

Simulación del funcionamiento de una fábrica

Pedro Pablo Álvarez Portelles Abel Llerena Domingo Daniel Ángel Arró Moreno

June 9, 2024

Abstract

Este estudio presenta un modelo de simulación para optimizar la eficiencia y rentabilidad en un entorno de producción industrial. El modelo se centra en una fábrica que opera con un conjunto de máquinas en funcionamiento y máquinas de repuesto. Se consideran variables críticas como el tiempo de vida y el tiempo de reparación de las máquinas, el costo de reparación y reemplazo, y la ganancia generada por la fábrica. El objetivo es proporcionar una herramienta de decisión para determinar si es más rentable reparar o reemplazar las máquinas que se rompen durante el proceso de producción. Los resultados de este estudio proporcionarán una visión valiosa para los gerentes de producción y los responsables de la toma de decisiones en la industria, ayudándoles a maximizar la eficiencia y la rentabilidad de sus operaciones.

1 Introducción

1.1 Breve descripción del proyecto

En el mundo de la producción industrial, la eficiencia y la rentabilidad son dos factores críticos que determinan el éxito de una empresa. Este artículo se centra en un proyecto de simulación que busca optimizar el funcionamiento de una fábrica con un conjunto de máquinas en funcionamiento y máquinas de repuesto.

1.2 Objetivos y metas

El objetivo principal de este proyecto es desarrollar un modelo de simulación que permita a los gerentes de producción tomar decisiones informadas sobre si reparar o reemplazar las máquinas que se rompen durante el proceso de producción. El modelo se basa en una serie de variables, incluyendo el costo y el tiempo de reparación de las máquinas, el costo de reemplazo de las máquinas, y la ganancia generada por la fábrica en función del tiempo.

1.3 Sistema específico a simular

El sistema específico a simular es una fábrica que opera con n máquinas las cuales deben estar todas en funcionamiento para que la fábrica esté operativa. Se cuenta además con s máquinas de repuesto. Cada máquina en funcionamiento tiene un tiempo de vida determinado por una distribución F , el cual al concluir, la máquina se rompe y es sustituida por una de repuesto (asumimos que este cambio se hace inmediatamente sin interrupción).

La máquina que se averió es entonces enviada a reparación, donde tendrá un tiempo de reparación r (determinado por una distribución G_1) y un costo de reparación c (determinado por una distribución G_2), en el propio taller se toma la decisión de repararla o no, en caso de decidir hacerlo, se pone en cola para ser reparada (las máquinas se reparan siguiendo una política FIFO *first-in-first-out*) y cuando llegue su turno se cobra el costo de reparación y se esperan r unidades de tiempo que demora. En caso de decidir no reparar, entonces se compra una máquina nueva, esta acción tiene un costo fijo C y un tiempo de entrega fijo R que demora en llegar a la fábrica.

La fábrica genera dinero continuamente mientras está en funcionamiento a un ritmo $p(t) > 0$ (la fábrica no tiene pérdidas) donde t es el tiempo transcurrido y p una función que define la ganancia. Con esto se dice que la ganancia hasta el momento t^* es igual a $P(t^*) = \int_0^{t^*} p(t)dt$.

Además, al inicio tiene un presupuesto inicial $f > 0$ al cual se le suman todas las ganancias obtenidas (reversión total) en todo momento, si en algún momento $f \leq 0$, la fábrica quiebra y por lo tanto se detiene la simulación.

Para decidir si una máquina será reparada o desechada se usa una función $D(c, r, C, R, f) \in \{0, 1\}$ donde 0 representa que se repara y 1 que se desecha.

Una unidad de tiempo se considera equivalente a un día de la vida real, una unidad de dinero se considera equivalente a 1000 dólares de la vida real para tener referencias.

Teniendo este sistema, la fábrica solo se detendrá por una de dos razones:

1. No hay suficientes repuestos para sustituir las máquinas rotas, con lo que no podrán haber n en funcionamiento
2. El presupuesto en un momento dado decrece a un valor menor o igual a cero con lo que entra en bancarrota.

1.4 Variables que describen el problema

El problema es descrito por las siguientes variables:

- Distribución de los tiempos de vida de las máquinas F .
- Distribución de los tiempos de reparación de las máquinas G_1 .
- Distribución de costo de reparación de las máquinas G_2 .
- Cantidad de máquinas en funcionamiento n y de repuesto s .
- Presupuesto inicial f
- Costo de compra C y tiempo de entrega R para las máquinas nuevas.

2 Detalles de Implementación

La implementación de la simulación se llevó a cabo con Python 3.10. Se utilizaron las siguientes librerías: para el manejo eficiente de datos numéricos numpy, para la graficación de los resultados matplotlib, para las pruebas de hipótesis scipy.stats y para organizar los datos de forma conveniente pandas.

2.1 Pasos seguidos para la implementación

La implementación consta de varias partes: primeramente se tiene un módulo principal que se encarga de la simulación *per-se*, en este se utilizan las variables que se le proveen como parámetros para simular los eventos ya descritos. Este módulo consta de una única función de Python que recibe los siguientes parámetros:

1. **machines:** cantidad de máquinas en funcionamiento
2. **spares:** cantidad de máquinas de repuesto
3. **found:** fondos iniciales

4. **buy_cost:** costo de comprar una máquina nueva
5. **delivery_wait_time:** tiempo de entrega al comprar una máquina nueva
6. **profit_function:** función que determina la ganancia con respecto al tiempo
7. **decision_function:** función para decidir si comprar o reparar
8. **event_distribution:** distribución de los tiempos de vida de las máquinas
9. **repair_time_distribution:** distribución de los tiempos de reparación de las máquinas
10. **repair_cost_distribution:** distribución de los costos de reparación de las máquinas
11. **max_steps:** cantidad máxima de pasos que dará la simulación
12. **sjf:** booleano que indica si usar o no la política SJF para organizar la reparación de las máquinas

Los otros dos módulos son respectivamente uno que contiene funciones de decisión y funciones de ganancia de ejemplo para suministrarlas al módulo principal, y un módulo que se encarga de ejecutar los experimentos, este último es un JupyterNotebook.

Al ejecutar la simulación, se generan n números aleatorios con la distribución F y se colocan en una cola de prioridad. En cada paso, se extrae un número de dicha cola, el cual representa el momento en el tiempo en que se rompe una máquina, se actualizan las colas de las máquinas en reparación y pendientes a entrega, se calculan los fondos actuales y se actualiza el estado de la simulación.

A las dos condiciones de parada anteriormente mencionadas se les añade la condición de **TIMEOUT**, la cual se agrega como prevención de simular indefinidamente los eventos.

2.2 Datos Recolectados

Los datos recolectados son:

- **time:** tiempo total simulado
- **found:** fondos obtenidos
- **repaired_machines:** total de máquinas reparadas
- **bought_machines:** total de máquinas compradas
- **average_repair_time:** tiempo medio de reparar una máquina

- `spent_repair_funds`: fondos gastados en reparaciones
- `spent_buy_funds`: fondos gastados en la compra de máquinas
- `stop_reason`: razón por la que se detuvo la simulación

3 Resultados y Experimentos

Como primer experimento, utilizamos los siguientes datos: $n = 20$, $s = 5$ $C = R = 0$, los costos de reparación son siempre 0 y la función de decisión siempre decide reparar. Este experimento es una reducción al propuesto en el libro *Simulation, Fifth Edition*, by Sheldon M. Ross, sección 7.7, páginas 123-125. En ese problema se propone una simulación reducida de la expuesta en este informe, ya que no tiene en cuenta los precios de las reparaciones y las compras de máquinas.

De esta primera simulación solo podemos variar el número de máquinas de repuesto. Si aumentamos su número a $s = 10$ obtenemos que la media del tiempo aumenta considerablemente (con $s = 5$ se tiene un tiempo promedio de 330.58, con $s = 10$ se tiene un tiempo promedio de 881.37) y la proporción de TIMEOUT (condición de parada que indica que se excedió el tiempo límite establecido para la simulación) que se obtienen aumenta del 8% al 97%.

Otro posible experimento es usar SJF (estrategia que prioriza hacer primero las tareas que tardarán el menor tiempo, por sus siglas en inglés *shortest job first*) y ver como afecta la variable del tiempo. Efectivamente, al usar esta política la media del tiempo aumenta a 681.46 y la proporción de TIMEOUT's aumenta al 53%. Esta política la utilizaremos más adelante para intentar obtener la mejor configuración.

3.1 Añadir costo a las reparaciones

Para añadir costo a las reparaciones en la simulación basta con cambiar la función constante $G_2 \equiv 0$ por una distribución distinta de 0. Para tener consistencia en los experimentos y poder comparar resultados utilizamos una misma distribución para G_2 la cual es *Weibull*(2, 1).

Inicialmente utilizamos una función de decisión que siempre escoge reparar las máquinas y la comparamos con una función que escoge de forma lineal si comprar o reparar una máquina de la siguiente forma:

$$D(c, r, C, R, f) = 1 \implies c \cdot r \leq C \cdot R$$

$$D(c, r, C, R, f) = 0 \implies c \cdot r > C \cdot R$$

Aparentemente esta función majoraría los resultados en cuanto al tiempo de ejecución ya que elige de forma más inteligente la mejor opción. Sin embargo los datos arrojados por la simulación indican que en realidad tiene el efecto contrario:

- Tiempo promedio reparando siempre: 340.461
- Tiempo promedio decidiendo linealmente: 233.877

Dado que ambas medias no están muy separadas entre sí podría tratarse de un error de medición, por tanto es necesario un mayor rigor, aplicando la prueba de *Kruskal-Wallis* obtenemos un p-value de $0.0029 < \alpha = 0.05$ por lo que rechazamos la hipótesis nula de que ambas poblaciones tienen la misma media. Comparando también la media de los fondos obtenidos en ambos experimentos también podemos ver que existe una disminución de las ganancias usando la política de decisión lineal, sin ella se obtiene una ganancia media de 3038.76 y con ella se obtiene en cambio una ganancia media de 2113.03.

3.2 Utilizar una función de ganancia no lineal

Los experimentos de la sección anterior tenían en común que utilizaban una función de ganancia $p'(t^*) = c$ con $c \in \mathbb{R}$ constante, como consecuencia se simula que la fábrica obtiene unas ganancias constantes sin importar el momento, esto es muy alejado de la realidad ya que una fábrica o empresa siempre tiene periodos de mayor y menor producción, si asumimos que estos periodos son a intervalos constantes podemos modelar una función de ganancia $p(t^*) = 1 + \sin(t^*)$ la cual tiene periodos de mayor y menor ganancias incluso momentos en los que tiene ganancia 0. Luego

$$P(t^*) = \int_0^{t^*} p(t) dt = 1 - \cos(t^*)$$

es la función que le debemos suministrar a nuestra simulación para tener un escenario más "realista" en cuanto a las ganancias. Esto en principio debería disminuir las ganancias ya que $p'(t^*) \leq p(t^*)$, esta hipótesis se confirma simplemente comparando las medias de ambos experimentos (variando únicamente la función de ganancia, si se usa $p'(t^*)$ la ganancia media es de 2113.03)

4 Modelo Matemático

Los resultados muestran que el tamaño máximo y promedio de la cola varían significativamente entre las simulaciones. El tiempo promedio de espera también presenta una alta variabilidad, lo que sugiere que las condiciones específicas de cada simulación influyen en gran medida en el rendimiento. Comparar estos resultados entre simulaciones con y sin la política SJF puede proporcionar información valiosa sobre la efectividad de dicha política.

4.1 Descripción del modelo de cómo modelos probabilísticos

4.2 Supuestos y restricciones

4.3 Comparación de los resultados obtenidos con los resultados experimentales

4.4 Conclusiones