Acción de control óptima

2025-05-11

Tabla de contenidos

Introducción

La teoría de control óptimo trata con problemas de optimización de sistemas dinámicos cuyo comportamiento puede verse influenciado por la aplicación de acciones (decisiones o controles), las cuales son seleccionadas mediante reglas conocidas como estrategias o políticas de control. La eficiencia de cada una de tales políticas se mide mediante un índice de funcionamiento del sistema conocido también como criterio de optimalidad, mismo que representa, un costo o una ganancia. Entonces el problema de control óptimo consiste en encontrar una estrategia óptima tal que, según sea el caso, minimice o maximice un índice de funcionamiento apropiado.

En este trabajo presentamos una útil aplicación matemática, especificamente mediante la descripción de un sistema de inventario. Conocer el nivel de mercancia, comunmente llamado stock, resulta de especial interés, pues permite determinar una estrategia óptima de operación que nos indique la cantidad adecuada de artículos a solicitar en cada pedido, a fin de satisfacer la demanda y buscando con esto optimizar el costo que conlleva la producción, así como el costo adicional generado por el manejo del stock.

La dinámica de dicho ejemplo será modelada utilizando un Modelo de Control de Markov. Muchos trabajos de investigación han abordado ya el estudio de los sistemas de inventario utilizando distintos enfoques, por lo que actualmente existe ya mucha literatura al respecto, misma que será utilizada como referencia: Sargent y Stachurski (2024) y Gomez (2022)

Para nuestro análisis consideramos espacios de estado y acciones finitos, además de considerar una cantidad finita de transiciones para el sistema y que todos los elementos en el MCM son ya conocidos. Esto con el objetivo de realizar simulaciones computacionales e implementar algoritmos ya conocidos para encontrar la función de valor óptimo, tales como el algoritmo de la programación dinámica y el de iteración de políticas.

1 Formulación del Proceso de decisión de Markov

Considere un sistema de inventario a tiempo discreto, de un solo producto y que cuenta con una capacidad finita $M > 0$. Supongamos que se pretende maximizar ganancias através dedecidir cuánto producto solicitar a su proveedor a fin de solventar la demanda de sus clientes en cada período y dependiendo del stock (producto en existencia almacenado), cuya información se obtiene al realizar el inventario. Bajo tal escenario, para cada $t \in \{0, 1, 2, ... N\}$ podemos suponer que

- x_t : representa el nivel de inventario al inicio de cada periodo t.
- a_t : representa la cantidad de producto que se ordena al inicio del periodo t.
- ξ_t: representa la demanda del producto durante el periodo t.

Se asume que la cantidad de producto que se ordena se abastece de forma inmediata, que la demanda que no se satisface en cada periodo se pierde y que el nivel de inventario inicial es $x_0 = x \in \mathbf{X}$.

De manera que considerando un Modelo de Control de Markov

$$(\mathbf{X}, \mathbf{A}, \{A(x) : x \in \mathbf{X}\}, \mathbf{P}, \mathbf{C})$$

donde, el espacio de estados y controles son:

$$\mathbf{X} = \mathbf{A} = \{0, 1, 2, ... M\}$$

El conjunto de controles admisibles cuando el nivel de inventario es $x \in \mathbf{X}$,es

$$A(x) = \{0, 1, 2, ... , M - x\}$$

2 Dinámica del Modelo

La dinámica del sistema evoluciona en el tiempo, de modo que en cada etapa de desición $t \in \{0, 1, 2, \dots, N\}$ el nivel de inventario es $x_t = x \in \mathbf{X}$ y el controlador toma una desición admisible $a_t = a \in A(x)$, es decir, solicita cierta cantidad de articulos de acuerdo a la cantidad existente. Entonces: se produce un costo $\mathbf{C}(x, a)$; luego, el sistema evoluciona a un nuevo estado $x_{t+1} = y \in \mathbf{X}$ de acuerdo a la ley de transición $\mathbf{P}_{x,y}(a)$, la cual explicaremos a detalle en esta sección.

Consideremos $\{\xi_t\}$ una sucesión de variables aleatorias independientes e idénticamente distribuidas, definidas en el mismo espacio de probabilidad $(\Omega, \mathbf{F}, P)$; tal que toman valores en algun conjunto numerable $\mathbf{S}$ y función de probabilidad comun f conocida. Es decir, para cada $k \in \mathbf{S}$

$$f(k) = P[\xi_t = k]$$

entonces el nivel de inventario es representado en cada periodo por medio de la siguiente ecuación:

$$x_{t+1} = (x_t + a_t - \xi_t)^+ = \max(x_t + a_t - \xi_t, 0)$$

Dicha expresión puede ser interpretada del siguiente modo: la cantidad de producto en el periodo $t + 1$, será la cantidad x_t que existía hasta el periodo anterior, más la cantidad a_t solicitada de producto, menos la demanda ξ_t que se surte a los clientes. Dado que no hay acumulación de demanda, la cantidad de producto no puede ser negativa, de manera que, si no se logra surtir toda la demanda, el nivel de inventario en el siguiente periodo será cero.

Suponemos que el nivel de inventario se a venido monitoreando hata el periodo t. Entonces, la probabilidad de contar con una cantidad x de articulos, solicitar a articulos y transitar a un nivel de inventario y es:

$$\begin{aligned}
\mathbf{P}_{x,y}(a) &= P[x_{t+1} = y \mid x_t = x, a_t = a] \\
&= P[(x_t + a_t - \xi_t)^+ = y \mid x_t = x, a_t = a] \\
&= P[(x + a - \xi_t)^+ = y] \\
&= \sum_{k \in Y} f(k)
\end{aligned}$$

donde $Y = \{k \in \mathbf{S} \mid (x + a - k)^+ = y\}$

3 Ejemplo:

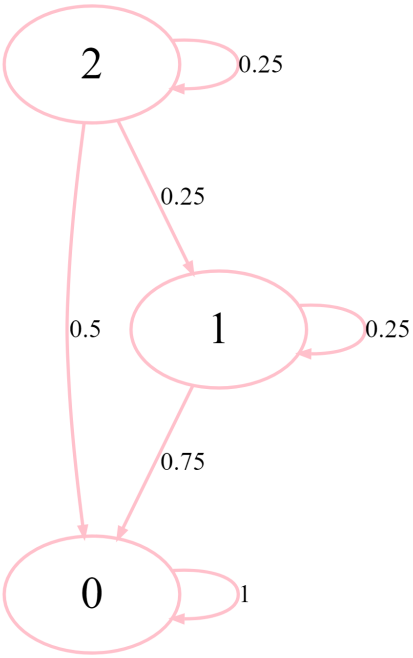
Consideremos el problema de control de inventario donde la capacidad del almacen es $M = 2$, entonces

$$\mathbf{X} = \mathbf{A} = \{0, 1, 2\}$$

por lo que, existen tres posibilidades: el almacen está vacio, contiene 1 árticulo, o contiene 2 árticulos. Además supongamos que la demanda ξ_t toma valores en el conjunto $\mathbf{S} = \{0, 1, 2, 3\}$ con distribución de probabilidad unifore en todos los periodos, es decir $P[\xi_t = k] = \frac{1}{4}$, para cada $k \in \mathbf{S}$.

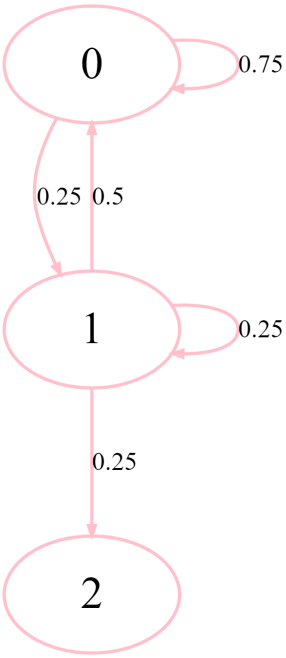
Para la acción $a = 0$ la cual es admisible para todos los estados, tenemos la siguiente tabla de transiciones:

x	a	y	$\mathbf{P}_{x,y}(a)$
contar con 0 artículos	solicitar 0 artículos	contar con 0 artículos	1
contar con 0 artículos	solicitar 0 artículos	contar con 1 artículos	0
contar con 0 artículos	solicitar 0 artículos	contar con 2 artículos	0
contar con 1 artículo	solicitar 0 artículos	contar con 0 artículos	$\frac{3}{4}$
contar con 1 artículo	solicitar 0 artículos	contar con 1 artículos	$\frac{1}{4}$
contar con 1 artículo	solicitar 0 artículos	contar con 2 artículos	0
contar con 2 artículos	solicitar 0 artículos	contar con 0 artículos	$\frac{2}{4}$
contar con 2 artículos	solicitar 0 artículos	contar con 1 artículos	$\frac{1}{4}$
contar con 2 artículos	solicitar 0 artículos	contar con 2 artículos	$\frac{1}{4}$



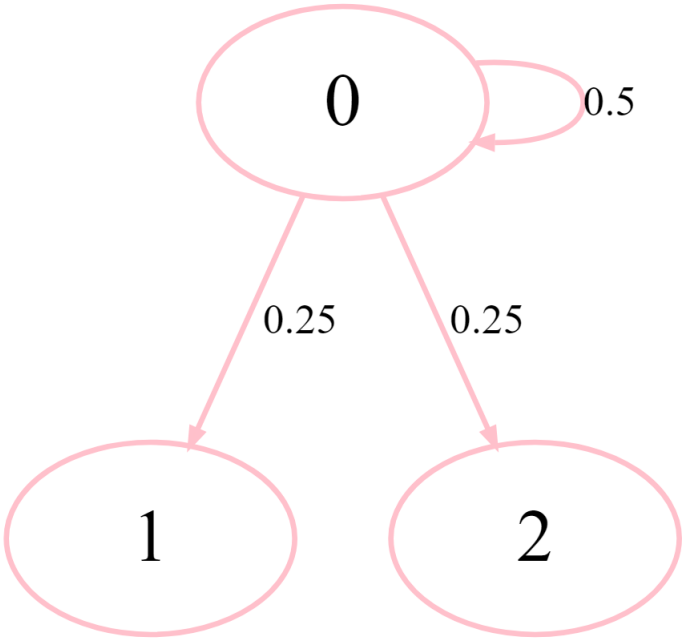
Para la acción $a = 1$ la cual es admisible cuando el nivel del almacen es 0 o 1, tenemos la siguiente tabla de transiciones:

x	a	y	$\mathbf{P}_{x,y}(a)$
contar con 0 artículos	solicitar 1 artículo	contar con 0 artículos	$\frac{3}{4}$
contar con 0 artículos	solicitar 1 artículo	contar con 1 artículos	$\frac{1}{4}$
contar con 0 artículos	solicitar 1 artículo	contar con 2 artículos	0
contar con 1 artículo	solicitar 1 artículo	contar con 0 artículos	$\frac{2}{4}$
contar con 1 artículo	solicitar 1 artículo	contar con 1 artículos	$\frac{1}{4}$
contar con 1 artículo	solicitar 1 artículo	contar con 2 artículos	$\frac{1}{4}$



Para la acción $a = 2$ la cual es admisible únicamente cuando el nivel del almacen es 0 tenemos la siguiente tabla de transiciones:

x	a	y	$\mathbf{P}_{x,y}(a)$
contar con 0 artículos	solicitar 2 artículos	contar con 0 artículos	$\frac{2}{4}$
contar con 0 artículos	solicitar 2 artículos	contar con 1 artículos	$\frac{1}{4}$
contar con 0 artículos	solicitar 2 artículos	contar con 2 artículos	$\frac{1}{4}$


0
0.5
0.25
0.25
1
2

4 Descripción y Justificación de la Función de Costo

Para la función de costo por etapa definimos las constantes

- λ : costo unitario de producción.
- h : costo unitario por almacenamiento
- p: costo unitario por demanda insatisfecha.

La función de costo por etapa para cada (x, a) está dada por

$$\mathbf{C}(x, a) = \lambda a + h \max\{0, x_t + a_t - \xi_t\} + p \max\{0, \xi_t - x_t - a_t\}$$

De manera que el costo en cada etapa es igual a la cantidad de unidades solicitadas a producción, considerando el costo de producción, mas el costo de almacenamiento, el cual dependerá del producto en existencia, y considerando que si la demanda excede la producción almacenada se genera un costo de penalización.

Si el objetivo es mejorar las ganancias a lo largo de N etapas bajo el panorama previamente descrito, entonces esto equivale a minimizar el indice de funcionamiento a continuación

$$J(\pi, x) := \mathbf{E}[\sum_{t=0}^{N-1} \mathbf{C}(x, a)]$$

5 Justificación de las Acciones

Existen modelos en los que se asume que la capacidad del inventario es infinita, lo cual pocas veces describe la realidad, es por esto que consideramos que la capacidad con la que contamos es finita, es decir que las acciones que se pueden realizar al estar a cargo de un inventario estan limitadas por la capacidad de la bodega y con la cantidad de producto existente en cada etapa.

6 Simulación del Proceso

Para faciltar los calculos se realizó el siguiente codigo en python, con el cual se genera la sucesión de acciones óptimas y los costos mínimos en cada etapa de decisión considerando cada estado del sistema

```
import numpy as np
import pandas as pd

# Parámetros
N = 4 # Períodos
CosOrd = 1.2  # Costo por ordenar
CosMan = 3  # Costo por mantener
CosEsc = 4  # Costo por escasez
p =[0.25,0.25,0.25,0.25] # Distribución de probabilidad de la demanda
MaxCap = 2  # Máxima capacidad del almacén
D = 3  # Valor máximo de la demanda

# Inicialización de la matriz de costos esperados
CostoEsperado = np.zeros((N+1 , MaxCap + 1))
DecisionOptima = np.zeros((N, MaxCap + 1), dtype=int)

# Iteración por etapas
for l in range(1, N +1):  # Etapas
    for s in range(0, MaxCap + 1):  # Estados
        estado = []
        for a in range(0, MaxCap - s + 1):  # Acciones
            esperado = 0
            for w in range(0, D + 1):  # Demanda
                esperado += p[w] * (
                    CosOrd * a
                    + CosMan * max(0, s + a - w)
                    + CosEsc * max(0, w - s - a)
                    + CostoEsperado[l - 1, max(0, s + a - w)]
                )
            estado.append(esperado)
        # Determinación del costo mínimo y la decisión óptima
        CostoEsperado[l, s] = estado[0]
        DecisionOptima[l - 1, s] = 0
        for a in range(1, len(estado)):
```

```
            if estado[a] < CostoEsperado[l, s]:
                CostoEsperado[l, s] = estado[a]
                DecisionOptima[l - 1, s] = a

# Resultados
print("Matriz de Costos Esperados:")
print(CostoEsperado)
print("Matriz de Decisiones Óptimas:")
print(DecisionOptima)
```

```
Matriz de Costos Esperados:
[[ 0.          0.          0.        ]
 [ 4.95        3.75        3.25      ]
 [ 9.6         8.4         7.475     ]
 [14.25       13.05       12.01875   ]
 [18.9        17.7        16.6421875]]
Matriz de Decisiones Óptimas:
[[1 0 0]
 [1 0 0]
 [1 0 0]
 [1 0 0]]
```

6.1 Tabla de Acción de Control Óptima

Tabla 6.1: Acción de control óptima

Épocas de decisión	**Estados**		
	0	**1**	**2**
3	1	0	0
2	1	0	0
1	1	0	0
0	1	0	0

Los resultados se resumen en las siguientes tablas

```
                Estados
-----------------------
```

Epocas de decisión	0	1	2
3			
2			

Epocas de decisión	0	1	2
1			
0			

Referencias

Gomez, Jazmín Sarahí Flores. 2022. «Algoritmo de Aproximación en Modelos de Control Semi-Markovianos y Markovianos con Costos Descontados». Universidad de Sonora.

Sargent, Thomas J, y John Stachurski. 2024. «Dynamic Programming: Finite States». *arXiv preprint arXiv:2401.10473*.