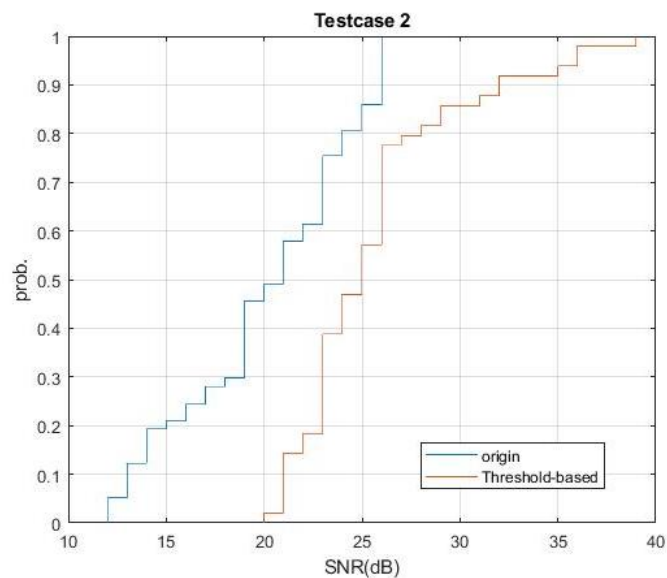


Figure 2. Data packet SNR of testcase 2

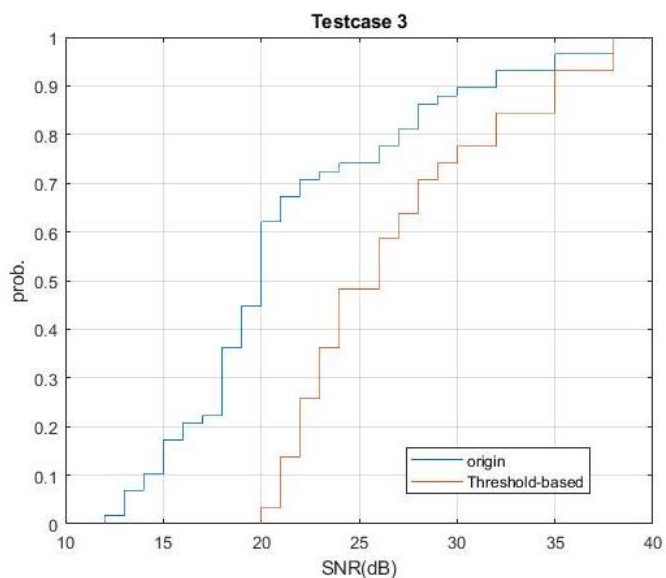


Figure 3. Data packet SNR of testcase 3

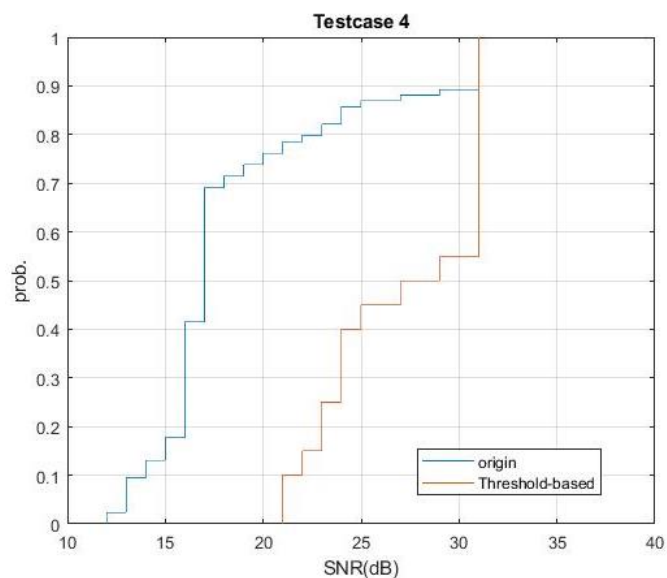


Figure 4. Data packet SNR of testcase 4

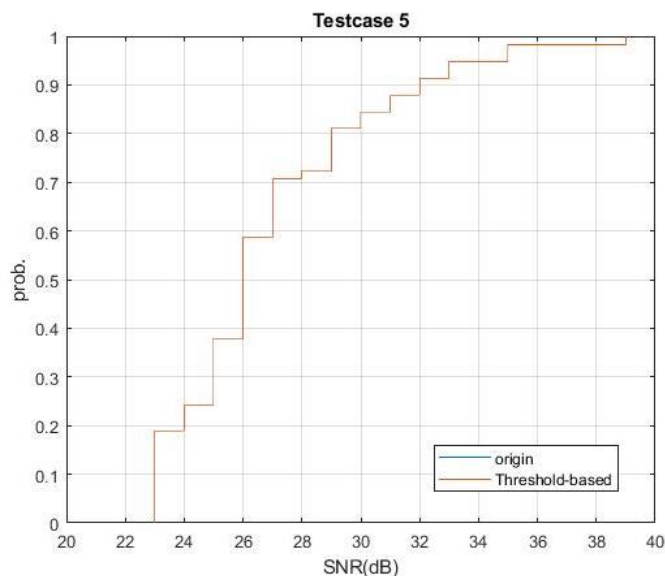


Figure 5. Data packet SNR of testcase 5

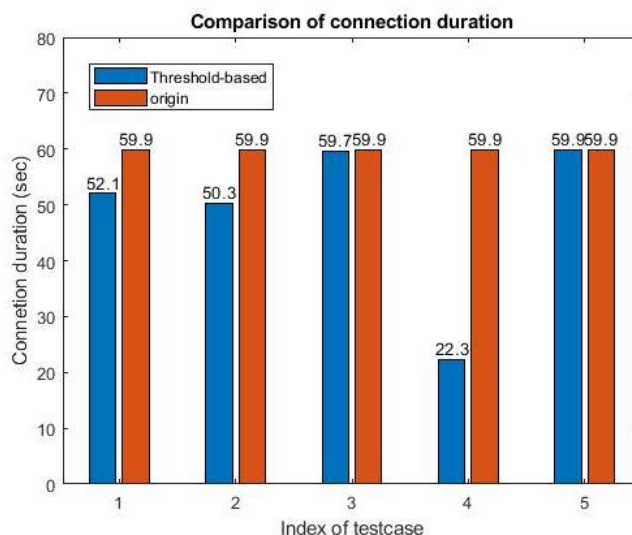


Figure 6. Comparison of connection duration

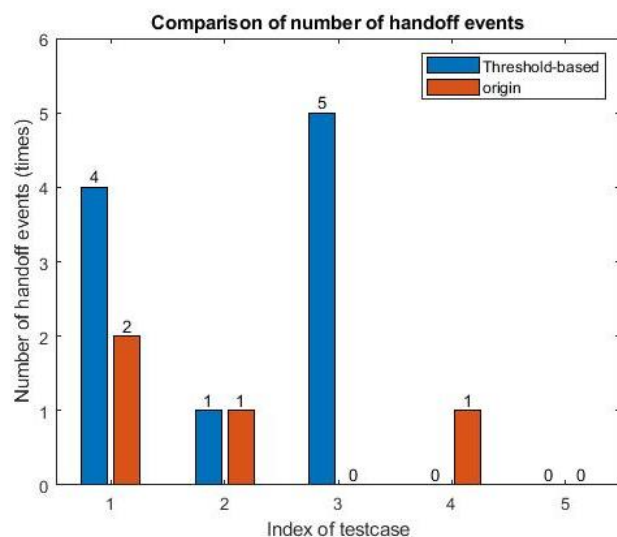


Figure 7. Comparison of number of handoff events

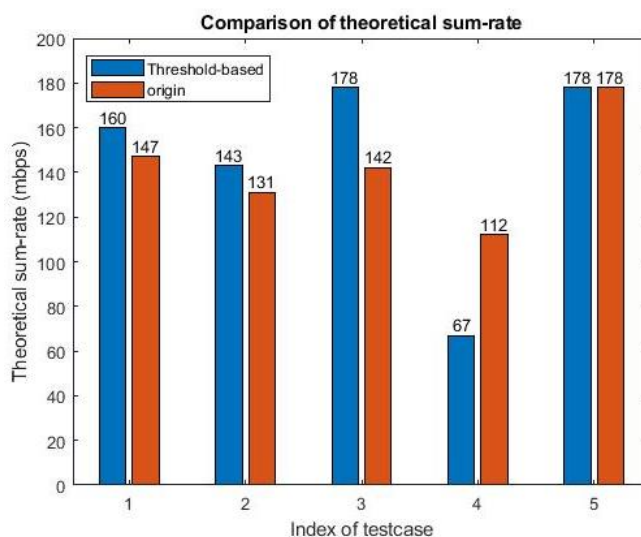


Figure 8. Comparison of theoretical sum-rate

Observation

1. 從 data packet SNR 的角度來看，在大多數歷經換手的情況下，會因為 threshold-based 演算法將 SNR 較差的連線斷線的緣故，所以 SNR 會維持在所設定的 threshold (20dB) 以上，整體的機率分布也會較原本的演算法要來的更右邊一點(SNR 較高)。
2. 因為 threshold-based 會主動將 beacon message 之 SNR 不佳的連線主動中斷，並且尋求 SNR 高於 threshold 的較佳品質連線，但由於本次實驗採用 active scanning，也就是說會依照 probing response 的 SNR 來當作建立連線與否的判斷，但在斷線前 client 並不會發出 probing request，所以不會知道從當下的 AP 斷線後會不會有符合條件的 AP 能夠重新連上，故整體的連線時長會比原先要來的低，在某些情況下更是不到原先的一半，也因為為了要維持連線品質，所以通常 threshold-based 會歷經較多次的換手。
3. 在絕大多數時候因為維持了連線品質，故 threshold-based 之 theoretical sum-rate 會比原先要好上一些。但有時候從 AP 斷線後不見得會重新連上另一個 AP(如 testcase_4)，或是中間斷線的時間太長，故連線時長會比原先要來的低，也因此導致了 theoretical sum-rate 大幅下降。
4. 另外 testcase_5 因為 SNR 都保持在 threshold 之上，故兩者的結果是一模一樣的。

Part II – Questions

1. Compare the advantages and disadvantages of the two handoff algorithms.

→ origin algorithm

- 優點: 換手次數較少、斷線時間較短
- 缺點: 在較佳連線品質的 AP 範圍中不會及早切換導致 SNR 以及 sum-rate 較差

→ Threshold-based algorithm

- 優點: 連線品質較佳、SNR 較佳、傳輸速率(theoretical sum-rate)較佳
- 缺點: 某些情況下會因為 AP 的連線品質不到門檻而無法連上線、可能會因為換手導致 packet loss

2. In task2, how do you make a client walk continuously?

(Hint: beacon interval is the smallest period of observation in the simulation)

- ➔ 在 task1 當中 client 會每一秒才移動一次，故 beacon message 的 SNR 變化會較為強烈，爲了讓 client 看起來移動得更爲“連續”，故原本 simulation 的 delay 從 1 秒改爲 0.125 秒，也就是說每一步切成 8 小步移動，這樣可以讓 SNR 的變化較為平緩，以達到讓 client 連續移動的效果。
- ➔ 另外會選擇 0.125 秒作為 simulation 的 delay 是因為 beacon interval 爲 0.1024 秒，也就是模擬中的最短觀察週期，若將 delay 設定得更低會沒有意義，且將每一步切成 8 小步的話，每一步的時長將會是 0.125 秒，讓整體的程式更有可讀性，也較不會有 rounding error。

3. In task3, how do you know the occurrence of handoff event? And how do you get connection duration?

(Which types of management packets did you use?)

- ➔ 在每經歷過一次 disassociation 後，client 重新獲得一次 association response 並且此封包是來自不同於上次斷線的 AP，這樣就構成了一次 handoff，實作方面是透過一個 boolean value 紀錄 client 與任何 AP 連線與否，並用一個 string 紀錄上次連線 AP 之 BSSID，如此一來便能夠判斷 handoff event 的發生與否。
- ➔ parser 可以從 association response 得知 client 和 AP 順利建立連線，以及透過 disassociation message 得知 client 從 AP 斷線。而 connection duration 則是利用 disassociation message 的 timestamp 減掉 association response 的 timestamp 來獲得，選擇 association response 而非 request 是因為以 client 的角度而言要接收到 response 才是建立連線。
- ➔ 然而在原本的演算法當中 client 與 AP 斷線並不會傳送 disassociation message，且只能透過收不到來自連線 AP 的 beacon message 當作斷線，故這邊選擇以斷線後才會出現的 probing request 當作是斷線的依據，故 connection duration 會利用斷線後的第一個 probing request 之 timestamp 減掉 association response 的 timestamp 來獲得。
- ➔ 因爲在完成 parser 的時候是透過 threshold-based algorithm 所產生之 pcap 檔案做爲測試，故一開始就是使用 disassociation message 作為斷線之依據，後面才爲了原本的演算法另外寫了第二個方法，這部分有在 code 裡面寫註解。

Part III – Course Feedback

以下問題皆以下半學期課程回答

- How difficult about the lectures do you feel about this class (1-10)?
→ 6，有些部分如果沒有基礎知識會比較聽不懂，但我認為比上半學期好理解很多。
- How difficult about the labs do you feel about this class (1-10)?
→ 5，不包含最後的 ns3 simulation 的話，基本上有上課不用花到太多時間做作業，可能一個晚上就可以繳交了。
→ 若包含 lab6 的話是 7，這個 lab 可能需要花一點時間看 source code 以及定義好的 class 來想要怎麼 implement 會比較好，以及斷線後要如何觸發 active scanning，但想通後其實不會太難。
- What is the most interesting part you have learned from the class?
→ Autorate，主要是因為自己的專題跟這個單元有一點相關，所以比較認真聽課也比較希望可以聽多一點。
→ mmWave，新穎的東西之前也只有聽過名詞沒有深入了解過。
- What is the most boring part you have learned from the class?
→ OFDM，雖然很重要但是真的太多數學公式了，也有可能是因為自己沒有很喜歡算數學所以才有這種想法，且如果沒有先修過訊號處理或是相關課程可能比較難理解。
- What is the most challenging part you have experienced in the labs?
→ Lab6 要花上不少時間來看 source code 以及 document 才能夠完成，在完成過程中花最多時間的就是理解 ns3 的 source code 在做什麼，例如連線以及斷線的流程，還有各種 class 要如何使用。
→ 其他次作業的話就是 Matlab 的程式編寫，雖然要寫的地方不多，但只有去年的網路通訊原理有稍微用過，所以一開始寫作業有一點點吃力，不過一下就可以克服了，有聽課的話 lab 本身不會太難。
- What is the most useful skill you learned from the lab?
→ 呈上題，從一堆 source code 的檔案中找出我們需要拿來參考的部分，不過因為 ns3 網站上的 code 可以透過點選就找到對應 function 或是 class 是在哪個檔案被定義以及實作的，所以省去了不少麻煩。
- Any modifications you expect to see in the future?
→ 課程主要以學理為主，且作業都是用軟體模擬，如果有機會碰到實體的機器會很讚，但今年因為疫情沒辦法就非常可惜。

補充心得：我的專題跟 Wi-Fi 有極大的關聯性，所以在選課的時候就對這門課非常有興趣，修完了一學期後也確實對我專題所需的專業知識有更進一步的了解，只可惜少了最後 OpenWrt 的部分。