



MODELADO Y SIMULACIÓN DE SISTEMAS DINÁMICOS

TRABAJO PRÁCTICO 2 SISTEMAS DE EVENTOS DISCRETOS

Tomás Fernández de Luco, F-3443/6

Comentarios generales

Si bien en la entrega se encuentra únicamente el archivo con el modelo del sistema completo, en cada uno de los apartados se describe la definición de cada componente, junto con gráficos y comentarios pertinentes.

1 Fuente

En este modelo no se necesita un conjunto de valores de entrada ni una función de transición externa, pues solamente emite los trabajos. Como consecuencia, el estado sólo necesita saber el tiempo de avance, el cual se determina al azar en el momento de cada transición interna.

$$\begin{aligned}X &= \emptyset \\Y &= \mathbb{R}^+ \\S &= \mathbb{R}^+ \\ \delta_{int}(s) &= random(0, T_{max}) \\ \lambda(s) &= random(J_{min}, J_{max}) \\ ta(\sigma) &= \sigma\end{aligned}$$

Donde $random(a, b)$ es un valor tomado al azar con una distribución uniforme en el intervalo $[0, 1]$. Debajo pueden verse los valores emitidos en una simulación con los parámetros propuestos en el enunciado.

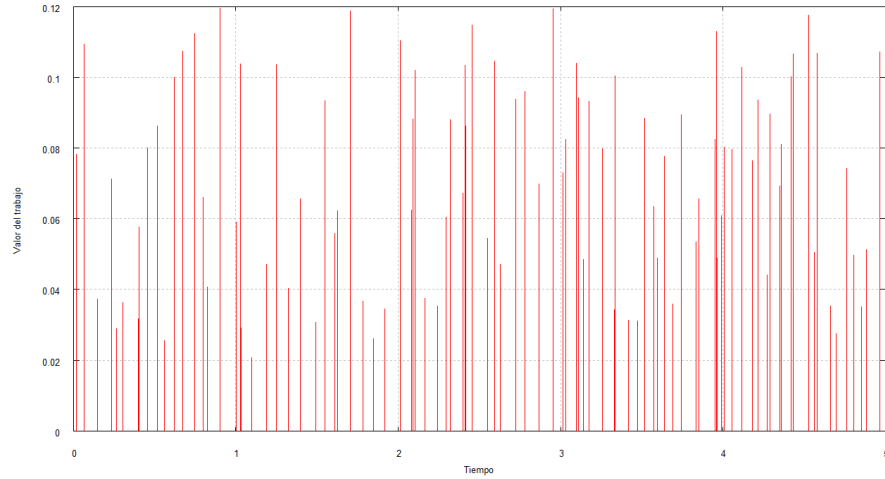


Figura 1: Simulación de la Fuente.

2 Procesador

Para el caso del procesador, el estado contiene dos valores: el tiempo de avance, que representa el tiempo de procesamiento del último trabajo recibido, o infinito en caso de estar libre; y un valor que indica si se encuentra procesando actualmente. La componente interesante en este modelo corresponde a la función de transición externa, la cual acepta un trabajo y toma su valor como tiempo de avance en caso de estar libre, mientras que lo descarta y reduce su tiempo de avance en caso contrario. El procesador comienza en un estado pasivo esperando la llegada de algún trabajo.

$$\begin{aligned}
 X &= \mathbb{R}^+ \\
 Y &= \{1\} \\
 S &= \mathbb{R}^+ \times \{true, false\} \\
 \delta_{ext}((\sigma, busy), e, x) &= \begin{cases} (\sigma - e, true) & \text{si } busy = true \\ (x, true) & \text{si } busy = false \end{cases} \\
 \delta_{int}(s) &= (\infty, false) \\
 \lambda(s) &= 1 \\
 ta(\sigma, busy) &= \sigma
 \end{aligned}$$

La siguiente simulación muestra una secuencia de valores emitidos por la fuente y las señales de procesamiento completo, donde pueden verse que hay trabajos ignorados en los intervalos en los que el procesador esté ocupado.

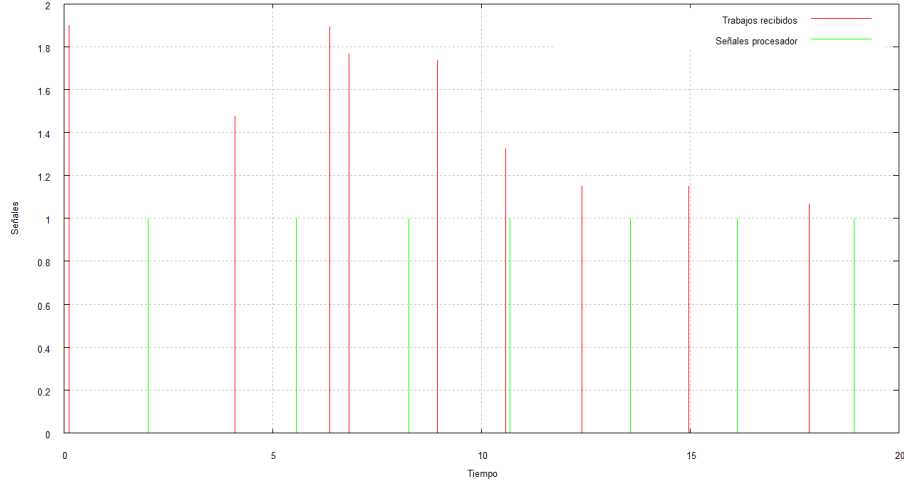


Figura 2: Simulación de Fuente y Procesador.

3 Cola

El estado de la cola, además del tiempo de avance que usamos casi siempre, son una lista de reales (que representa los trabajos encolados) y el estado de trabajo del procesador. La parte interesante del modelo de la cola radica en su función de transición externa, ya que el estado interno sólo oscila entre estados transitorios y pasivos dependiendo del mensaje que deba emitir al procesador. Con respecto a la transición externa, la distinción principal en su comportamiento radica en el puerto por el que llegue la señal: si proviene de la fuente, debe encolarse el trabajo recibido y revisar si debe comunicársele al procesador; si viene del procesador, hay que enviarle el próximo trabajo en la cola, si es que existe. El estado inicial es en un estado pasivo con la cola vacía y un indicador de procesador libre.

$$\begin{aligned}
X &= (\mathbb{R}^+ \times \{0\}) \cup \{(1, 1)\} \\
Y &= \mathbb{R}^+ \\
S &= \mathbb{R}^* \times \{0, \infty\} \times \{true, false\} \\
\delta_{ext}((q, \sigma, busy), e, (v, port)) &= \begin{cases} (\langle v \rangle \frown q, \sigma, busy) & \text{si } port = 0 \wedge busy = true \\ (\langle v \rangle \frown q, 0, busy) & \text{si } port = 0 \wedge busy = false \\ (q, \sigma, false) & \text{si } port = 1 \wedge q = \langle \rangle \\ (q, 0, false) & \text{si } port = 1 \wedge q \neq \langle \rangle \end{cases} \\
\delta_{int}(\langle v \rangle \frown q, \sigma, busy) &= (q, \infty, true) \\
\lambda(\langle v \rangle \frown q, \sigma, busy) &= v \\
ta(q, \sigma, busy) &= \sigma
\end{aligned}$$

Los siguientes gráficos son los resultados de una simulación del sistema fuente-cola-procesador. En ellos pueden verse las señales de entrada y salida de la cola, pudiendo observarse que los valores emitidos son los mismos que los recibidos, pero más separados temporalmente dadas las esperas de tiempos de procesamiento. A su vez, se nota que cada evento de salida ocurre en el mismo momento que una señal proveniente del procesador cuando terminó con el trabajo anterior.

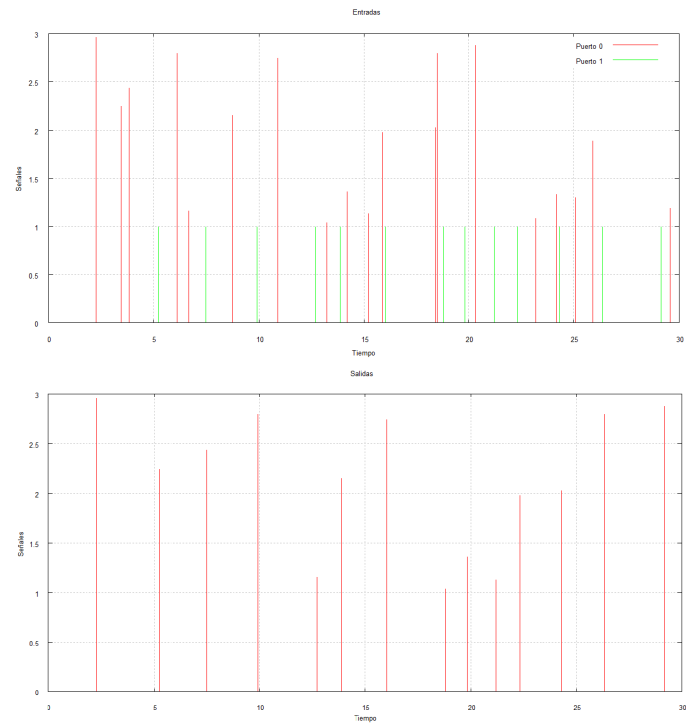


Figura 3: Simulación de la Cola.

4 Sensor

El sensor necesita preservar en su estado el contador de trabajos acumulados y el tiempo de vida del estado. Como emite eventos únicamente cuando el valor del contador cambia, se mantendrá en un estado pasivo hasta recibir algún evento de entrada, donde pasará a uno transitorio para emitir el valor del contador antes de volver a esperar. Esperablemente, su estado inicial es en un estado pasivo con un contador en cero.

$$\begin{aligned}
 X &= (\mathbb{R}^+ \times \{0\}) \cup (\mathbb{R}^+ \times \{1\}) \\
 Y &= \mathbb{N} \\
 S &= \{0, \infty\} \times \mathbb{N} \\
 \delta_{ext}((\sigma, n), e, (v, port)) &= \begin{cases} (0, n+1) & \text{si } port = 0 \\ (0, n-1) & \text{si } port = 1 \end{cases} \\
 \delta_{int}(\sigma, n) &= (\infty, n) \\
 \lambda(\sigma, n) &= n \\
 ta(\sigma, n) &= \sigma
 \end{aligned}$$

En el gráfico debajo puede verse cómo el contador interno del sensor se va modificando en los instantes de tiempo en los que la cola recibe y emite una señal.

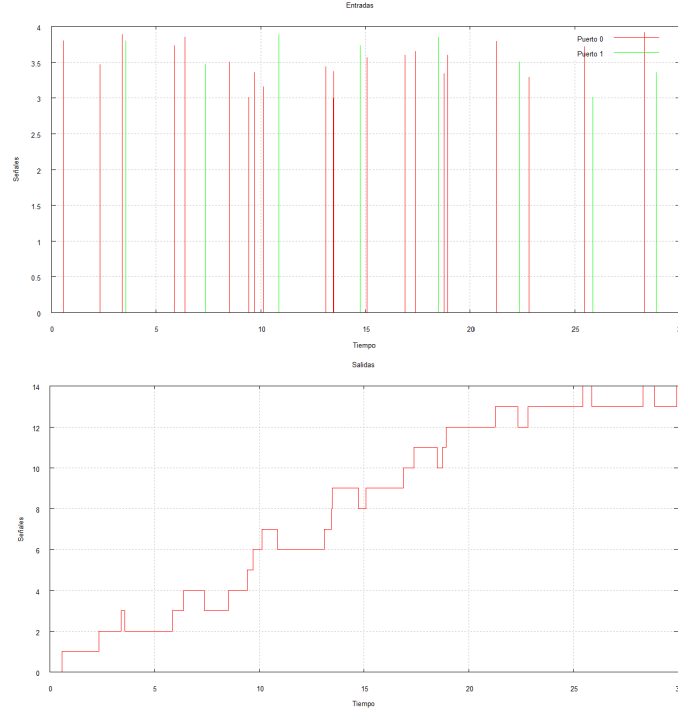


Figura 4: Simulación del Sensor.

5 Filtro

El filtro necesita preservar la probabilidad de aceptación de los trabajos, el valor del último trabajo recibido y el tiempo de vida. En caso de que llegue una probabilidad nueva, actualiza su valor interno. En caso de que sea un trabajo, decide si aceptarlo y emite su señal en el puerto correspondiente. Más allá de eso, se queda esperando eventos en un estado pasivo.

$$\begin{aligned}
 X &= (\mathbb{R}^+ \times \{0\}) \cup ([0, 1] \times \{1\}) \\
 Y &= (\mathbb{R}^+ \times \{0\}) \cup (\mathbb{R}^+ \times \{1\}) \\
 S &= \mathbb{R}^+ \times \{0, \infty\} \times [0, 1] \\
 \delta_{ext}((job, \sigma, p), e, (v, port)) &= \begin{cases} (v, 0, p) & \text{si } port = 0 \\ (job, \sigma, v) & \text{si } port = 1 \end{cases} \\
 \delta_{int}(job, \sigma, p) &= (job, \infty, p) \\
 \lambda(job, \sigma, p) &= \begin{cases} (job, 0) & \text{si } rand < p \\ (job, 1) & \text{si } rand \geq p \end{cases}, rand = random(0, 1) \\
 ta(job, \sigma, p) &= \sigma
 \end{aligned}$$

El gráfico siguiente muestra la clasificación de trabajos que hizo un filtro que inició con una probabilidad de aceptación de 0.5 y la cambió a 0.1 en el instante 5. La diferencia en proporciones de colores de los trabajos refleja el cambio interno en el proceso de selección.

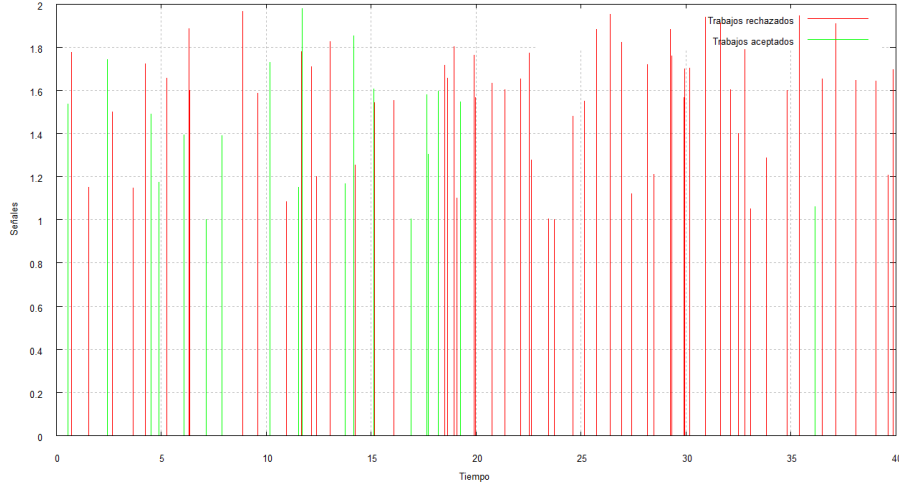


Figura 5: Simulación del Filtro.

6 Muestreador

El muestreador necesita almacenar el valor que debe emitir y el tiempo de vida. A diferencia de la mayoría de los modelos anteriores, no se mantiene en un estado pasivo sino que inicia emitiendo su valor inicial y en cada transición interna vuelve a un estado con tiempo de vida igual a T . Las señales recibidas actualizan el valor interno y reducen el tiempo de vida para preservar la frecuencia de emisión cada T unidades.

$$\begin{aligned}
 X &= \mathbb{R}^+ \\
 Y &= \mathbb{R}^+ \\
 S &= \mathbb{R}^+ \times \mathbb{R}^+ \\
 \delta_{ext}((last, \sigma), e, v) &= (v, \sigma - e) \\
 \delta_{int}(last, \sigma) &= (last, T) \\
 \lambda(last, \sigma) &= last \\
 ta(last, \sigma) &= \sigma
 \end{aligned}$$

En el ejemplo siguiente se muestran las señales de entrada y salida del Muestreador en una simulación del sistema hasta este momento. Puede verse cómo las señales de salida periódicamente transmiten el último valor recibido por parte del Sensor.

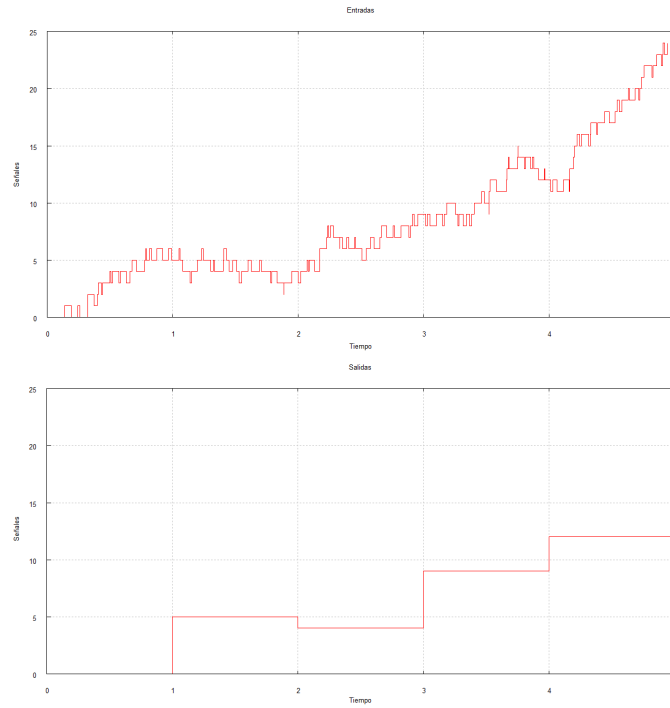


Figura 6: Simulación del Muestreador.

7 Controlador

El modelo para el controlador fue implementado como un sistema acoplado de tiempo discreto usando elementos de las librerías de PowerDevs. El mismo recibe el largo actual de la cola de procesamiento en cada instante de tiempo de parte del Muestreador y emite la probabilidad de aceptación para intentar mantener el valor de la cola lo más cerca posible al valor deseado l_{ref} .

8 Sistema Completo

A continuación se muestran los resultados de múltiples simulaciones del sistema, modificando algunos parámetros puntuales para analizar qué efecto tienen en el funcionamiento general. Para cada caso, se grafican el largo muestreado de la cola y la probabilidad de aceptación del filtro. Primero se ve el caso en el que se usan todos los parámetros sugeridos por el enunciado del trabajo. Puede verse que la cola se llena aceleradamente en los primeros instantes de tiempo, logrando estabilizar el largo alrededor del valor deseado usando una probabilidad cercana a 0.7, la cual varía ligeramente en los picos de llegada de trabajos.

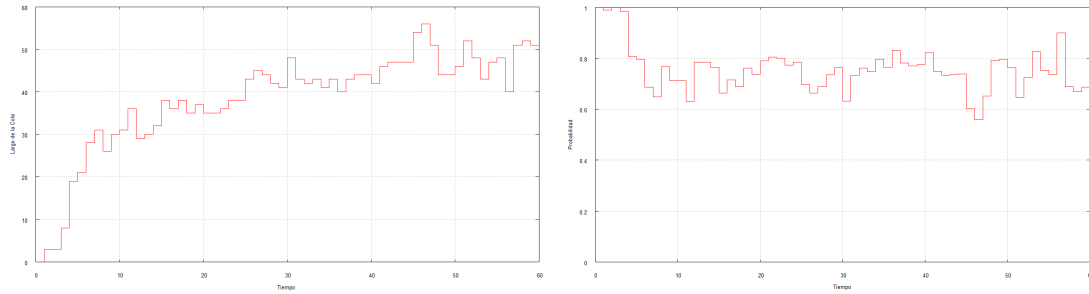


Figura 7: Simulación del Sistema Completo con los parámetros sugeridos.

Como el sistema parece haberse estabilizado sin mayores inconvenientes en el escenario anterior, se consideró interesante analizar su desempeño en un entorno más demandante. La siguiente simulación usó un valor de 0.05 para el T_{min} de la fuente, aumentando notablemente la cantidad de trabajos generados. Aún así, puede verse que también se logra estabilizar el largo de la cola, aunque para eso se acepten sólo alrededor de un tercio de los trabajos recibidos.

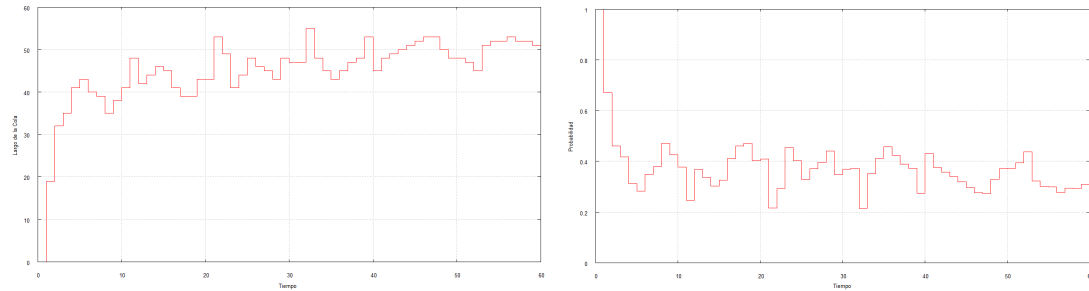


Figura 8: Simulación del Sistema Completo con mayor generación de trabajos.

Los parámetros que parecían interesantes para analizar eran los factores que ajustan el sistema de ecuaciones del controlador. Modificar l_{ref} no dio ningún resultado llamativo pues solamente variaba la velocidad con la que se llegaba al punto en el que la longitud de la cola se estabiliza. Sin embargo, aumentar drásticamente k_1 a un valor como 0.2 vuelve al controlador mucho más susceptible a la distancia entre los largos deseado y actual de la cola, haciendo que la probabilidad oscile prácticamente entre 0 y 1 en cada instante de tiempo mientras trata de estabilizar el valor

cerca de l_{ref} . Como resultado, el largo de la cola es más inestable cerca de la longitud deseada, con numerosos picos y valles en el número de trabajos acumulados.

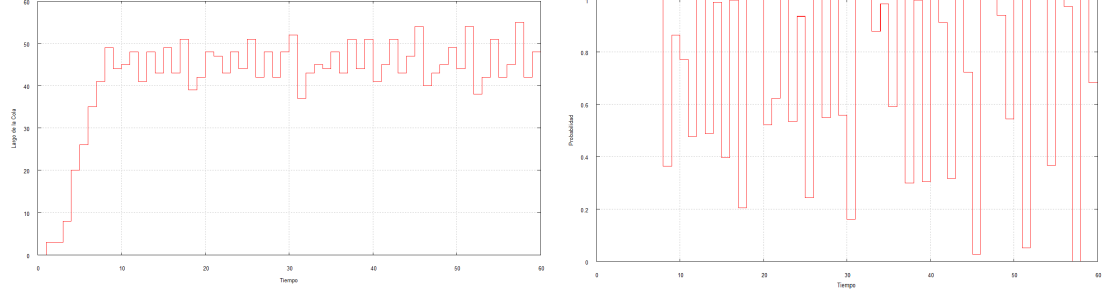


Figura 9: Simulación del Sistema Completo con un mayor $k1$.

Esperablemente, elevar $k2$ tiene un efecto inverso, en el que el sistema tiende a variar los valores de una manera mucho más suave. Esta relajación hace que el filtro no pueda reducir lo suficientemente rápido la frecuencia de ingreso de trabajos luego de la acumulación inicial, teniendo que disminuir drásticamente la probabilidad de aceptación para cuando ya hay más de 90 trabajos encolados. Luego de eso, el largo de la cola se normaliza con variaciones menores entre cada instante de tiempo que en los casos anteriores.

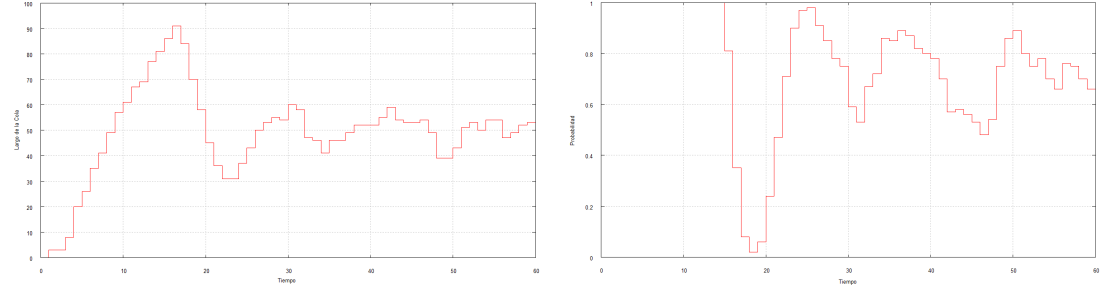


Figura 10: Simulación del Sistema Completo con un mayor $k2$.