

Principles of Communications Networks

Homework assignment 2

Apply language: C++

0310120 電工 07 陳家煒

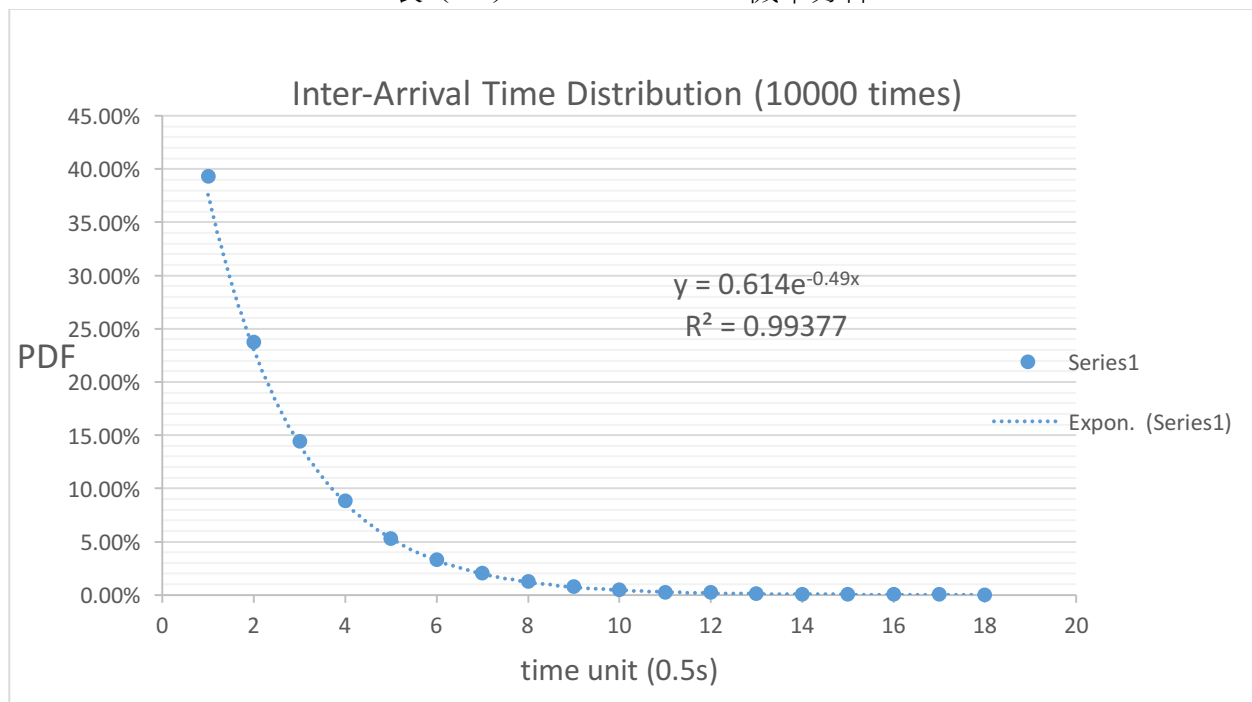
Problem 1 :

1. Please write a channel request generator. The arrival rate of the channel request follows Poisson distribution with mean 0.5. In addition, please show the inter-arrival time distribution of these channel requests is Exponential distribution.

先做一個符合 Poisson Distribution 的 channel request generator，我的做法是使用 C++ 提供的一個函式庫 <random>，使用 `poisson_distribution<int>` 以及 `default_random_engine` 去實作，模擬 inter-arrival time 的方式是去跑 10000 次 request，並且記錄下 request 之間的時間次數，換算成 PDF，再利用 Excel 畫出分布圖。結果如下表（一）、圖（一）：

Inter-arrival time	1	2	3	4	5	6	7	8	9
PDF	38.73%	22.98%	15.20%	8.81%	5.26%	3.27%	2.04%	0.0116111	1.26%
Inter-arrival time	10	11	12	13	14	15	16	17	18
PDF	0.77%	0.44%	0.23%	0.19%	0.10%	0.03%	0.04%	0.03%	0.02%

表（一） inter-arrival time 機率分佈



圖（一） Excel 的 inter-arrival time 曲線圖

可以從圖（一）中看出與理論值 $f(t) = \lambda * \exp(-\lambda * t)$ 差不多。

Problem 2-1:

建立 12 個 BS，每一個分配到的 arrival rate 以及 service rate 分別為 $((i \% 4)+1)*\lambda$ 、 $((i \% 4)+1)*\mu$ 其中 $\lambda = 1/\text{min}$ 、 $\mu = 0.2/\text{min}$ ，接著顧客的 request 會根據 Poisson distribution，每個 request 的 service time 會根據 exponential distribution。

我的做法是去跑 100000 次 loop，每 1000 次的時候每個 CELL 會根據 Poisson distribution request 一次（本題不考慮 burst request，因此在計算 blocking probability 會少上許多），當 channel 第一次被佔據時，會配置給他相對應的 service time（根據 exponential distribution），而每次 loop 時若通道使用時 $\rightarrow \text{channel}[i][j] \neq 0$ ， $\text{service_time}[i][j] -= 0.001$ ，如此一來便可以模擬每跑 1000 次 loop 剛好跑完 1min 的結果。

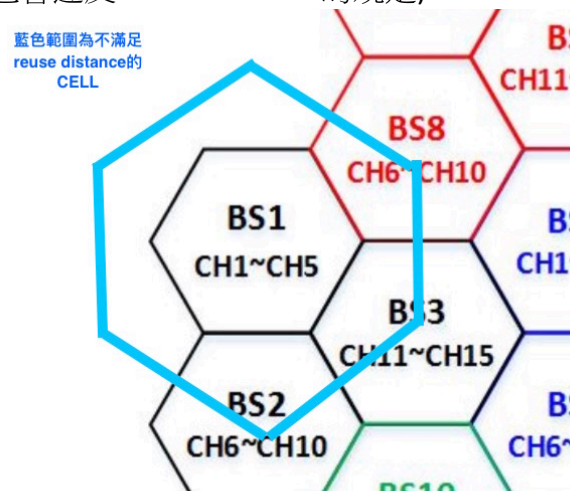
結果如下表：

Test time : 1000min

channel 1 blocking probability: 0.6148%
channel 2 blocking probability: 0.03155%
channel 3 blocking probability: 0.01017%
channel 4 blocking probability: 7.48%
channel 5 blocking probability: 0.5334%
channel 6 blocking probability: 0.0317%
channel 7 blocking probability: 0%
channel 8 blocking probability: 6.134%
channel 9 blocking probability: 0.6436%
channel 10 blocking probability: 0.02107%
channel 11 blocking probability: 0%
channel 12 blocking probability: 6.653%

Problem 2-2:

建立在 2-1 的基礎，此時要增加的部分是每次可能 blocking 時，要去判斷身邊的 channel 是否還有空閒，為此我先設定了一個 12x12 的 2 維矩陣，去把周圍會影響的記錄下來，有的話去找存在有那條 channel 中最富有的 BS(e.g. BS1 滿時，去尋找 reuse distance BS2、BS3、BS8 內 6~15 是否還有空閒，若都沒有就代表此時無法在借任何 channel 過來使用，因為就算借過來也會違反 reuse distance 的規定)



另外在每次迴圈時，要做 reallocation 的判斷，如果跟別人借用 channel 時，我會在借到的 channel 上做一個 tag，當自己有一個 channel free 時，會把借用的 channel 中的 service time 轉移到自己的 free channel 中。

結果如下表：

Test time: 1000min

channel 1	blocking probability:	0%
channel 2	blocking probability:	0%
channel 3	blocking probability:	0%
channel 4	blocking probability:	0%
channel 5	blocking probability:	0%
channel 6	blocking probability:	0%
channel 7	blocking probability:	0%
channel 8	blocking probability:	0.165%
channel 9	blocking probability:	0.1171%
channel 10	blocking probability:	0%
channel 11	blocking probability:	0%
channel 12	blocking probability:	0%