Un panorama general del aprendizaje por refuerzo en líneas de espera controladas a tiempo discreto

An overview of reinforcement learning in discrete-time controlled queues

*Gustavo Portillo-Ramírez**
Benemérita Universidad Autónoma de Puebla, México
gustavo.portillor@alumno.buap.mx
https://orcid.org/0000-0002-2457-2033

Hugo Cruz-Suárez
Benemérita Universidad Autónoma de Puebla, México
hcs@fcfm.buap.mx
https://orcid.org/0000-0002-0732-4943

Fernando Velasco-Luna
Benemérita Universidad Autónoma de Puebla, México
fvelasco@fcfm.buap.mx
https://orcid.org/0000-0002-8616-8378

Recepción: 20 de marzo de 2024
Aprobación: 6 de agosto de 2024

Resumen
Se revisa un panorama general de procesos de decisión de Markov a través de un modelo de línea de espera controlada. El modelo utilizado se trabaja bajo un criterio de optimalidad promedio para el cual se muestra la existencia de políticas estacionarias. Posteriormente, vía programación dinámica y Q-learning se determina el costo óptimo y la política óptima. Para finalizar, se proporcionan algunos experimentos numéricos que validan los resultados obtenidos por programación dinámica y Q-learning. Además, se realiza una comparación de las soluciones que producen estas dos técnicas.

Palabras clave: procesos de decisión de Markov, teoría de líneas de espera, Q-learning, costo promedio, políticas estacionarias.

Abstract
An overview of Markov decision processes is reviewed using a controlled queueing model. The model used is treated under an average optimality criterion for which the existence of stationary strategies is shown. Subsequently, dynamic programming and Q-learning are used to determine the optimal cost and the optimal policy. Finally, some numerical experiments are presented to validate the results obtained by dynamic programming and Q-learning. Furthermore, the solutions obtained with these two techniques are compared.

Keywords: Markov decision processes, queueing theory, Q-learning, average cost, stationary policies.

Introducción

Las líneas de espera o colas están presentes en la vida diaria. Se encuentran en bancos, supermercados, hospitales, en el tránsito vehicular o cuando se realiza algún trámite administrativo. Es fundamental que un sistema de colas trabaje de forma óptima, ya que, de acuerdo con Hillier (2005), el tiempo que la población de un país pierde al esperar en las colas es un factor determinante tanto de la calidad de vida como de la eficiencia de su economía. En general, los clientes o consumidores de un sistema de líneas de espera no necesariamente están compuestos por personas. Existen situaciones donde, en lugar de clientes, los elementos de las colas son bienes perecederos, máquinas, datos, bits, etc. De este modo, las líneas de espera se extienden a diferentes ámbitos científicos y áreas de aplicación: ingeniería (Hamilton *et al*., 2018), telecomunicaciones (Afolalu *et*

al., 2021), inteligencia artificial (Lovas y Rásonyi, 2021), medicina (Fomundam y Herrmann, 2007), finanzas (Lari Dashtbayaz *et al*., 2019). En este trabajo, la atención se centra en mostrar un panorama general de líneas de espera. Nos interesa, en específico, identificar las componentes del problema de líneas de espera con un modelo de decisión de Markov. El modelo de decisión utilizado tiene asociado un criterio de recompensa promedio, pues, como bien menciona Cao (2021), el criterio de recompensa promedio ocupa una piedra angular de la teoría de control de colas, especialmente cuando se aplica al control de sistemas informáticos y redes de comunicación. Los modelos de decisión se ocupan de situaciones en donde se deben tomar decisiones en etapas múltiples y sucesivas. En cada etapa, dependiendo del estatus actual de la situación y de la decisión tomada, se obtiene de manera inmediata un costo o una recompensa. Para la identificación requerida, se revisa de forma general la teoría de procesos de decisión de Markov (PDM) y las técnicas que resuelven problemas de control de este tipo, a saber, programación dinámica (PD) y aprendizaje por refuerzo (AR). Una de las técnicas que se ha propuesto para trabajar con modelos de decisión secuencial es la programación dinámica. En la literatura, la PD se considera una herramienta poderosa para encontrar tanto una estrategia óptima como el costo óptimo bajo algún criterio de rendimiento. El modelo teórico de un sistema está compuesto por la matriz de transiciones, matriz de costos/recompensas y el tiempo. Es frecuente que la evaluación de las probabilidades de transición involucre la evaluación de múltiples integrales que contengan funciones de densidad de probabilidad o funciones de masa de probabilidad de las variables aleatorias presentes en el modelo. Sin embargo, cuando se tiene un sistema con muchas variables aleatorias, la tarea anterior se convierte en un trabajo extremadamente desafiante. Este problema se conoce como la *maldición del modelaje de PD*. Por otro lado, cuando se trabaja con sistemas estocásticos grandes aparece la llamada *maldición de la dimensionalidad de PD*. Por lo anterior, se dice que la PD sufre de una doble maldición: la maldición de la dimensionalidad y la maldición del modelaje (Gosavi, 2015). Para combatir estas problemáticas, se ha implementado la metodología de AR, la cual se presenta como un tipo de PD basada en simulación. Una característica interesante de AR es que no se necesita la matriz de transición de probabilidad; en su lugar, se requiere solo de la distribución de las variables aleatorias que gobiernan el comportamiento del sistema estocástico. Después de proporcionar una visión general de programación dinámica y aprendizaje por refuerzo, se utiliza un sistema estocástico a tiempo discreto, el cual cuenta con un mecanismo de transición controlado que describe el modelo de líneas de espera con el objetivo de implementar los algoritmos de los procedimientos expuestos en Sennott (2009) para PD y en Gosavi (2015) para AR. Los algoritmos se desarrollaron en el *software* R (R Project, 2022). Con este enfoque, se deduce una estrategia de operación estacionaria de tipo umbral (Kitaev y Rykov, 1995). El artículo está organizado del modo siguiente: la sección 1 proporciona una visión general de modelos de líneas de espera y se presenta el modelo con el que se trabaja a lo largo del artículo. En la sección 2 se revisa de manera general la teoría de procesos de decisión de Markov y se aborda el modelo de líneas de espera propuesto en la sección 1. En la sección 3 se utiliza PD para dar solución al modelo de líneas de espera a través de PDM. Después, en la sección 4 se revisa la teoría de aprendizaje por refuerzo con el objetivo de obtener una versión numérica del modelo de líneas de espera de la sección 1. Finalmente, en la sección 5 se muestran ejemplos numéricos de la teoría de líneas de espera usando el enfoque de programación dinámica y aprendizaje por refuerzo.

1. Modelo de líneas de espera

Un sistema de colas incluye servidores, clientes y, usualmente, líneas de espera, las cuales se producen cuando la demanda de un servicio supera la capacidad de proporcionar dicho servicio. En la vida diaria nos encontramos con una variedad importante de sistemas de colas. Sin embargo, existen situaciones en las que los servidores son computadoras, unidades de mantenimiento vehicular, líneas de comunicación, servicio técnico, hospitales, estaciones o líneas de producción, mientras que los clientes son mensajes, bloques de

bits u objetos en un proceso de servicio o de manufactura. La característica constante en cada uno de estos ejemplos es que no se sabe con seguridad el tiempo de llegada de los clientes ni el tiempo de servicio. Por otro lado, en un sistema de colas existen dos escenarios que se desean evitar. El primero, cuando un cliente sale de la fila a causa de una fila excesiva, debido a que los servidores proveen el servicio con una tasa de atención baja. El segundo, cuando la atención del servicio es eficiente, pero se generan costos excesivos a causa de la tasa de atención tan alta. Para evitar este tipo de escenarios, se necesita que un sistema de colas opere de manera óptima; el enfoque de teoría de control ofrece una alternativa para este tipo de operación. En algunos contextos se ha documentado que, aunque los servidores trabajen a una tasa de atención baja, los clientes están dispuestos a esperar por la necesidad acceder al servicio; sin embargo, su deseo es conocer el tiempo de espera. Por lo tanto, es claro que existen múltiples factores de interés en una cola. En este marco, se tienen algunas de las medidas de desempeño de las colas, por ejemplo, longitud de la cola, probabilidad de pérdida, tiempos de espera, tiempo del sistema, carga de trabajo, proceso de antigüedad, periodo ocupado, periodo de inactividad y tiempos de salida (Alfa, 2016). Algunos de los parámetros más utilizados para caracterizar un sistema de colas incluyen los procesos de llegada y servicio, capacidad del sistema, la cual puede ser finita o infinita, y disciplina de cola como FIFO (primero en llegar, primero en salir), LIFO (último en entrar, primero en salir), SIRO (servicio en orden aleatorio) y PRIO (prioridad). El estudio de un sistema de colas ocurre de manera natural a tiempo discreto y a tiempo continuo (Puterman, 2005). En este trabajo será usado un modelo a tiempo discreto. Para abordar modelos de colas en tiempo continuo, por ejemplo, en modelos semimarkovianos, se sugiere consultar Kitaev y Rykov (1995).

Una de las características para obtener fórmulas cerradas para las medidas de desarrollo de un sistema de líneas de espera es bajo el supuesto de ergodicidad o estabilidad en el número de clientes/consumidores en el sistema. Tal supuesto tiene como base el hecho de que el sistema determinista asociado, cadena de nacimiento y muerte, se encuentra compuesto por estados del tipo recurrente positivo. La afirmación anterior nos dice que a corto plazo dichas expresiones se encuentran muy lejanas del valor de la medida en cuestión. Por lo tanto, la aproximación mejora cuando el tiempo avanza. Además, en el trabajo de Sharma y Gupta (1982) se encuentran fórmulas explícitas, pero complejas para el caso transitorio.

En el contexto del párrafo anterior, en líneas de espera de redes de comunicación el tiempo de operación en un estado estable se considera en intervalos de tiempo que son largos en comparación con las constantes de tiempo del sistema (Arapostathis *et al*., 1993). Por lo tanto, los costos/recompensas generados en el proceso de funcionamiento del sistema se miden a través de un criterio promedio. En este trabajo se propone ilustrar un ejemplo numérico de un sistema de líneas de espera que puede operar bajo distintas tasas de servicio, en función del número de clientes actuales en el sistema. Cuando el sistema opera con tasas altas, se generan costos mayores, lo cual se traduce en que el costo de operación es una función creciente en función de la tasa de operación. El objetivo planteado es determinar una estrategia óptima de operación que minimice el costo promedio. El siguiente modelo de líneas de espera se plantea por el problema de líneas de espera, expuesto en Sennott (2009), para un modelo de capacidad dos. En la sección 5 se proporcionan dos procedimientos numéricos para optimizar el funcionamiento de esta línea de espera.

Considere un modelo de línea de espera con capacidad K, donde K es un entero positivo (finito) conocido. El sistema está compuesto por un único servidor y puede trabajar con $N \geq 2$, niveles (tasas) de intensidad: $0 < a_1 < a_2 < a_3 < \cdots < a_N < 1$. Estos niveles son intercambiados en función del número de clientes en el sistema. Debe tomarse en cuenta que operar a una tasa a, genera un costo $L(a)$, para $a \in \{a_1, a_2, ..., a_N\}$ donde $0 < L(a_1) < L(a_2) < L(a_3) < \cdots < L(a_N)$. Adicionalmente, el sistema genera un costo de atención en función del número de clientes dado por $M(i)$, $i = 0, 1, ...,$ y un costo de servicio en función de la tasa de servicio $L(a_i)$, $i \in \{1, 2\leq, ..., N\}$. Se supone que existe un valor $p \in (0, 1)$, el cual representa la probabilidad de que un cliente ingrese al sistema. El problema consiste en determinar una estrategia de operación en función del número de clientes y de la elección adecuada de la tasa de servicio que minimice el costo promedio a largo plazo.

2. Enfoque vía procesos de decisión de Markov

Los modelos de decisión secuencial son abstracciones matemáticas de situaciones en las que se deben tomar decisiones en distintas etapas secuenciales. Cada decisión puede influir en las circunstancias bajo las cuales se tomarán decisiones en el futuro de modo que, si se quiere minimizar un costo total, se debe equilibrar el deseo de minimizar el costo de la decisión con el anhelo de evitar situaciones futuras donde el alto costo es inevitable. Entre la variedad de problemas de decisión secuenciales, existen problemas de control óptimo, problemas de decisión de Markov y semi-Markov, problemas de control minimax y juegos secuenciales (Bertsekas, 1995). En este trabajo nos enfocamos en los PDM para abordar el modelo de línea de espera. Los PDM proveen una herramienta poderosa para dar solución a problemas de la vida real. Por lo anterior, muchos investigadores han sido atraídos por estos modelos y, como consecuencia, los han aplicado en una amplia variedad de disciplinas: ingeniería, informática, comunicaciones, economía, entre otros (Boucherie y Van Dijk, 2017; Hernández-Hernández y Minjárez-Sosa, 2012). La teoría de procesos de decisión de Markov presentada es referente a optimización secuencial de sistemas estocásticos a tiempo discreto. Cada política de control define el proceso estocástico y los valores de las funciones objetivo asociadas a este proceso. El objetivo es elegir una *buena* política de control.

En general, los PDM se definen a través de los siguientes objetos: un espacio de estados X; conjunto de acciones A; una familia $\{A(x)|\, x \in X\}$ de subconjuntos medibles $A(x)$ de A, donde $A(x)$ denota el conjunto de acciones admisibles cuando el sistema se encuentra en el estado $x \in X$; probabilidades de transición controladas, denotadas por $p_{x,y}(a), x, y \in X, a \in A(x)$ una función de costo $C: X \times A \to \mathrm{R}$, $C(x, a)$ denota el costo en un paso cuando se aplica la acción a en el estado x, $x \in X$ y $a \in A(x)$. La interpretación es la siguiente: cuando el sistema se encuentra en el estado x un controlador selecciona una acción $a \in A(x)$. Después, el sistema se mueve al siguiente estado, digamos y, de acuerdo con la distribución de probabilidad $p_{x,y}(a)$. Como consecuencia, se incurre en un costo $C(x, a)$. Ahora el estado actual es y, posteriormente el proceso se repite.

Un PDM es llamado *finito* si el espacio de estados X y el conjunto de acciones A son finitos. Adicionalmente, decimos que un conjunto es discreto si este es finito o numerable. De esta manera, un PDM es llamado *discreto* si el espacio de estados y el conjunto de acciones son conjuntos discretos. Un elemento indispensable de los PDM es el criterio de rendimiento, es decir, la función objetivo. El criterio de rendimiento es crucial para trabajar con PDM ya que es la herramienta básica para comparar políticas. En lo que sigue, se describen de manera breve tres tipos de criterios de rendimiento: costo total, costo descontado y costo promedio. Sean T_c una función de costo terminal y $\beta \in [0,1]$ el factor de descuento (Feinberg y Shwartz, 2012; Puterman, 2005). Denote por $V_N(x, \pi, \beta, T_c)$, el costo total esperado sobre las primeras N etapas, $N \in \mathbb{N}$, esto es:

$$V_N(x, \pi, \beta, T_c) = E_x^{\pi} \left[\sum_{n=0}^{N-1} \beta^n \mathrm{C}(x_n, a_n) + \beta^N T_c(x_N)\right], \tag{1}$$

donde E_x^{π} denota la esperanza condicional dado el estado $x \in X$ y la política $\pi \in \Pi$, Π denota el conjunto de todas las políticas. Si $\beta \in [0,1)$, nos encontramos con el costo descontado total esperado. Ahora, si $\beta = 1$, obtenemos el costo total. Observe que la expresión (1) produce un problema con horizonte finito N si es fijo y $N < \infty$. En cambio, para obtener un problema de horizonte infinito basta configurar el costo terminal como 0 y sustituir ∞ en lugar de N. Por lo anterior, el costo total esperado sobre un horizonte infinito se define como $V(x, \pi, \beta) = V_{\infty}(x, \pi, \beta, 0)$. Si la función de costo C es acotada, el costo total esperado sobre el horizonte infinito está bien definido cuando $\beta \in [0,1)$. Para el caso $\beta = 1$, se necesitan condiciones extra para que el costo total esperado $V(x, \pi, 1)$ esté bien definido (Puterman, 2005). Note que en un problema con horizonte infinito, como consecuencia de que $\beta = 1$, la principal dificultad es que la suma que aparece en $V(x, \pi, 1)$ podría no ser convergente. Por lo anterior, es natural considerar el costo esperado por unidad de tiempo:

$$J(x, \pi) = limsup_{N \to \infty} \frac{1}{N} V_N(x, \pi, 1, 0). \tag{2}$$

Observación: en algunas otras referencias se considera el límite inferior en (2), por ejemplo Puterman (2005), ya que el límite en general no existe para toda política $\pi \in \Pi$.

Observe que si $U(x, \pi)$ es un criterio de rendimiento definido para cada política $\pi \in \Pi$, entonces denote

$$U(x) = \inf_{\pi \in \Pi} U(x, \pi). \tag{3}$$

Adicionalmente, si existe $\pi^* \in \Pi$ tal que $U(x, \pi^*) = \inf_{\pi \in \Pi} U(x, \pi)$ a π^* se le *llama política óptima*. En términos de los criterios de rendimiento definidos, de la expresión (3) se obtiene que

$$V_N(x, \beta, T_c) = inf_{\pi \in \Pi} V_N(x, \pi, \beta, T_c),\ V(x, \beta) = inf_{\pi \in \Pi} V(x, \pi, \beta),\ J(x) = inf_{\pi \in \Pi} J(x, \pi). \tag{4}$$

Las expresiones que aparecen en (4), se conocen como función de valor óptimo para el caso total, descontado y promedio, respectivamente.

Observación: en otras referencias se considera función de recompensa, en lugar de función de costo (Feinberg y Shwartz, 2012; Puterman, 2005). En consecuencia, el nuevo problema consiste en maximizar la recompensa total, por lo que se debe cambiar el ínfimo que aparece en las expresiones (3) y (4) por un supremo.

En este trabajo nos enfocamos en el costo promedio a largo plazo, que por (2) resulta estar definido como

$$J(\pi, x) = limsup_{N \to \infty} \frac{1}{N} E_x^{\pi} \left[\sum_{t=0}^{N-1} C(x_t, a_t) \right], \tag{5}$$

$\pi \in \Pi$, $x \in \mathrm{X}$. La construcción adecuada del espacio de probabilidad es vía el teorema de la medida producto. Para detalles de la construcción, véase Puterman (2005). En este punto, se tienen las condiciones para definir el problema de optimización (o control) de interés; una política π^* se denomina *óptima promedio* si $J(\pi^*, x) = inf_{\pi \in \Pi} J(\pi, x)$, y $J(x) := J(\pi^*, x)$ se denomina *costo promedio óptimo*.

Las componentes del modelo descrito en la sección 1 pueden identificarse en un modelo de control de Markov de la manera siguiente. Considere el espacio de estados $X = \{0, 1, 2, \ldots, N\}$, con N finito, el espacio de acciones $A = \{a_1, a_2, \ldots, a_N\}$; las acciones admisibles coinciden con A; la ley de transición se describe como sigue:

$$\begin{aligned} p_{00}(a) &= 1 - p + ap,\ p_{01}(a) = (1 - a)p,\ p_{N, N-1}(a) = a(1 - p),\ p_{NN}(a) = 1 - a(1 - p),\ p_{i, i-1}(a) \\ &= a(1 - p),\ p_{ii}(a) = ap + (1 - a)(1 - p),\ p_{i, i+1}(a) = (1 - a)p. \end{aligned} \tag{6}$$

Para $i \in \{1, 2, \ldots, N - 1\}$ y $a \in A$, donde $p_{ij}(a) = Prob(X_{t+1} = j | X_t = i, a_t = a)$, para $i, j \in X$ y $a \in A$. La interpretación de las componentes de la ley de transición es la siguiente: $p_{00}(a)$ representa la probabilidad de que el sistema permanezca en el estado 0; esto ocurre cuando no hay arribo de clientes $(1 - p)$ o bien hay un arribo y se completa un servicio (ap); $p_{i, i-1}$ corresponde a la probabilidad de que el sistema inicialmente tenga i clientes y en la siguiente etapa tenga $i - 1$. Este escenario ocurre cuando no hay arribos $(1 - p)$ y se atiende a un cliente a una tasa a. De manera similar se interpretan las probabilidades de transición restantes.

Por último, el costo en un paso está dado por $C(i, a) = M(i) + L(a)$ donde $i \in X$ y $a \in A$. Por lo tanto, se obtiene un modelo de control de Markov $(X, A, \{A(i): i \in X\}, P, C)$, donde $P = \left[p_{ij}(a)\right]$. La dinámica del proceso se describe como sigue: en cada etapa de tiempo $t = 0, 1, \ldots$; se observa el sistema y se tienen $x_t = x$ clientes en el sistema. Entonces, se selecciona una tasa de servicio $a_t = a \in A$ y ocurre lo siguiente: el número de clientes en el sistema cambia a un estado $y \in X$ de acuerdo con la ley de transición $p_{xy}(a) = Prob(x_{t+1} = y | x_t = x, a_t = a)$. Después, se contabiliza el costo generado por el número de clientes x y la acción aplicada a, el cual está dado por $C(x, a)$.

En las siguientes secciones, se abordará este problema de control vía las técnicas de PD y AR. El análisis se presenta bajo el enfoque de espacio de estados finito. El objetivo es determinar una estrategia de operación para el sistema de líneas de espera. Tales estrategias son elegidas en función de la información que se cuenta en cada instante de tiempo. En consecuencia, se requiere de la historia generada por el proceso, la cual se define a continuación. Para cada $t \geq 0$, se define el espacio H_t, denominado *el espacio de historias* hasta el tiempo t, como $H_0 = X$ y $H_t = K \times X$ si $t \geq 1$, donde $K = \{(x, a): x \in X, a \in A\}$. Una política es una sucesión $\pi = \{\pi_t\}$ de *kernels* estocásticos, donde cada π_t está definido sobre A dado H_t y satisface que $\pi_t (A(x_t)|h_t) = 1$ para cada $h_t \in H_t$ con $t \geq 0$. Además, se denota a la familia de probabilidades condicionales sobre A dado X, como $P(A|X)$. Sea Φ el conjunto de todas las probabilidades condicionales ϕ en $P(A|X)$ tal que para todo $x \in X$ se tiene que $\phi(A(x)|x) = 1$. Con base en esta propuesta, se distinguen tres tipos de política $\pi = \{\pi_t\}$: *Markoviana Aleatorizada*, si existe una sucesión $\{\phi_t\} \subseteq \Phi$ tal que $\pi_t (\cdot|h_t) = \phi(\cdot|x_t)$, para todo $h_t \in \mathcal{H}_t$ y $t \geq 0$. Esta clase de políticas se denota por Π_{RM}. *Determinista*, si existe una sucesión $\{g_t\}$ de funciones medibles g_t: $\mathcal{H}_t \rightarrow A$, tales que para cada $h_t \in \mathcal{H}_t$ y $t \geq 0$ se tiene que $g_t (h_t) \in A(x_t)$ y $\pi_t (\cdot|h_t)$ está concentrada en $g_t (h_t)$. Esta clase de políticas será denotada por Π_D. Finalmente, determinista Markov-estacionaria si existe una función medible f: $X \rightarrow A$, tal que $f(x_t) \in A(x_t)$ y $\pi_t (\cdot|h_t)$ está concentrada en $f(x_t)$ para cada $h_t \in \mathcal{H}_t$ y $t \geq 0$. El conjunto de políticas estacionarias se denota por F.

3. Enfoque de programación dinámica: caso finito

Como uno de los métodos principales para el análisis de problemas de decisión secuencial, se tiene la PD. Esta metodología se popularizó como consecuencia de los estudios sistemáticos que Bellman realizó a principios de los años cincuenta. Bellman identificó los componentes comunes en los problemas que se tenían hasta el momento y mediante su trabajo sobre ecuaciones funcionales, programación dinámica y el principio de optimización se convirtió en líder en el campo. Él plasmó en Bellman (1957) numerosas referencias a su propia investigación y otras investigaciones tempranas; de esta manera, logró demostrar el amplio alcance de la PD. Posteriormente, se investigaron de forma extensa los problemas matemáticos y algorítmicos asociados con los problemas de horizonte finito y se formularon y analizaron las soluciones a los problemas de tiempo continuo. Durante los primeros años de la teoría de procesos de decisión de Markov gran parte de la investigación se centró solo en el estudio de ecuaciones de optimización y de los métodos para su solución. Esta solución incluye la iteración de políticas y valores. Además, PD se usó en una variedad muy amplia de aplicaciones: diversas ramas de la ingeniería, estadística, finanzas, economía y en algunas ciencias sociales. El costo correspondiente a una política y la iteración básica del algoritmo de PD pueden describirse mediante un determinado mapeo que difiere de un problema a otro en detalles que en gran medida no son esenciales. Normalmente, tal mapeo resume los datos del problema y determina las cantidades de interés para el analista. Si se toma este mapeo como punto de partida, se pueden proporcionar resultados analíticos poderosos que son aplicables a una gran colección de problemas de decisión secuencial. Por lo general, PD se aplica a problemas de horizonte infinito de criterio único con un costo/recompensa total esperado o un costo/recompensa promedio por unidad de tiempo. Para introducir los tres criterios de rendimiento descritos en la sección 2, en Ross (1970) es posible consultar algunos de problemas de control clásicos que utilizan espacios de estados finito. En esta sección se presenta la teoría general de PDM con espacio de estados finito para el criterio de costo promedio. Considere el modelo de control de Markov: M: = $(X, A, \{A(x): x \in X\}, P, C)$, bajo el criterio promedio que aparece en (5). El problema de control óptimo consiste en determinar una política $\pi^* \in \Pi$, tal que $J(\pi^*, x) = inf_{\pi \in \Pi} J(\pi, x)$, para cada $x \in X$. Entonces, se define el costo promedio mínimo (función de valor promedio) para cada como:

$$g(x) = inf_{\pi \in \Pi} J(\pi, x). \tag{7}$$

El siguiente teorema establece la caracterización del costo promedio mínimo, en el cual resulta constante digamos g^*, y muestra la existencia de políticas estacionarias (Sennott, 2009).

Teorema 1. Considere el modelo M con espacio de estados finito. Suponga que existe una constante g^* y una función V:X $\rightarrow \mathbb{R}$ tales que:

$$g^* + V(x) = \min_a \{C(x, a) + \sum_j p_{xj}(a)V(j)\}. \quad (8)$$

Entonces la función de valor promedio (7) es igual a g^*, y cualquier política estacionaria que minimice el lado derecho de la ecuación (8) es la política óptima, denotada por f^*.

Con la finalidad de proponer un algoritmo de aproximación para determinar una política óptima es necesario considerar el siguiente supuesto, referente a la ergodicidad de la cadena de Markov inducida por la estrategia óptima.

Condición 1. Cada clase recurrente positiva en la cadena de Markov inducida por la política estacionaria óptima f^* es aperiódica.

Bajo la Condición 1, se tiene el siguiente algoritmo iterativo (AI) para aproximar una política óptima y determinar el costo promedio óptimo.

Iniciar con $n = 0, \epsilon > 0, u_0 := 0$ e $i \in S$ un estado fijo denominado estado distinguido.

Definir $w_n(x) := min_a \{C(x, a) + E_x [u_n (X_1)]\}$.

Si $n = 0$, *defina* $\delta := 1$. Si $n \geq 1$, entonces proponga $\delta := max_{x \in S}|w_n(x) - w_{n-1}(x)|$. Si $\delta < \epsilon$, proceder al paso 6.

Hacer $u_{n+1} := w_n(x) - w_n(i)$.

Ir al paso 2 y reemplazar n por $n + 1$.

Imprimir $w_n(i)$ y una política estacionaria que resuelva el problema de optimización: $min_a\{C(x, a) + E_x[u_n(X_1)]\}$.

En la sección 5 se presenta un ejemplo numérico donde se implementó el AI y las soluciones se comparan con las obtenidas por el algoritmo Q-learning de aprendizaje por refuerzo.

4. Aprendizaje por refuerzo

Una solución para evitar la doble maldición de la programación dinámica es la metodología de aprendizaje por refuerzo. Esto es posible, ya que el esfuerzo de modelado en AR es menor que el de PD. Una cuestión notable es que en PDM a gran escala, el enfoque de PD ya no es factible y AR se convierte en una alternativa (Gosavi, 2015). Además, para resolver problemas cuyas probabilidades de transición son difíciles de estimar, AR es un enfoque atractivo que produce soluciones casi óptimas. La principal herramienta empleada por AR es la simulación, la cual se utiliza para evitar el cálculo de probabilidades de transición. Sin embargo, cabe mencionar que en general si se tiene acceso a las matrices de costos/recompensas y probabilidades de transiciones, se debe usar PD porque garantizaría la generación de soluciones óptimas y AR no necesariamente. Gosavi (2015) establece el AR como una rama de PD, por lo que el algoritmo de AR se deriva de sus contrapartes de PD. En la metodología de programación dinámica el primer paso es para generar la matriz de probabilidad de transición y la matriz de transición de costos/recompensas y el siguiente paso es usar estas matrices en un algoritmo adecuado para generar la solución óptima. En aprendizaje por refuerzo no se estima ninguna matriz, sino que se simula el sistema utilizando las distribuciones de las variables aleatorias gobernantes. Después, dentro del simulador, se utiliza un algoritmo adecuado para obtener la solución. A continuación, se discutirán algunos de los conceptos relacionados con AR: Q-factores, el algoritmo de Robbins-Monro, tamaños de paso y cómo se combinan estas ideas para resolver PDM dentro de los simuladores. Aunque el escrito se enfoca en el estudio de aprendizaje por refuerzo para colas en tiempo discreto, también se puede usar cuando se trabaja con colas a tiempo continuo (Abdulla y Bhatnagar, 2007).

4.1. Q-factor

En AR la función de valor se almacena en los denominados Q-factores. En este punto vale la pena señalar que la función de valor de costo promedio puede definirse por la ecuación de optimización de Bellman:

$$J^*(i) = min_{a\in A(i)} \left\{\sum_{j=1}^{n} p_{ij}(a)\left[C(i,a,j) + J^*(j)\right] - g^*\right\}, \tag{9}$$

para $i \in X$, donde $n = |X|$ es el número de estados de la cadena de Markov; $J^*(i)$ denota el -ésimo elemento del vector de la función de valor; $A(i)$ es el conjunto de acciones admisibles en el estado i; $p_{ij}(a)$; denota la probabilidad de transición en un paso de ir del estado i al estado j bajo la influencia de la acción a; $C(i, a, j)$ denota el costo inmediato en el estado i cuando la acción a es seleccionada y como resultado el sistema transita al estado j; g^* denota el costo promedio óptimo (Teorema 1). En PD, dado un estado $i \in X$, asociamos un solo elemento de la función de valor: $J^*(i)$. En cambio, en el enfoque de AR se utiliza una pareja de estado-acción para asociarlo a un vector llamado *Q-factor*. Para una pareja de estado-acción $(i, a) \in K$ se define el Q-factor como:

$$Q(i,a) = \sum_{j=1}^{n} p_{ij}(a)\left[C(i,a,j) + J^*(j)\right]. \tag{10}$$

Observe que si se combinan (9) y (10), se obtiene que

$$J^*(i) = min_{a\in A(i)} Q(i,a) - g^*. \tag{11}$$

La expresión en (11) dice que si se conocen los valores de los Q-factores, es posible obtener la función de valor J^*. Usando la igualdad (11), la ecuación (10), se describe como

$$Q(i,a) = \sum_{j=1}^{n} p_{ij}(a)\left[C(i,a,j) + min_{a\in A(i)} Q(i,a) - g^*\right], \tag{12}$$

para todo $(i, a) \in K$. La ecuación (12) puede interpretarse como la versión Q-factor de la ecuación de optimalidad de Bellman para el costo promedio de PDM, véase (9). Note que por (12), el algoritmo anterior es equivalente al algoritmo de iteración de valor regular. Observe que los Q-factores deben actualizarse en cada etapa del tiempo, por lo que superindexamos a Q con el número de iteración correspondiente. En PD, la actualización de Q se realiza de la siguiente manera: $Q^{k+1}(i,a) \leftarrow \sum_{j=1}^{n} p_{ij}(a)\left[C(i,a,j) + min_{b\in A(j)} Q^k(j,b)\right] - g^*$ para $K = 1,\ldots, k_{max}$, donde k_{max} es el número máximo de iteraciones. Cuando se usa AR, se realiza un cambio en la regla de actualización de Q que propone PD. Este cambio se logra aplicando algoritmo de Robbins-Monro.

4. 2. Q-factores y Robbins-Monro

El algoritmo de Robbins-Monro (Gosavi, 2015) es un algoritmo popular y ampliamente usado que ayuda a estimar la media de una variable aleatoria a partir de muestras. Denote la suma de las primeras k valores de X por $S_k := \sum_{i=1}^{k} x_i/k$, $k = 1, 2,\ldots$ Con esta notación, se puede escribir $S_{k+1} = (1 - \frac{1}{k})\, S_k + \frac{x_k + 1}{k + 1}$. Si denota $\alpha_k = \frac{1}{k}$, para $k \geq 1$, entonces la expresión anterior se escribe como $S_{k+1} = (1 - \alpha_{k+1})\, S_k + \alpha_{k+1}\, x_{k+1}$, la cual se denomina *algoritmo de Robbins-Monro* y α es llamada *tamaño de paso o tasa de aprendizaje*. En la literatura se han estudiado distintas reglas que son muy usadas para α: $\frac{1}{k}$, $\frac{A}{k + B}$, $\frac{\log(k)}{k}$ y $\frac{A}{V(i,a)}$, donde $k = 1, 2,\ldots,$ A y B son constantes y $V(i, a)$ denota el número de visitas a la pareja estado-acción $(i, a) \in K$ (Gosavi, 2008;

Gosavi 2015). En este trabajo la implementación numérica se desarrolló con la tasa de aprendizaje $\frac{A}{k+B}$, $k = 1, 2,\ldots$, con A y B constantes. Usaremos el algoritmo de Robbins-Monro para actualizar los Q-factores dentro de los simuladores. Se puede demostrar que cada Q-factor se puede expresar como un promedio de una variable aleatoria. Observe que de la expresión (12) se consigue que $Q(i, a) = \sum_{j=1}^{n} p_{ij}(a)\left[C(i, a, j) - g^* + \min_{b\in A(j)} Q(j, b)\right] = E\left[C(i, a, j) - g^* + \min_{b\in A(j)} Q(j, b)\right]$. Por lo tanto, si se generan muestras de la variable aleatoria que aparece entre corchetes en la expresión anterior, es posible usar el esquema que proporciona el algoritmo de Robbins-Monro para actualizar los Q-factores. Utilizando este algoritmo, se tiene que

$$Q^{k+1}(i, a) \leftarrow (1 - \alpha_{k+1})\, Q_k(i, a) + \alpha_{k+1}\left[C(i, a, j) - g^* + \min_{b\in A(j)} Q^k(j, b)\right], \tag{13}$$

para cada $(i, a) \in K$. Dado que el valor de g^* no se conoce antes de resolver el problema, de la misma forma que en PD, se prefiere usar iteración de valor relativo. Para resolver el problema que presenta la ecuación de actualización (13), se elige una pareja de estado-acción arbitraria denotada por (i^*, a^*), llamada *pareja de estado-acción distinguida*. Entonces, la actualización de Q-factor, se lleva a cabo cambiando g^* por $Q(i^*, a^*)$ en (13):

$$Q^{k+1}(i, a) \leftarrow (1 - \alpha)\, Q^k(i, a) + \alpha\left[C(i, a, j) - Q(i^*, a^*) + \min_{b\in A(j)} Q^k(j, b)\right]. \tag{14}$$

El algoritmo resultante es llamado *Q-learnig relativo* (Gosavi, 2015). Además, se ha demostrado que $Q(i^*, a^*)$ convergerá a g^* (Abounadi *et al*., 2001). En consecuencia, el algoritmo convergerá a la solución óptima de la ecuación de Bellman para el costo promedio.

4.3. Reescribiendo el modelo

La técnica de aprendizaje por refuerzo necesita las variables que manejan el comportamiento del sistema, por lo que sustituiremos la ley de transición (6) por una ecuación en diferencias, cuyo comportamiento depende de una variable aleatoria. Se seleccionará la variable aleatoria que proporcione un sistema equivalente al que presenta la ley de transición (6). Cabe mencionar que, según la literatura, dado que se conoce explícitamente la matriz de probabilidad de transición y la matriz de costo de transiciones, se espera que en este ejemplo PD supere a AR. La ley de transición (6), descrita para el modelo de línea de espera, se puede modelar mediante la siguiente ecuación en diferencias:

$$X_{t+1} = \begin{cases} (X_t + \xi_{t+1})^+, \text{ si } X_t < N, \\ X_t - (-\xi_{t+1})^+, \text{ en otro caso.} \end{cases} \tag{15}$$

Donde $(H)^+ = max(H, 0)$ para $H \in \mathbb{R}$, X_t representa el estado actual y $\{\xi_t\}$ es una sucesión de variables aleatorias independientes e idénticamente distribuidas con función de masa descrita en el cuadro 1. En este punto, ξ denota un término genérico de ξ_t, $t = 1, 2, 3,\ldots$

CUADRO 1
Función de masa de la variable aleatoria

x	−1	0	1
$P(\xi = x)$	$a(1 - p)$	$ap + (1 - a)(1 - p)$	$(1 - a)p$

Fuente: elaboración propia.

Con AR, el procedimiento para obtener la solución para el problema de optimización vía Q-learnig es el siguiente.

1. Elija una pareja de estado-acción distinguida (i^*, a^*), defina el tamaño de paso α, introduzca k_{max} y considere a i como el estado inicial. Tome $k = 1$, $\epsilon > 0$, y $Q^0(i, a) = 0$, para todo $(i, a) \in X \times A(i)$.
2. Simule una acción $a \in A(i)$ con probabilidad $1/|A(i)|$.
3. El siguiente estado, digamos j, se obtiene mediante la expresión (15).
4. Para $(i, a) \in X \times A(i)$, calcule
$$Q^{k+1}(i, a) \leftarrow (1 - \alpha)\, Q^k(i, a) + \alpha\Big[C(i, a, j) - Q(i^*, a^*) + \min_{b \in A(j)} Q^k(j, b)\Big].$$
5. Actualice $k \leftarrow k + 1$. *Si* $k < k_{max}$ *entonces* $i \leftarrow j$ y regrese al paso 2, en otro caso vaya a 6.
6. El costo promedio es $Q(i^*, a^*)$ y para cada $i \in X$, calcule $d(i) = argmin_{b \in A(i)} Q(i, b)$, donde d denota la $\in$-óptima política y pare.

5. Ejemplos numéricos

En esta sección se trabaja con el modelo de control de Markov: $M = (X, A, \{A(i): i \in X\}, P, C)$ bajo el criterio promedio (5), donde $X = \{0, 1, 2,..., N\}$, y se usan dos tasas de servicio $A = \{a_1, a_2\}$ con $a_1 = 0.2$ y $a_2 = 0.6$. El modelo que conduce el comportamiento del sistema es el presentado en (6) equivalente a (15), donde un cliente ingresa al sistema con probabilidad $p = 0.2$. El costo de servicio en función de la tasa de servicio se supondrá que es $L(a_1) = 2$ y $L(a_2) = 6$, mientras que el costo de atención en función del número de clientes es $M(i) = 3i$, $i \geq 0$. De esta manera, el costo en un paso es $C(i, a) = M(i) + L(a)$, $a \in A$, $i = 0, 1,..., N$. En el algoritmo de PD se considera a 0 como el estado distinguido, mientras que en el de Q-learning se toma a $(0, a_1)$ como la pareja estado-acción distinguida. En los cuadros 2 y 3 la política óptima contiene los términos 1 y 2: 1 representa que el sistema trabaja con la tasa a_1, mientras que 2 representa que el sistema trabaja con la tasa a_2. En los dos casos se consideró un error de $\epsilon = 0.001$. En el cuadro 2 se presenta la evolución de la política óptima y los valores de $w_n(0)$ del AI cuando el espacio de estados es $X_5 = \{0, 1, 2, 3, 4, 5\}$, es decir, $|X_5| = 6$. En este cuadro se observa que cuando $n \to \infty$, el valor promedio óptimo en el estado distinguido 0 tiende a 4.1673.

CUADRO 2
Evolución de W_n y la política óptima cuando $|X_4| = 5$ con PD

n	f_n	$W_n(0)$	n	f_n	$W_n(0)$	n	f_n	$W_n(0)$	n	f_n	$W_n(0)$
0	1 1 1 1 1 1	2	4	1 2 2 2 2 1	3.5476	8	1 2 2 2 2	4.0686	25	1 2 2 2 2	4.1673
1	1 1 1 1 1 1	2.4799	5	1 2 2 2 2 1	3.8274	9	1 2 2 2 2	4.0976	30	1 2 2 2 2	4.1673
2	1 1 1 1 1 1	2.8831	6	1 2 2 2 2 2	3.9541	10	1 2 2 2 2	4.1174	40	1 2 2 2 2	4.1673
3	1 1 1 1 1 1	3.2341	7	1 2 2 2 2	4.0247	20	1 2 2 2 2	4.1673	50	1 2 2 2 2	4.1673

Fuente: elaboración propia.

En el cuadro 3 se presenta la evolución del costo promedio y la política óptima de algunos modelos con espacio de estados $N \geq 3$. En este cuadro e_2^m denota la política que contiene a 1 en la primera entrada y 2 en las $m - 1$ entradas restantes y n es la etapa en la que f_n se estaciona. Observe que cuando N crece, la política tiende a utilizar tasa de servicio más alta y el valor promedio óptimo en el estado distinguido 0 es 4.1685.

El algoritmo de iteración de valores que se utilizó para obtener los datos de los cuadros 2 y 3 es el algoritmo 1 en Portillo-Ramírez (2024). Por otra parte, se ilustrará la metodología de Q-learning: considere el sistema estocástico (15), con estado inicial 0. En el cuadro 4 se presenta la evolución de la política óptima y los valores del costo promedio óptimos, donde $A = 150$ y $B = 3000$ para $N \leq 6$ y $A = 15000$ y $B = 30000$ para $N \geq 7$.

El algoritmo propuesto es el algoritmo 2 en Portillo-Ramírez (2024). Observe que Q-learning provee la solución óptima. La ventaja es que no se necesita la matriz de transición.

CUADRO 3
Costo promedio óptimo y la política óptima con PD

$\|X\|$	n	f_n	$W_n(0)$	$\|X\|$	n	f_n	$W_n(0)$	$\|X\|$	n	f_n	$W_n(0)$	$\|X\|$	n	f_n	$W_n(0)$
3	15	1 2 2	4.0787	6	19	1 2 2 2 2 2	4.1673	9	20	1 2 2 2 2 2 2 2 2	4.1685	30	20	e_2^{30}	4.1685
4	18	1 2 2 2	4.1505	7	20	1 2 2 2 2 2 2	4.1685	10	20	e_2^{10}	4.1685	40	20	e_2^{40}	4.1685
5	19	1 2 2 2 2	4.1650	8	20	1 2 2 2 2 2 2 2	4.1685	20	20	e_2^{20}	4.1685	50	20	e_2^{50}	4.1685

Fuente: elaboración propia.

CUADRO 4
Costo promedio óptimo y política óptima con Q-learning

$N = \|X\|$	k_{max}	f_N	W^*	$N = \|X\|$	k_{max}	f_N	W^*
3	100000	1 2 2	4.0768	7	1000000	1 2 2 2 2 2 2	4.2349
4	100000	1 2 2 2	4.1001	8	1000000	1 2 2 2 2 2 2 2	3.8955
5	100000	1 2 2 2 2	4.3871	9	10000000	1 2 2 2 2 2 2 2 2	4.2718
6	100000	1 2 2 2 2 2	4.4575	10	10000000	e_2^{10}	4.2718

Fuente: elaboración propia.

Prospectiva

Del análisis desarrollado, se constata que Q-learning proporciona soluciones *casi-óptimas*, aunque no se tenga la ley de transición. Únicamente se necesitan las distribuciones de las variables que gobiernan el sistema, lo que refleja una disminución en el cálculo computacional. Sin embargo, se debe tener cuidado, ya que, en general, el tiempo de ejecución de Q-learning supera a PD, por lo que trabajos futuros contemplan enfrentar esta problemática a partir de nuevos mecanismos u optimizando los ya existentes. Lo anterior con el objetivo de proponer una solución en sistemas de líneas de espera a través de procesos de decisión de Markov.

Conclusiones

En este trabajo se propone un modelo de líneas de espera desde el enfoque de procesos de decisión de Markov bajo un criterio de optimalidad promedio. A la par, se muestra el panorama teórico de líneas de espera: procesos de decisión de Markov y aprendizaje por refuerzo. Bajo el enfoque de líneas de espera, se utilizó tanto programación dinámica como aprendizaje por refuerzo para determinar el costo óptimo. Para los resultados teóricos, se realizaron algunos experimentos numéricos en los que se consideró un proceso de control M con espacio de estados finito. Con los ejemplos numéricos, se pudo comprobar que, en efecto, la programación dinámica provee las soluciones óptimas cuando contamos con la ley de transición. Sin embargo, Q-learning no se queda atrás, ya que, sin tener acceso a la ley de transición y con solo una realización del proceso estocástico subyacente, proporciona soluciones casi-óptimas. Los algoritmos utilizados se implementaron en el *software* R.

Agradecimientos

Agradecemos profundamente a los revisores y al editor asociado por su cuidadosa lectura del manuscrito y por sus consejos para mejorar el artículo.

Referencias

Abdulla, M. S., & Bhatnagar, S. (2007). Reinforcement learning based algorithms for average cost Markov decision processes. *Discrete Event Dynamic Systems*, *17*(1), 23-52.

Abounadi, J., Bertsekas, D., & Borkar, V. S. (2001). Learning algorithms for Markov decision processes with average cost. *SIAM Journal on Control and Optimization*, *40*(3), 681-698. https://doi.org/10.1137/S0363012999361977

Afolalu, S. A., Ikumapayi, O. M., Abdulkareem, A., Emetere, M. E., & Adejumo, O. (2021). A short review on queuing theory as a deterministic tool in sustainable telecommunication systems. *Materials Today: Proceedings*, *44*, 2884-2888. https://doi.org/10.1016/j.matpr.2021.01.092

Alfa, A. S. (2016). *Applied discrete-time queues* (2nd. ed.). Springer New York.

Arapostathis, A., Borkar, V. S., Fernández-Gaucherand, E., Ghosh, M. K., & Marcus, S. I. (1993). Discrete-time controlled Markov processes with average cost criterion: A survey. *SIAM Journal on Control and Optimization*, *31*(2), 282-344. https://doi.org/10.1137/0331018

Bellman, R. (1957). *Dynamic programming*. Princeton University Press.

Bertsekas, D. (1995). *Dynamic programming and optimal control: Volume I*. Athena Scientific.

Boucherie, R. J., & Van Dijk, N. M. (Eds.) (2017). *Markov Decision Processes in Practice*. Springer.

Cao, X. R. (2021). *Foundations of average-cost nonhomogeneous controlled Markov chains*. Springer International Publishing.

Feinberg, E. A., & Shwartz, A. (Eds.). (2012). *Handbook of Markov decision processes: methods and applications*. Springer Science & Business Media New York.

Fomundam, S., & Herrmann, J. W. (2007). *ISR Technical Report 2007-24. A survey of queuing theory applications in healthcare*. Institute for Systems Research, 1-22.

Gosavi, A. (2015). *Simulation-based optimization: parametric optimization techniques and reinforcement learning*. Springer.

Gosavi, A. (2008). On step sizes, stochastic shortest paths, and survival probabilities in reinforcement learning. In *2008 Winter*, *Simulation Conference* (pp. 525-531). IEEE. https://doi.org/10.1109/WSC.2008.4736109

Hamilton, M. A., Jaradat, R., Jones, P., Wall, E. S., Dayarathna, V. L., Ray, D., & Hsu, G. S. E. (2018). Immersive virtual training environment for teaching single-and multi-queuing theory: Industrial engineering queuing theory concepts. In *2018 ASEE Annual Conference & Exposition*. https://doi.org/10.18260/1-2--30597

Hernández-Hernández, D., & Minjárez-Sosa, J. A. (Eds.). (2012). *Optimization*, *control*, *and applications of stochastic systems: in honor of Onésimo Hernández-Lerma*. Birkhäuser.

Hillier, F. S. (2005). *Introduction to operations research*. McGraw-Hill Science Engineering.

Lari Dashtbayaz, M., ZolfagharArani, M. H., & Akrami, K. (2019). Competing Earnings Announcements; Study of queuing theory of Investor behavior in the Analysis of Earnings Announcements. *Journal Financial Accounting Knowledge*, *5*(4), 103-126. https://doi.org/10.30479/JFAK.2019.1571

Lovas, A., & Rásonyi, M. (2021). Markov chains in random environment with applications in queuing theory and machine learning. *Stochastic Processes and their Applications*, *137*, 294-326. https://doi.org/10.1016/j.spa.2021.04.002

Portillo-Ramírez, G. (2024). *Un ejemplo de aplicación del aprendizaje por refuerzo en líneas de espera controladas*. https://docs.google.com/document/d/1Y8c_IVcC9ZDAe-WQCakOz6qrtkFrCMuN/edit?pli=1&tab=t.0

Puterman, M. L. (2005). *Markov decision processes: discrete stochastic dynamic programming*. John Wiley & Sons.

R Project (2022). *The R Project for Statistical Computing*. https://www.R-project.org/

Ross, S. M. (1970). *Applied probability models with optimization applications*. Dover Publications Inc New York.

Rykov, V., & Kitaev, M. Y. (1995). Controlled queueing systems. *Journal of Applied Mathematics and Stochastic Analysis*, *8*(4), 433–435. https://doi.org/10.1155/S1048953395000414

Sharma, O. P., & Gupta, U. C. (1982). Transient behaviour of an M/M/1/N queue. *Stochastic Processes and their Applications*, *13*(3), 327-331. https://doi.org/10.1016/0304-4149(82)90019-9

Sennott, L. I. (2009). *Stochastic dynamic programming and the control of queueing systems*. John Wiley & Sons.

CIENCIA *ergo-sum*

Disponible en:

Cómo citar el artículo:

Portillo-Ramírez, G., Cruz-Suárez, H. y Velasco-Luna, F. (2025). Un panorama general del aprendizaje por refuerzo en líneas de espera controladas a tiempo discreto. CIENCIA *ergo-sum*, *32*. https://doi.org/10.30878/ces.v32n0a29

DOI: https://doi.org/10.30878/ces.v32n0a29

Declaración de autoría (CRediT):

Gustavo Portillo-Ramírez: conceptualización, análisis formal, investigación, metodología, administración del proyecto, *software*, validación, visualización, redacción-borrador original, redacción-revisión y edición.
Hugo Adán Cruz-Suárez: conceptualización, análisis formal, investigación, metodología, administración del proyecto, supervisión, validación, visualización, redacción-borrador original, redacción-revisión y edición.
Fernando Velasco-Luna: conceptualización, redacción-borrador original, redacción-revisión y edición.