

Introducción

Ejercicio 1

En este ejercicio debemos simular un canal en el que los paquetes llegan a él con una distribución de Poisson. El largo de los paquetes corresponde a una exponencial de media $1/\mu$, o también de parámetro μ , en donde $\mu = 1$. También asumimos que el canal puede transmitir $C = 1$ bits por segundo. De esta forma, la tasa de servicio del canal será de $\mu * C = 1$ paquetes por segundo.

El algoritmo principal que simula el canal generará paquetes utilizando un Proceso de Poisson Homogéneo y luego atenderá los paquetes colocando los resultados en los arreglos correspondientes.

Inciso 1

Para poder estudiar la variable `tau_m` correspondiente al tiempo medio de demora el algoritmo irá agregando en un arreglo el tiempo que demora cada paquete desde que llega al canal hasta que efectivamente se envía. Estos, luego serán los datos simulados que utilizaremos en los ejercicios 1, 2, 3 y 4.

Para obtener el intervalo de confianza de la estimación de la variable `tau_m` utilizaremos el algoritmo visto en el teórico con pequeñas modificaciones.

Inciso 2

En este inciso debemos generar histogramas para los `Nsim` tiempos de demora obtenidos en el inciso 1 y para los nuevos valores que serán generados a partir de una exponencial de parámetro 0.5. Así conformaremos nuestra hipótesis nula, y en el siguiente inciso realizaremos un test de bondad de ajuste Chi cuadrado para comprobar la hipótesis.

Inciso 3

Para aplicar el test Chi-Cuadrado, dado que nuestros datos simulados son tiempos continuos necesitamos poder discretizarlos. Para esto distribuiremos los datos en 18 intervalos. Luego recorreremos los intervalos desde 0 hasta 18 y el primero intervalo `i` que observemos que su número de observaciones sea menor a 5, será unificado junto con los siguientes `i, i+1, .., n, n+1` intervalos en donde el ultimo `n+1` albergará los valores desde `[x, inf)`.

Con los intervalos ya conformados, obtendremos el estadístico T que nos permitirá evaluar la Chi-Cuadrado para obtener el p-valor y así finalmente rechazar o no rechazar nuestra hipótesis nula.

Inciso 4

Para analizar la variable tiempo de espera en cola o `t_c_media` utilizaremos los datos que generamos en el Inciso 1 en el algoritmo para el cálculo del intervalo de confianza. Estos datos los comparemos con `Nsim` nuevos datos simulados de una distribución Gamma con parámetros $\alpha=0.5$ y $\beta=2$. De esta forma, cumplimos con $\alpha * \beta = t_{c_media}$.

Estos nuevos datos luego los distribuiremos, al igual que en el Inciso 2, en los mismos intervalos que utilizamos para los datos simulados para luego poder calcular la probabilidad teórica de cada intervalo que

necesitaremos para el cálculo del estadístico T. Una vez que consigamos este estadístico podremos rechazar o no rechazar la hipótesis nula de que los **tiempos de espera en cola** provienen de una distribución Gamma con parámetros $\alpha=0.5$ y $\beta=2$.

Ejercicio 2

En este ejercicio, el ingreso de paquetes a un canal corresponde a una distribución de Poisson de parámetro λ y será simulada mediante un proceso de Poisson con λ .

En la simulación de los protocolos **Aloha Puro** y **Aloha Ranurado** como la tasa de servicio es constante e igual a 1, se asume que todos los paquetes tendrán una longitud igual a 1. Además, en el **Aloha Ranurado**, los paquetes solo podrán ingresar al canal en los momentos $t_0, t_1, t_2, t_3, \dots$ etc. Es decir, si un paquete está listo para ingresar al canal en el momento $(t_i + t_{i+1}) / 2$ deberá esperar hasta t_{i+1} para recién ingresar.

En ambos protocolos, si dos o más paquetes quieren enviarse en el mismo instante de tiempo, ambos colisionarán, se dañarán y serán descartados.

Para obtener los resultados se iterará sobre los distintos valores posibles de λ (0.1, 0.2, 0.3, ..., 3.0) y se realizarán simulaciones 10000 unidades de tiempo o segundos para estimar la tasa de uso del canal y la probabilidad de que la transmisión de un paquete sea exitoso.

Algoritmo y descripción de las variables

Ejercicio 2

La función que genera el proceso de Poisson homogéneo es llamada **Poisson_homogeneo** y fue extraída del teórico sin modificaciones. Luego tenemos 2 funciones llamadas **simular_canal_aloha** y **simular_canal_aloha_ranurado**. Ambas reciben como parámetros un λ para la tasa de generación y un T para el tiempo máximo de simulación.

Los algoritmos asumen que no hay detección de colisiones en el canal, por lo tanto como todos los paquetes son de ancho 1, cuando deban enviarse por el canal lo harán por 1 unidad de tiempo haya o no alguna colisión.

Simular_canal_aloha, Aloha Puro

En **simular_canal_aloha** primero se generan los paquetes, se acumulará el tiempo actual de procesamiento en **t_actual** y se guarda el array de paquetes en la lista **canal**. De aquí luego se van sacando paquetes mientras este no esté vacío y se revisa si el paquete tomado tendrá tiempo suficiente para ser enviado. Si hay tiempo suficiente, el paquete se agrega a la lista de **paquetes_enviados**, se lo elimina del canal y se avanza **t_actual** al tiempo de salida del paquete.

Si el paquete no tiene tiempo para ser enviado se lo agrega a la lista **paquetes_rotos**, se lo elimina del canal, se suma 1 a la variable **n_paquetes_rotos** y se adelanta el **t_actual** hasta el tiempo de salida del paquete que no se envió.

Si el tiempo actual es mayor al tiempo del paquete que se tomó se asume que estoy viendo un paquete que colisionó con alguno que ya se eliminó. Entonces se lo elimina, y se actualiza **t_actual** hasta el momento

en el que el paquete deja de enviarse por el canal.

Por ultimo, cuando en el canal solo queda un paquete, se revisa si hay tiempo suficiente antes de T para enviarlo. Si es así el paquete se envia sino se descarta y se finaliza la simulación.

Simular_canal_aloha_ranurado, Aloha Ranurado

`simular_canal_aloha_ranurado` es similar al aloha puro, pero con la diferencia de que los paquetes solo pueden ser enviados en los momentos de tiempo: t_0, t_1, t_2, \dots . Por lo tanto, se asume que si algún paquete esta listo para ser enviado en el instante $(t_i + t_{i+1})/2$ deberá esperar hasta t_{i+1} para ser enviado por el canal.

Este algoritmo toma y devuelve los mismos parámetros que `simular_canal_aloha`. Esto despues nos será útil para poder generalizar las funciones en los siguientes incisos.

El cuerpo de la funcion utiliza los mismos condicionales, y las mismas variables que vimos en la función anterior pero considerando ahora los tiempos redondeados al siguiente entero positivo con la funcion `ceil`.

De esta forma, estamos "juntando" todos los paquetes que se encontraban en un mismo intervalo de tiempo y podemos asegurar que si solo un paquete quiere enviar en un intervalo lo va a poder hacer y si habian dos o mas paquetes para ser enviados en un mismo intervalo todos van a colapsar y solo 'romperan' el canal por 1 intervalo de tiempo.

Ejercicio 2.a

Para esta sección se crearon 3 funciones, `tasa_uso_puro` que dado un λ y un T calcula la tasa de uso del canal con aloha puro, `tasa_uso_ranurado` que dado un λ y un T calcula la tasa de uso del canal con aloha ranurado y `ejercicio1` que dado una de estas dos funciones y su nombre imprime en consola los resultados correctos.

El calculo de las tasas de uso se resolvió calculando la cantidad de paquetes enviados en el canal durante la simulación sobre el Tiempo total de simulación. Esto es:

- $$\text{len}(\text{paquetes_enviados}) / T * 100$$

lo que nos da el porcentaje de uso del canal según el método que se haya simulado.

Ejercicio 2.b

Para esta sección se crearon 3 funciones, `prob_paquete_puro` que dado un λ y un T calcula la probabilidad de que un paquete sea enviado exitosamente con aloha puro, `prob_paquete_ranurado` que dado un λ y un T calcula la probabilidad de que un paquete sea enviado exitosamente con aloha ranurado y `ejercicio2` que dado una de estas dos funciones y su nombre imprime en consola los resultados correctos.

El calculo de las probabilidad se resolvió calculando la cantidad de paquetes enviados exitosamente en el canal sobre el total de paquetes simulados. Esto es:

- $$\text{cant_paquetes_enviados} / \text{neventos}$$

lo que nos da la probabilidad de que un paquete sea enviado correctamente en el canal.

Ejercicio 2.c

Para esta sección se creó una funcion `ejercicio3`, que toma una funcion de las dos creadas en el ejercicio 1, `tasa_uso_puro` o `tasa_uso_ranurado`, y su nombre e imprime por consola cada lambda con su tasa de uso.

La funcion `ejercicio3` crea dos listas vacias `tasas` y `lambs` y genera un for desde 1 hasta 31 para generar los lambdas desde 0.1, 0.2, 0.3, ..., 3.0 y evaluar con cada uno de ellos la tasa de uso del canal segun el método que se esté simulando. En las listas se guarda el lambda y su tasa.

Luego una vez que termina el ciclo se calcula el indice de la maxima tasa y se la devuelve en una tupla junto con su respectivo lambda.

Ejercicio 2.d

Para generar el grafico solicitado se utilizó `matplotlib.pyplot`. Se generaron todas las tasas para los dos métodos en sus respectivos arrays `tasas_uso_puro` y `tasas_uso_ranurado` y se generó un array con todos los lambdas (0.1, 0.2, ..., 3.0).

Luego se graficó en el `eje x` los lambdas y en `eje y` las tasas. En color azul los resultados para el aloha puro y en color rojo para aloha ranurado.

Resultados

Nota: Todos los resultados obtenidos se realizaron utilizando la semilla 0 para el generador pseudo-aleatorio de numeros.

Ejercicio 2

Ejercicio 2.a

Las tasas de uso para Aloha Puro obtenidas son:

Tasas	Lambdas
8.08	0.10
12.90	0.20
17.13	0.30
18.28	0.40
17.64	0.50
17.88	0.60
17.29	0.70
15.90	0.80

Tasas	Lambdas
14.46	0.90
13.38	1.00
12.35	1.10
10.91	1.20
9.44	1.30
8.70	1.40
7.57	1.50
6.74	1.60
6.16	1.70
4.88	1.80
4.33	1.90
3.87	2.00
3.34	2.10
2.96	2.20
2.41	2.30
1.73	2.40
1.77	2.50
1.45	2.60
1.24	2.70
0.96	2.80
0.88	2.90
0.56	3.00

Las tasas de uso para Aloha Ranurado obtenidas son:

Tasas	Lambdas
9.26	0.10
16.73	0.20
22.42	0.30
26.98	0.40
29.93	0.50

Tasas	Lambdas
33.30	0.60
33.95	0.70
35.67	0.80
36.01	0.90
36.26	1.00
36.85	1.10
36.73	1.20
35.53	1.30
34.18	1.40
33.52	1.50
32.21	1.60
31.33	1.70
30.34	1.80
28.82	1.90
27.16	2.00
25.07	2.10
24.21	2.20
23.89	2.30
22.48	2.40
19.85	2.50
19.73	2.60
18.01	2.70
16.62	2.80
16.33	2.90
14.92	3.00

Ejercicio 2.b

Las probabilidades de que un paquete sea transmitido sin daños para cada lambda, para los dos protocolos son:

Aloha Puro:

Tasas	Lambdas
0.82	0.10
0.67	0.20
0.54	0.30
0.44	0.40
0.37	0.50
0.29	0.60
0.25	0.70
0.20	0.80
0.17	0.90
0.13	1.00
0.11	1.10
0.09	1.20
0.07	1.30
0.06	1.40
0.05	1.50
0.04	1.60
0.04	1.70
0.03	1.80
0.02	1.90
0.02	2.00
0.01	2.10
0.01	2.20
0.01	2.30
0.01	2.40
0.01	2.50
0.01	2.60
0.00	2.70
0.00	2.80
0.00	2.90
0.00	3.00

Aloha Ranurado:

Tasas	Lambdas
0.91	0.10
0.82	0.20
0.73	0.30
0.67	0.40
0.61	0.50
0.55	0.60
0.52	0.70
0.45	0.80
0.39	0.90
0.38	1.00
0.34	1.10
0.32	1.20
0.27	1.30
0.25	1.40
0.22	1.50
0.20	1.60
0.19	1.70
0.17	1.80
0.15	1.90
0.14	2.00
0.12	2.10
0.11	2.20
0.10	2.30
0.09	2.40
0.08	2.50
0.07	2.60
0.07	2.70
0.06	2.80
0.06	2.90

Tasas	Lambdas
0.05	3.00

Ejercicio 2.c

El valor de lambda que optimiza la tasa de uso del canal es:

Para Aloha puro

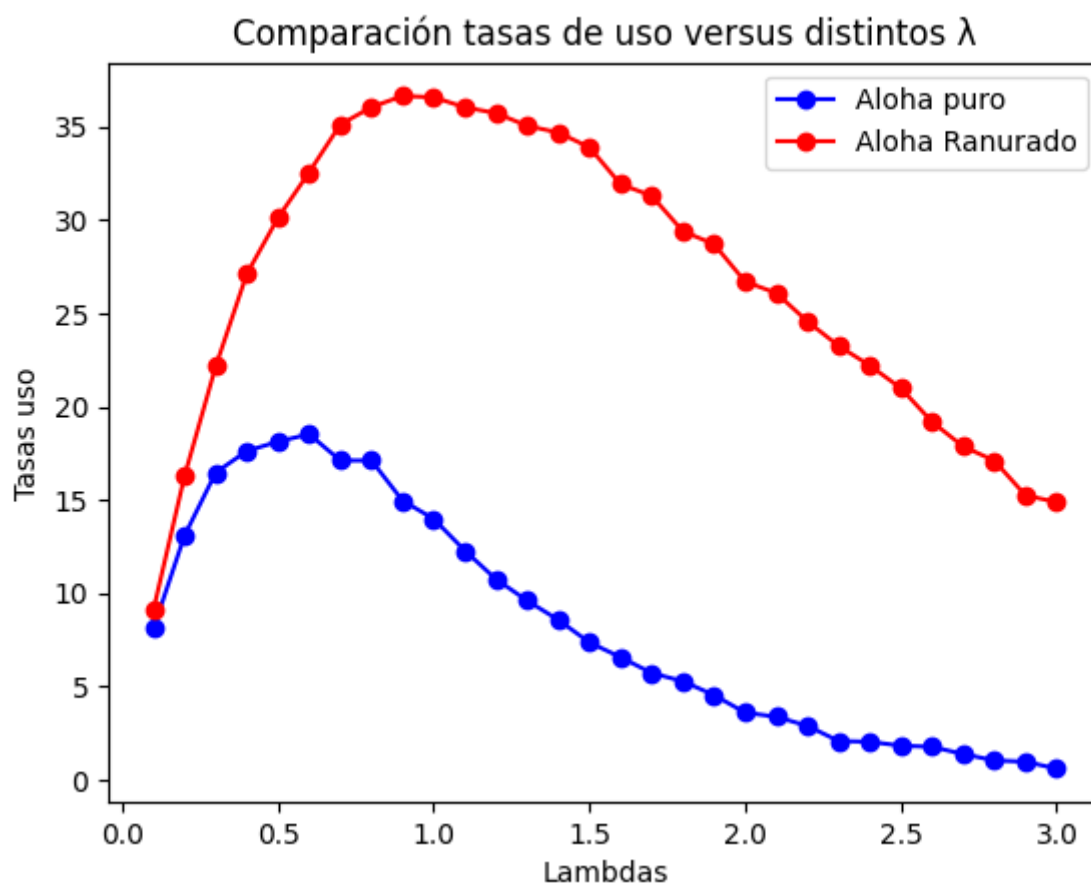
Lambda = 0.6, con una tasa de uso del 18.73 %

Para Aloha Ranurado

Lambda = 1.0, con una tasa de uso del 37.34 %

Ejercicio 2.d

El gráfico que compara la tasa de uso versus el lambda para ambos protocolos es:



Ejercicio 2.e

Gracias a la información obtenida podemos concluir que el protocolo Aloha Ranurado provee una mejor utilización del canal

Conclusiones

