

Trabajo Especial de Modelos y Simulación

Lucas Alonso
Guillermo Incatasciato

16 de Junio, 2017

FAMAF, UNC

1 Introducción

En este proyecto estudiaremos el *tiempo operativo* de un sistema a través de la simulación del mismo.

El sistema sera un Lavadero Automático, que inicialmente consta de N maquinas de lavar en uso, S de repuesto, y un operario que repara las maquinas que se van rompiendo. Cuando una maquina en uso se rompe inmediatamente se manda a reparación y se la reemplaza por una de repuesto. El operario arregla las maquinas rotas una a la vez. Para ser operativo el Lavadero tiene que tener N maquinas funcionando, por lo que llamaremos *tiempo operativo* al tiempo transcurrido hasta que el lavadero deja de tener N maquinas en funcionamiento, o lo que es lo mismo, tiene $S + 1$ en reparación. Se conoce la distribución del tiempo de funcionamiento de una maquina cualquiera hasta su ruptura, y la distribución del tiempo de reparación de una maquina por el operario.

Se quiere conocer el tiempo operativo medio del sistema y su desviación estandar.

Estimaremos estos parametros mediante una simulación del Lavadero. Ademas, tambien mediante simulaciones veremos, en el caso de que se quiera aumentar el tiempo operativo medio del sistema, si es mejor agregar un operario o comprar una maquina nueva de repuesto.

2 Algoritmo y Descripción de las Variables

En esta sección describiremos los Algoritmos para la simulación del sistema con un operario y con dos operarios.

2.1 Variables

Para el Algoritmo de un solo operario las variables son:

t: el tiempo transcurrido desde el comienzo de la simulación.

rotas: la cantidad de maquinas rotas al momento.

final_de_reparacion: el momento en que el operario va a terminar de arreglar la maquina que esta reparando(si no hay ninguna maquina en reparación su valor es infinito).

tiempos_de_fallas: es una lista que contiene los tiempos en que fallaran las maquinas que estan en uso(se mantiene ordenada de menor a mayor).

Para el Algoritmo de dos operarios las variables son las misma salvo que **final_de_reparacion** no esta, y en remplazo de ella tenemos:

tiempos_de_reparacion: es una lista de tamaño 2, que contiene los tiempos en que estaran arregladas las maquinas en reparación. Si no hay maquinas en reparación la lista tendra sus valores en infinito, si una sola maquina esta siendo reparada uno de los valores estara en infinito. Esto es porque ahora hay dos operarios para reparar las maquinas.

2.2 El Algoritmo para un solo operari3n

Utilizaremos F_F para referirnos a la distribuci3n del tiempo de falla de una maquina y F_R para la distribuci3n del tiempo de reparaci3n de una maquina.

El Algoritmo consiste en ir saltando de evento en evento en la linea del tiempo (se va incrementando la variable t), los eventos pueden ser la falla de una maquina o una reparaci3n completada. Cuando finalmente el sistema deje de ser operativo ($rotas = S + 1$) la simulaci3n termina. Cuando la simulaci3n comienza se tiene $t = 0$, cero tiempo transcurrido, $rotas = 0$, ya que no hay maquinas rotas a3n y $final_de_reparacion = infinito$, ya que tampoco hay maquinas en reparaci3n. Nos referiremos a t como tiempo transcurrido, a $rotas$ como cantidad de maquinas rotas, a $final_de_reparacion$ como tiempo de reparaci3n completada y a $tiempos_de_fallas$ como tiempos de fallas.

Cuando el algoritmo inicia se generan N variables aleatorias con distribuci3n F_F que se colocaran en la lista tiempos de fallas, que representan los momentos de fallas de las N maquinas en uso, luego se ordena la lista de menor a mayor (por lo que en la primera posici3n de la lista se tendra la falla mas cercana). Luego de esto se entra a un `while true`, que contiene un solo `if`.

El `if` principal verifica si el proximo evento es una reparaci3n completada o una falla.

En el caso de una falla:

1. Se aumenta la cantidad de maquinas rotas.
2. Se avanza el tiempo transcurrido hasta el momento de la falla.
3. En caso de tener mas maquinas rotas que de repuesto ($S + 1$) se termina la simulaci3n.
4. En caso de tener menos rotas que de repuesto: la maquina rota es reemplazada por una de repuesto y se la manda a reparaci3n, por lo que se genera una variable aleatoria F_F y se agrega a la lista de tiempos de fallas un nuevo valor igual a el tiempo transcurrido + F_F (representaria el momento de falla de la maquina de repuesto puesta en funcionamiento), se quita el tiempo correspondiente de la maquina que ya esta en reparaci3n.
5. Si la cantidad de maquinas rotas es 1, quiere decir que el operario no estaba arreglando ninguna maquina. Se genera una variable aleatoria con distribuci3n F_R y se cambia al tiempo de reparaci3n completada a tiempo transcurrido + F_R .

En caso de que se haya completado una reparaci3n:

1. Se disminuye la cantidad de maquinas rotas.
2. Se avanza el tiempo transcurrido hasta el momento de la reparaci3n completada.
3. En caso de a3n haber maquinas rotas, se comienza a arreglar otra maquina, se genera una variable aleatoria con distribuci3n F_R y se cambia el tiempo de reparaci3n completada a tiempo transcurrido + F_R .
4. En caso de que no queden maquinas rotas: Se coloca el tiempo de reparaci3n completada en infinito.

Algorithm 1 Algoritmo para n máquinas en funcionamiento, s de repuesto y un operario

procedure SIMULACIÓN

▷ Inicialización de variables

$t \leftarrow 0$

$rotas \leftarrow 0$

$final_de_reparacion \leftarrow \infty$

$tiempos_de_fallas \leftarrow []$

▷ Se generan los tiempos de fallas de las máquinas

for $i \leftarrow 1$ **to** n **do**

▷ Se agrega un tiempo con distribución F_F a la lista $tiempos_de_fallas$

$tiempos_de_fallas \triangleleft F_F$

end for

▷ Se ordena de menor a mayor los tiempos de fallas

$ordenar(tiempos_de_fallas)$

while $True$ **do**

if $tiempos_de_fallas[0] \leq final_de_reparacion$ **then**

$t \leftarrow tiempos_de_fallas[0]$

$rotas \leftarrow rotas + 1$

if $rotas = s + 1$ **then**

return t

end if

if $rotas < s + 1$ **then**

$tiempos_de_fallas[0] \leftarrow t + F_F$

$ordenar(tiempos_de_fallas)$

end if

if $rotas = 1$ **then**

$final_de_reparacion \leftarrow t + F_R$

end if

else

$t \leftarrow final_de_reparacion$

$rotas \leftarrow rotas - 1$

if $rotas > 0$ **then**

$final_de_reparacion \leftarrow t + F_R$

end if

if $rotas = 0$ **then**

$final_de_reparacion \leftarrow \infty$

end if

end if

end while

end procedure

2.3 El Algoritmo para dos operarios

El Algoritmo para dos operarios cambia levemente del anterior. Al tener dos operarios hay que guardar los tiempos de finalización de reparación de cada uno, en el caso de que esten reparando una maquina, por eso en vez de **final_de_reparacion**, tendremos una lista de largo 2 llamada **tiempos_de_reparacion**. Los eventos posibles seran ahora tres, falla de una maquina, reparación completa del operario 1, reparación completa del operario 2. El if para verificar si el proximo evento es una falla o un final de reparación, compara el tiempo de la falla mas proxima con el minimo de **tiempos_de_reparacion**.

En el caso de una falla:

1. Se aumenta la cantidad de maquinas rotas.
2. Se avanza el tiempo transcurrido hasta el momento de la falla.
3. En caso de tener mas maquinas rotas que de repuesto($S + 1$) se termina la simulación.
4. En caso de tener menos rotas que de repuesto: la maquina rota es reemplazada por una de repuesto y se la manda a reparación, por lo que se genera una variable aleatoria F_F y se agrega a la lista de tiempo de fallas un nuevo valor igual a el tiempo transcurrido + F_F (representaria el momento de falla de la maquina de repuesto puesta en funcionamiento), se quita el tiempo correspondiente de la maquina que ya esta en reparación.
5. Si la cantidad de maquinas rotas es 1 o 2 , quiere decir que habia un operario libre. Se genera una variable aleatoria con distribución F_R y se agrega a **tiempos_de_reparacion** un valor con el tiempo transcurrido + F_R .

En caso de que se haya completado una reparación:

1. Se disminuye la cantidad de maquinas rotas.
2. Se avanza el tiempo transcurrido hasta el momento de la reparación completada.
3. En caso de aún haber maquinas rotas, el tiempo de reparacion correspondiente al operario que acaba de terminar, es decir, el minimo de **tiempos_de_reparacion** , es reemplazado por el tiempo actual t mas un Y con $Y \sim F_R$. Es decir este operario se encargara de reparar otra de las rotas.
4. En el caso de que no queden maquinas rotas, se colocan los dos valores de **tiempos_de_reparacion** en infinito, ya que ninguno de los operarios tiene maquinas por reparar.

Algorithm 2 Algoritmo para n máquinas en funcionamiento, s de repuesto y dos operarios

procedure SIMULACIÓN

▷ Inicialización de variables

$t \leftarrow 0$

$rotas \leftarrow 0$

$tiempos_de_reparacion \leftarrow [\infty, \infty]$

$tiempos_de_fallas \leftarrow []$

▷ Se generan los tiempos de fallas de las máquinas

for $i \leftarrow 1$ **to** n **do**

▷ Se agrega un tiempo con distribución F_F a la lista $tiempos_de_fallas$

$tiempos_de_fallas \triangleleft F_F$

end for

▷ Se ordena de menor a mayor los tiempos de fallas

$ordenar(tiempos_de_fallas)$

while $True$ **do**

if $tiempos_de_fallas[0] \leq \min(tiempos_de_reparacion)$ **then**

$t \leftarrow tiempos_de_fallas[0]$

$rotas \leftarrow rotas + 1$

if $rotas = s + 1$ **then**

return t

end if

if $rotas < s + 1$ **then**

$tiempos_de_fallas[0] \leftarrow t + F_F$

$ordenar(tiempos_de_fallas)$

end if

if $rotas = 1$ o $rotas = 2$ **then**

se coloca $t + F_R$ en la posición que contenga ∞ en $tiempos_de_reparacion$

end if

else

$t \leftarrow \min(tiempos_de_reparacion)$

$rotas \leftarrow rotas - 1$

if $rotas > 0$ **then**

se coloca $t + F_R$ en la posición que contenga el menor tiempo en

$tiempos_de_reparacion$

end if

if $rotas = 0$ **then**

$tiempos_de_reparacion \leftarrow [\infty, \infty]$

end if

end if

end while

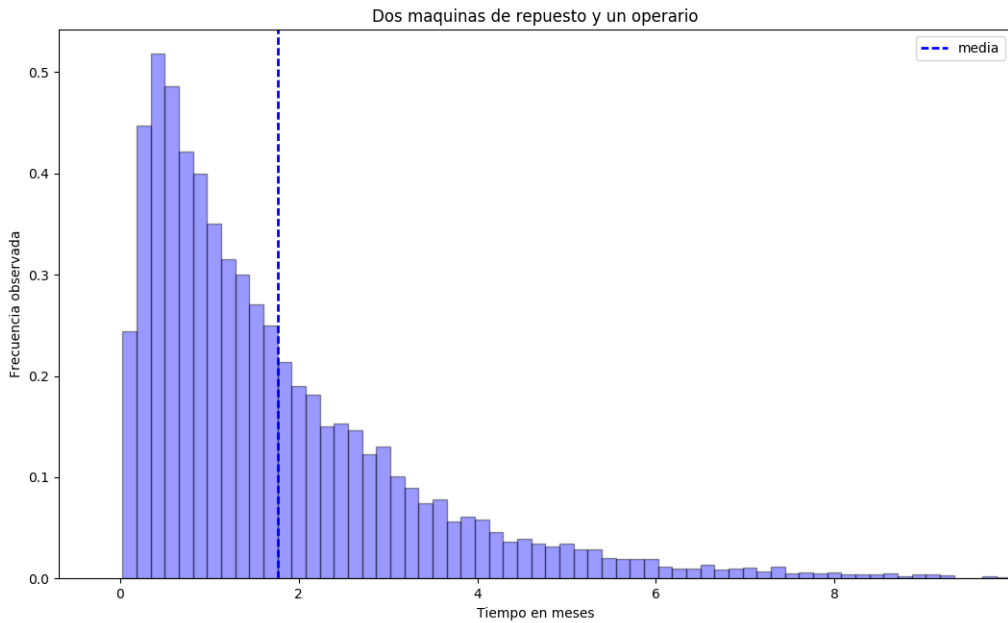
end procedure

3 Resultados

El sistema específico que utilizaremos para hacer las simulaciones sera , $N = 5$, $S = 2$. Consideraremos la unidad de medida de tiempo el mes. Los tiempos de funcionamiento de las maquinas hasta descomponerse son variables independientes exponenciales con tiempo medio de fallo $T_F = 1$, es decir la distribución del tiempo de falla de una maquina es $F_F \sim \varepsilon(1)$. El tiempo de reparación de una máquina es una variable exponencial independiente a las anteriores con tiempo medio de reparación $T_R = 1/8$, por lo que tenemos que la distribución del tiempo de reparación de una maquina es $F_R \sim \varepsilon(8)$.

3.1 Resultados de la simulación del sistema con 5 máquinas en uso, 2 máquinas de repuesto y 1 operario

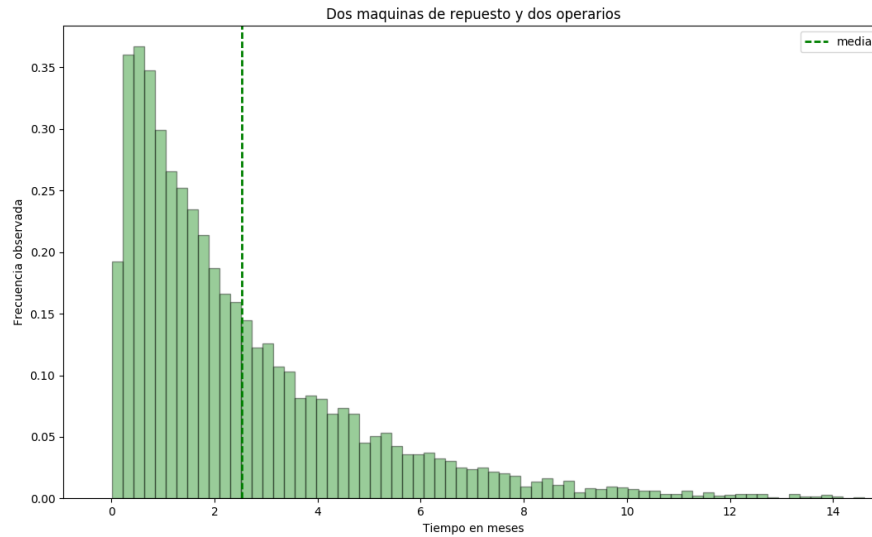
El siguiente histograma representa los valores obtenidos de 10000 simulaciones de tiempo de fallo del sistema.



$$\mu = 01.7614 \quad \sigma = 1.6124$$

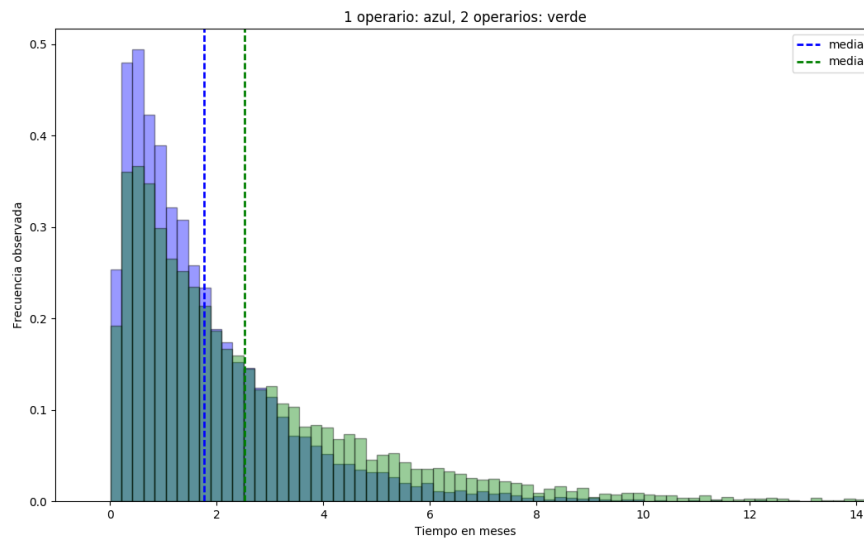
Ahora veremos que conviene mas , si sumar un operario o agregar una nueva maquina de repuesto, en el caso de que se quiera aumentar el tiempo operativo medio del sistema. Para eso simularemos el sistema con una maquina de repuesto mas , y otra simulación con un operario mas. Considerando que el tiempo de falla de la nueva maquina es igual a las que ya estan en el sistema y que el tiempo de reparación del nuevo operario es igual al del operario que ya esta trabajando.

3.2 Resultados de la simulación del sistema con 5 máquinas en uso, 2 máquinas de repuesto y 2 operarios



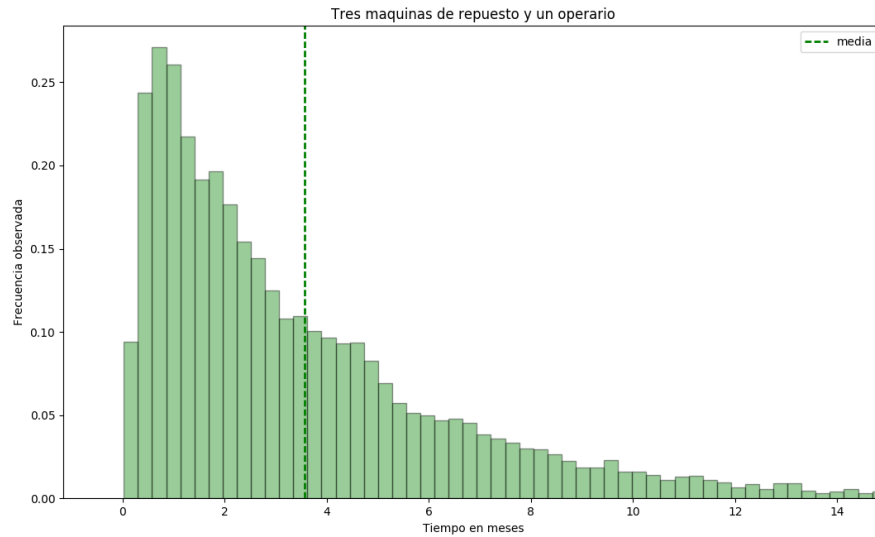
$$\mu = 2.5704 \quad \sigma = 2.4382$$

Comparación entre el sistema original y el sistema con dos operarios:



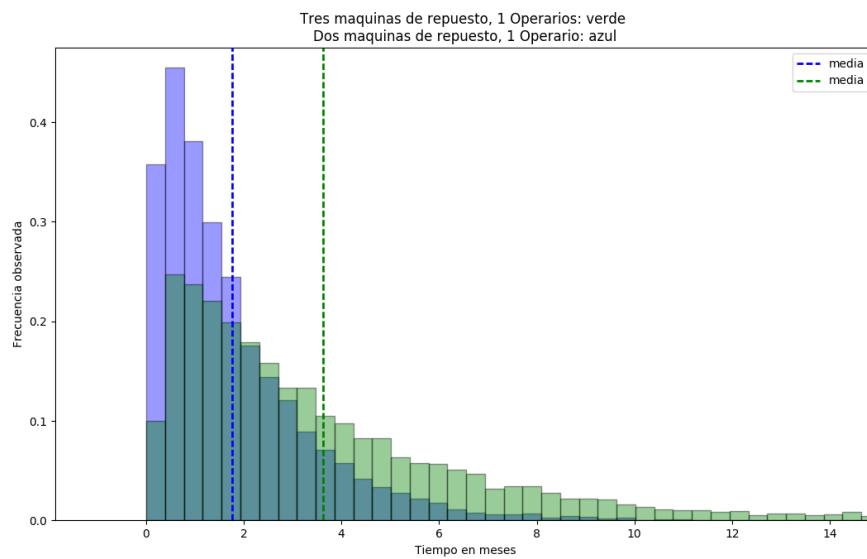
Como se puede observar en el grafico de arriba, al emplear un operario mas en el lavadero, se logra que en promedio, el lavadero pueda estar en funcionamiento por aproximadamente 0.8 meses mas.

3.3 Resultados de la simulación del sistema con 5 máquinas en uso, 3 máquinas de repuesto y 1 operario



$$\mu = 3.5994 \quad \sigma = 3.2763$$

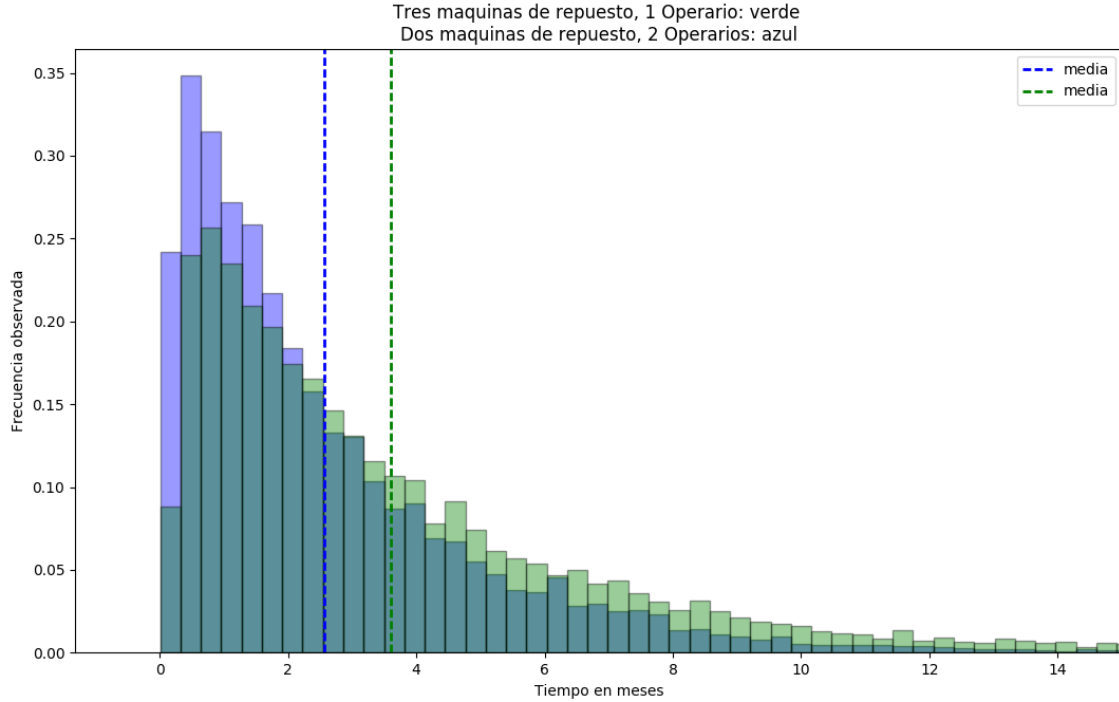
Comparación entre el sistema original y el sistema con una máquina de repuesto mas:



En este caso la mejora con respecto al original seria aproximadamente de 1.84 meses.

3.4 Comparación de las mejoras del sistema

El siguiente gráfico muestra la comparación entre las dos mejoras:



Esta figura muestra que la mejora correspondiente a agregar una maquina es mas efectiva que la de agregar un operario.

4 Conclusiones

Finalmente la estimación que logramos mediante la simulación de 10000 fallos del sistema es que el tiempo operativo medio del sistema es $\mu = 1.7614$ meses y su desviación estandar $\sigma = 1.6124$, Ademas pudimos comprobar que para aumentar el tiempo operativo medio del sistema conviene agregar una maquina de repuesto ya que por la simulación estimamos que el tiempo medio seria $\mu = 3.5994$ meses con esta mejora , en comparación con la simulación con un operario mas , que estimo un tiempo medio operativo de $\mu = 2.570$.