

Performance Modeling Final Project Report

組員名單：

R00922020 沈昱辰

R00922067 彭正龍

R00922132 何桓毅

Problem definition：

在 unstructured peer-to-peer 的環境下，有 N 個 user， O 個檔案大小不同的歌曲，每個 user 的頻寬大小、興趣以及上線時間都有所不同，我們現在希望能夠讓每個人從 request 到載完單位大小檔案的時間越小越好。

程式流程：

- (1) 產生 100 個 user，初始每個 user 對於每個 type 的歌都隨機擁有一首。上線的時間以 uniform distribution 隨機指定，在 peak hour 的時候有比較高的上線機率。並依照 small world 的方法，將這些 user 連接起來。
- (2) 產生一個 request，request 資訊包含 User A 在時間 i 要求一首歌 m 。紀錄從 request 到成功取得這首歌所花的時間、找不到檔案的次數。
- (3) 如果成功取得一首歌，則會選擇一些 user 來備份這首歌。
- (4) 重複步驟 (2) 和 (3) 一定次數後，計算總共的 request 時間以及找不到檔案的機率。

Request:

當 user A 請求了一首歌後，會去找到在這個時間點，這個網路中擁有這首歌的那些 user，然後把能夠從這些 user 拿到的頻寬給加總起來，除以檔案的大小即為 request 這首歌所需要花的下載時間。

Request 的時候會有三種情形：一、能夠 request 到這首歌，回傳的即為所需時間。二、這首歌 user A 已經擁有，回傳 0 秒。三、找不到這首歌，回傳一個 penalty 並紀錄下來。Penalty 決定的方法則是在所有 request 成功所需要的下載時間中取最大值。

Replication:

當 user A 在時間 i 請求了一首歌後，如果成功的拿到這首歌，就會選出兩個 user 並把這首歌曲複製給這兩個 user。這兩個 user 的選法有兩種：

(1) 一個考慮 user 的頻寬 + 一個考慮 user 的喜好：

User A 會從在時間 i 有上線的其他 user 中，把每個 user 所能提供的頻寬依據不同時段所分配到的權重作加總(例如 peak hour 加權兩倍)，選出最大值的那個 user，這個人就是可以提供最多頻寬的人。

另一個 user 的選法，是除去上面選出的 user 後，在剩下的 user 裡面選出和自己興趣最相近的那個 user 當作複製的對象。

(2) 兩個考慮 user 的喜好：

User A 會從在時間 i 有上線的其他 user 中，選出和自己興趣最相近的兩個 user，並且把檔案複製給這兩個人。興趣的相似度是以 user 擁有的每種類型歌曲的比例當作向量，再作內積。內積值越大就代表兩者的興趣越相近。

因為每個 user 有 buffer size 的限制，所以從上面被選中的 user 如果 buffer size 已經滿了，就會使用 LRU 的方法，把最久以來拿到的那首音樂刪除，直到 buffer 可以放入這首要複製的歌曲為止。

Simulation 參數設定：

N	100
O	20
Time slot	24/day
Peak hour	8~12pm
Peak hour 的 request 量是 Off-peak 的兩倍	
File size	5~10MB, uniform distribution
Buffer size	50MB
Number of neighbor	3~5
Song Type	3

Simulation result：兩種機率各跑 100 次做平均，每次跑 7 天

(1) Off-peak hour online probability = 0.25

Peak hour online probability = 0.5

考慮頻寬與興趣		只考慮興趣	
找到的時間	找不到的機率	找到的時間	找不到的機率
2647.636	0.26316684	2682.9544	0.26697601

總共 Request 數量：1457

(2) Off-peak hour online probability = 0.1

Peak hour online probability = 0.2

考慮頻寬與興趣		只考慮興趣	
找到的時間	找不到的機率	找到的時間	找不到的機率
3267.686067	0.680296146	3342.290112	0.696035404

總共 Request 數量：556