**UNIVERSIDAD NACIONAL MAYOR DE SAN MARCOS**

FACULTAD DE INGENIERÍA DE SISTEMAS E INFORMATICA 

**CURSO:**

[AUTOMATIZACIÓN Y CONTROL DE SOFTWARE](https://classroom.google.com/u/1/c/NzA3Njc4MTg0Mzk2)

**TEMA:**

----------

**PROFESOR:**

YESSICA ROSAS CUEVA

**ESTUDIANTES:**

* Chuquispuma Merino, Fabricio Vidal: [0009-0008-6474-2691](https://orcid.org/0009-0008-6474-2691)
* Felix Huayhua, Axel Patrick: [0009-0009-6357-9308](https://orcid.org/0009-0009-6357-9308)
* Fernandez Camacho, Geomar Willy: [0009-0005-0838-6504](https://orcid.org/0009-0005-0838-6504)
* Saavedra Monterrey Max Bruno: [0009-0003-7066-2499](https://orcid.org/0009-0003-7066-2499)
* Sanchez Saldaña, Sergio Antonio: [0009-0000-8297-0761](https://orcid.org/0009-0000-8297-0761)

**Lima - Perú**

**2024**

Resumen

El resumen ha de tener una longitud de un párrafo de entre 150 y 250 palabras sin sangría. Los títulos de sección, como la palabra Resumen anterior, no se consideran títulos, por lo que no se usa formato de título en negrita. En su lugar, use el estilo Título de sección. Este estilo inicia automáticamente la sección en una nueva página, por lo que no es necesario que agregue saltos de página. (Para ver el documento con la paginación, seleccione la pestaña Vista y haga clic en Vista de lectura). Tenga en cuenta que todos los estilos de texto de esta plantilla están disponibles en la pestaña Inicio de la cinta, en la galería de estilos.

Palabras clave: Agregue aquí las palabras clave. Para reemplazar este (o cualquier) texto de sugerencia por el suyo, selecciónelo y comience a escribir. No incluya espacios a la izquierda o a la derecha de los caracteres de la selección.

Capítulo 1: Introducción

* 1. Antecedentes del Problema
  2. Formulación del problema
  3. Justificación
  4. Objetivos
     1. Objetivo general
     2. Objetivos específicos
  5. Organización del Artículo

Capítulo 2: Marco teórico

Péndulo Invertido

Algoritmo

Aprendizaje por Refuerzo

DQ Network

Q-Learning

SARSA

Capítulo 3: Estado del arte

## Artículo 1

**Título:** Aprendizaje por refuerzo profundo difuso guiado por conocimiento

**Aporte:** El artículo propone un nuevo método de aprendizaje por refuerzo profundo, llamado KFDQN, que utiliza sistemas difusos para integrar conocimiento previo en el proceso de aprendizaje. Esto mejora la eficiencia del aprendizaje, la estabilidad del entrenamiento y el rendimiento general del modelo. KFDQN integra el conocimiento en la selección de acciones y en la definición de objetivos de aprendizaje. Se valida el algoritmo mediante simulaciones y experimentos con robots móviles, mostrando una mejora significativa en eficiencia y rendimiento. El estudio destaca la importancia de combinar el aprendizaje automático con el conocimiento humano para lograr mejores resultados.

**Proceso:** El estudio se basa en la combinación de aprendizaje por refuerzo profundo (DQN) con un sistema difuso para representar el conocimiento. El método HYAS (Hybrid Action Selection Strategy) se usa para integrar el conocimiento en la selección de acciones, reduciendo la aleatoriedad. HYL (Hybrid Learning Method) combina conocimiento y aprendizaje automático en el objetivo de aprendizaje. Se utilizan tareas en OpenAI Gym (CartPole, MountainCar) y en robots móviles (llegar a un objetivo y evitar obstáculos) para validar el método. Se evalúa el rendimiento en términos de eficiencia de aprendizaje, estabilidad y rendimiento del modelo.

**Resultados:** Los resultados experimentales demuestran una mejora significativa en la eficiencia del aprendizaje (28.6%), el rendimiento (19.56%) y la estabilidad del entrenamiento de KFDQN en comparación con los algoritmos base. KFDQN mostró una mejor eficiencia de aprendizaje en tareas de planificación a corto y largo plazo. En robots móviles, KFDQN logró mejor rendimiento en tareas de alcanzar un objetivo y esquivar obstáculos. KFDQN superó a algoritmos base en estabilidad y tiempos de ejecución en ambas tareas. El método propuesto destaca la eficacia de la integración del conocimiento previo en el aprendizaje por refuerzo.

## Artículo 2

**Título:** Control de péndulo invertido: un estudio comparativo entre el control convencional y el aprendizaje por refuerzo

**Aporte:** Este artículo se aborda la comparación entre métodos de control convencionales (compensador basado en locus de raíces, compensador de estado y control proporcional-derivativo, PD) y un enfoque basado en aprendizaje por refuerzo (Proximal Policy Optimization, PPO) para controlar un sistema de péndulo invertido. Su contribución principal radica en evidenciar que, bajo un entrenamiento adecuado, el aprendizaje por refuerzo puede lograr un rendimiento comparable a los métodos tradicionales sin requerir conocimiento previo del modelo del sistema. Esto resulta valioso en sistemas donde el modelado es complejo o inaccesible.

**Proceso:** El procedimiento descrito en el artículo se dividió en tres fases principales. En el primer apartado, se trató el proceso de modelado y formulación del sistema de péndulo invertido. Mediante los principios de las leyes de Newton, se ha desarrollado un modelo dinámico que explica las relaciones entre el vehículo y el péndulo. Posteriormente, se procedió a linealizar este modelo no lineal con el fin de simplificar la implementación de técnicas de control convencionales. Para la metodología basada en el aprendizaje por refuerzo, se creó una función de recompensa que tuvo en cuenta la ubicación y la inclinación del péndulo, con el objetivo de mantenerlo en una posición estable.  
En la fase dos, se llevaron a cabo la implementación de los algoritmos de control. Se han elegido tres métodos de control convencionales: el compensador basado en el lugar de las raíces, el compensador de estado y el control proporcional-derivativo (PD). Además, se empleó el algoritmo de optimización de políticas proximales (PPO) como un ejemplo de los enfoques de aprendizaje por refuerzo. El último modelo fue entrenado con 250,000 episodios en un entorno de simulación modificado de OpenAI Gym. Este entorno es compatible con acciones continuas y proporciona recompensas basadas en el comportamiento del sistema.  
Por último, en la tercera fase, se procedió a evaluar los algoritmos a través de pruebas realizadas en un modelo de simulación del péndulo invertido. Se examinaron las respuestas temporales, abarcando el tiempo máximo, el tiempo de estabilización y el exceso de respuesta, además de las respuestas en equilibrio, teniendo en cuenta el error en equilibrio y la energía total consumida.

**Resultados:** El examen de los resultados indicó que el compensador que se basa en el locus de raíces tuvo el rendimiento más sobresaliente en general, presentando un tiempo de estabilización rápido y un error en estado estacionario mínimo. Este procedimiento consiguió estabilizar el péndulo de forma eficaz, destacándose sobre otros enfoques en diversos aspectos fundamentales. El algoritmo PPO ha mostrado ser competitivo, especialmente en cuanto a eficiencia energética, al requerir menos energía para estabilizar el sistema en comparación con las técnicas convencionales. No obstante, el rendimiento durante el periodo de estabilización y el momento de mayor actividad fue deficiente, esto se debió a las restricciones de la función de recompensa empleada, la cual no incluyó de manera explícita criterios de rendimiento temporal. Por otra parte, los métodos de compensación de estado y control proporcional-derivativo mostraron un rendimiento más restringido. Específicamente, el control proporcional-derivativo presentó oscilaciones continuas y no logró estabilizar adecuadamente el péndulo en el tiempo asignado para las pruebas.

## Artículo 3

**Titulo:** Modeling an Inverted Pendulum via Differential Equations and Reinforcement Learning Techniques

**Aporte:** Este trabajo demuestra la viabilidad de usar técnicas de aprendizaje por refuerzo (RL), específicamente Deep Q-Network (DQN), para controlar un péndulo invertido sin recurrir a los métodos de control clásicos como PID o LQR. El aporte radica en la aplicación exitosa de DQN a un problema de control dinámico no lineal, mostrando una alternativa robusta y adaptable a la complejidad del sistema. El documento también proporciona un desarrollo detallado de las ecuaciones diferenciales que modelan el sistema y una explicación clara de los fundamentos del aprendizaje por refuerzo.

**Proceso:** El proceso de investigación se puede dividir en estas etapas:  Desarrollo de las ecuaciones diferenciales que describen la dinámica del péndulo invertido, considerando tanto el modelo lineal como el no lineal.Aplicación de la técnica de Deep Q-Learning (DQN) para resolver el problema de control. Esto implica definir el espacio de estados, las acciones posibles, la función de recompensa y la arquitectura de la red neuronal para aproximar la función Q. Entrenamiento de la red neuronal DQN utilizando un entorno de simulación para el péndulo invertido. Se describe la fase de aprendizaje del sistema, incluyendo la optimización de la función de pérdida y la convergencia del algoritmo. Presente los resultados del entrenamiento en forma de gráficas que ilustran la convergencia del algoritmo y su rendimiento en la tarea de estabilizar el péndulo.

**Resultados:**  Los resultados muestran que el algoritmo DQN entrenado logra controlar con éxito el péndulo invertido, manteniéndolo en equilibrio durante un período prolongado. Se presenta una gráfica que muestra la reducción en el número de pasos necesarios para estabilizar el péndulo a medida que avanza el entrenamiento. Aunque no se presentan métricas cuantitativas precisas de rendimiento (tiempo de respuesta, sobreimpulso, etc.), la gráfica indica claramente la efectividad del método propuesto en comparación con los métodos de control tradicionales. El éxito del DQN en este problema no lineal y complejo valida su aplicabilidad como una alternativa a los métodos clásicos.

## Artículo 4

**Titulo:** Control basado en Q-learning para balanceo y equilibrio de péndulo invertido

**Aporte:** La investigación vincula la teoría de control clásica con técnicas

De aprendizaje por refuerzo, contribuyendo para demostrar versatilidad y potencial del aprendizaje automático en tareas de control. Se identificó que el factor de descuento óptimo influye significativamente en el resultado del aprendizaje y en la estabilidad del sistema.

**Proceso:** Para la simulación del modelo se empleó una representación precisa de la dinámica física la cual rigue al sistema, combinada con un algoritmo de aprendizaje por refuerzo para el control de movimientos del carro, las fuerzas actuantes son la fuerza aplicada la fuerza gravitacional del pendulo y la inercia del pendulo. Se establecio un limite en cada lado con una distancia igual desde la posición inicial del carro, lo que se busca es que el angulo en el que oscila el pendulo este dentro de los 20 grados. Con relación al estado, el estado siguiente responde al estado actual y a la acción realizada, los parámetros actualizados derivan de ecuaciones físicas simplificadas. Se hicieron usos de 3 tasas de aprendizajes (0.5,0.7 y 0.9) y 3 factores de descuentos (0.1,0.2 y 0.5)

**Resultados:** El factor de aprendizaje de 0,7 alcanzo la estabilidad por un periodo de (3,6 s), lo que implica que es el equilibrio sostenido mas largo entre los identificados. Con relación al factor de descuento el mas destacable es el 0,2 , es el que establece el equilibrio mas optimo

## Artículo 5

**Título:** Reinforcement learning approach to control an inverted pendulum: A general framework for educational purposes

**Aporte:** Este artículo presenta un marco general para el control de un péndulo invertido utilizando algoritmos de aprendizaje por refuerzo (Reinforcement Learning, RL), destacándose por su aplicabilidad en entornos educativos. Proporciona una guía detallada para implementar los métodos Q-Learning y Deep Q-Networks (DQN) tanto en simulaciones como en experimentos reales. Además, aborda desafíos clave, como el ruido y la fricción, a través de simulaciones que replican las condiciones del mundo real, y valida la robustez de estos métodos para condiciones iniciales variadas. Al incluir un repositorio de código abierto, el trabajo fomenta la experimentación y el aprendizaje práctico de tecnologías avanzadas de control en un contexto académico, abriendo oportunidades para explorar el aprendizaje automático en aplicaciones dinámicas complejas.

**Proceso:** El estudio combina simulaciones y experimentos físicos para implementar el control del péndulo invertido mediante RL. Primero, se desarrolló un modelo físico del sistema y se definió un entorno RL que utiliza estados, acciones y recompensas para guiar el aprendizaje del agente. Se implementaron algoritmos Q-Learning y DQN, comparándolos en términos de eficiencia y precisión. Las simulaciones incluyeron variaciones en parámetros físicos clave como el voltaje, la fricción y el ruido, para evaluar el impacto en el aprendizaje y la estabilidad del control. Finalmente, se realizaron experimentos reales utilizando un sistema equipado con sensores y un motor controlado por un microcontrolador, validando los resultados obtenidos en simulaciones y destacando las ventajas de DQN frente a Q-Learning.

**Resultados:** Los resultados muestran que el algoritmo DQN logró estabilizar el péndulo invertido de manera eficiente y robusta frente a diversas condiciones iniciales y parámetros físicos, mientras que Q-Learning presentó limitaciones significativas en cuanto a tiempo de aprendizaje y precisión. En los experimentos, se identificó que un voltaje mínimo de 5.9 V es esencial para alcanzar un control óptimo, y las simulaciones confirmaron la efectividad del enfoque para manejar ruido y fricción moderados. Este trabajo destaca la capacidad de DQN para generalizar y adaptarse a condiciones variables, ofreciendo un marco educativo sólido para explorar aplicaciones de aprendizaje por refuerzo en sistemas dinámicos.

Capítulo 3: Metodología

## Metodología del desarrollo

La metodología seguida para resolver el problema del péndulo invertido con un carro deslizante basado en Q-Learning se dividió en varias fases clave. Estas fases incluyen la formulación del problema, la implementación del entorno, el diseño del agente de aprendizaje, y la evaluación del desempeño del modelo. A continuación, se describe cada una de estas etapas en detalle.

### Formulación del Problema

El objetivo principal de este estudio fue desarrollar un sistema de control para un péndulo invertido sobre un carro deslizante, que mantuviera el péndulo en su posición vertical y el carro lo más cerca posible de su posición central. Se definieron los siguientes aspectos clave del problema:

* **Espacio de estados**: El estado del sistema fue representado mediante cinco variables continuas: el ángulo del péndulo (), la velocidad angular (), la posición del carro (), la velocidad del carro (), y la aceleración del carro (). Para facilitar el uso del algoritmo de Q-Learning, estas variables fueron discretizadas en un conjunto finito de valores.
* **Acciones posibles**: El agente podía ejecutar tres posibles acciones: mover el carro hacia la izquierda, mantener el carro quieto, o moverlo hacia la derecha. Cada acción tiene un impacto directo en el sistema y puede influir en la estabilidad del péndulo y la posición del carro.
* **Función de recompensa**: La función de recompensa fue diseñada para penalizar los estados donde el péndulo no se encuentra en posición vertical o cuando el carro se aleja demasiado del centro del riel. La recompensa se calcula en función de la distancia del ángulo al valor cero (posición vertical) y de la posición del carro respecto al centro del riel.

## Implementación del Entorno

El entorno fue modelado siguiendo las leyes físicas que gobiernan el movimiento del péndulo y el carro. Se utilizó un modelo simplificado que describía el comportamiento dinámico del sistema, donde las fuerzas que actuaban sobre el carro y el péndulo se reflejaban en las variables de estado mencionadas previamente.

* **Discretización del espacio de estados**: Debido a que el Q-Learning requiere un espacio de estados finito, las variables continuas del sistema (por ejemplo, , , etc.) fueron discretizadas. Se definieron rangos específicos para cada variable y se estableció una resolución de discretización adecuada para garantizar que el espacio de búsqueda fuera lo suficientemente denso para el aprendizaje, pero sin ser demasiado grande como para hacerlo computacionalmente ineficiente.
* **Dinámica del sistema**: La evolución del sistema tras la aplicación de una acción se modeló a través de ecuaciones diferenciales que describen el movimiento del péndulo y el carro. Estas ecuaciones fueron resueltas de forma iterativa en cada paso del agente.

#### Implementación del Agente de Q-Learning

El agente de aprendizaje fue diseñado utilizando el algoritmo Q-Learning, un enfoque de refuerzo basado en la actualización de una tabla de valores , que representa la calidad de tomar una acción en un estado .

* **Estrategia de exploración-explotación**: Se utilizó una política *epsilon-greedy* para gestionar la exploración y explotación. Al principio, el agente realizó más exploración, eligiendo acciones de manera aleatoria con una probabilidad de para descubrir el entorno. A medida que el aprendizaje avanzaba, el valor de disminuía gradualmente, favoreciendo la explotación de las acciones con los valores más altos.
* **Actualización de la tabla** : En cada iteración, después de realizar una acción y observar el siguiente estado y la recompensa obtenida, el valor de la tabla se actualizaba utilizando la ecuación de Bellman. Esta actualización permitió al agente aprender gradualmente cuál era la mejor acción para tomar en cada estado para maximizar la recompensa acumulada.

#### Entrenamiento y Evaluación

El agente fue entrenado mediante simulaciones repetidas del sistema. Durante cada episodio de entrenamiento, el agente interactuó con el entorno, seleccionó acciones según su política , y actualizó la tabla en función de las recompensas obtenidas. Se definieron los siguientes criterios para la evaluación del desempeño:

* **Número de episodios de entrenamiento**: El agente fue entrenado durante un número fijo de episodios para asegurarse de que el proceso de aprendizaje fuera lo suficientemente largo para que el agente pudiera converger hacia una política efectiva.
* **Desempeño del agente**: El desempeño del agente se evaluó observando la capacidad del sistema para mantener el péndulo en posición vertical y el carro centrado durante los episodios de prueba. La recompensa promedio obtenida por el agente a lo largo de los episodios de prueba se utilizó como una métrica para medir la calidad de la política aprendida.

#### Ajuste de Hiperparámetros

Durante el entrenamiento, se realizaron ajustes en varios hiperparámetros clave del algoritmo para optimizar el proceso de aprendizaje:

* **Tasa de aprendizaje** : Controla la velocidad con la que se ajustan los valores en la tabla en función de las recompensas obtenidas. Se utilizó un valor moderado para evitar cambios bruscos en la política durante el entrenamiento.
* **Factor de descuento** : Este parámetro determina la importancia de las recompensas futuras frente a las recompensas inmediatas. Se fijó en , lo que indica que las recompensas futuras son casi tan importantes como las inmediatas.
* **Tasa de exploración inicial** : Inicialmente, se configuró en , favoreciendo la exploración. Posteriormente, se redujo gradualmente hasta un valor mínimo de , favoreciendo la explotación a medida que el agente aprendía.

Capítulo 4: Desarrollo

## Modelado del Sistema

El sistema físico está compuesto por un carro de masa ​ que puede desplazarse sobre un riel horizontal y un péndulo de masa y longitud fijado en la parte superior del carro. La dinámica de este sistema se describe mediante ecuaciones diferenciales acopladas que gobiernan el movimiento del carro y la oscilación del péndulo (Sutton & Barto, 2018). Las ecuaciones del sistema son las siguientes:

1. **Ecuación del péndulo:**
2. **Ecuación del carro:**

**Donde:**

* es la aceleración gravitacional.
* es la masa total del sistema.
* son las aceleraciones lineal y angular, respectivamente.
* es la fuerza aplicada por el agente al carro.

El objetivo del sistema es mantener el péndulo en posición vertical mientras el carro permanece lo más cercano posible al centro del riel.

## Sistema de Recompensas

La función de recompensa está diseñada para incentivar la estabilidad tanto del péndulo como del carro. La expresión utilizada es la siguiente:

Donde:

* es el ángulo del péndulo.
* es la posición del carro.
* es el valor central deseado para .

Esta función penaliza tanto el ángulo (favoreciendo su estabilización) como la posición , para mantener el carro centrado.

## Algoritmo Q-Learning

El algoritmo Q-Learning se utiliza para aprender la política óptima del agente. Este algoritmo actualiza una tabla que almacena los valores esperados de las acciones en cada estado, usando la ecuación de Bellman:

**Donde:**

* es la tasa de aprendizaje.
* es el factor de descuento.
* es la recompensa obtenida en el estado después de realizar la acción .
* es el valor máximo esperado para el siguiente estado.

## Estados y Discretización

El sistema tiene 5 variables de estado continuas que definen el comportamiento del entorno:

1. ángulo del péndulo.
2. velocidad angular del péndulo.
3. posición del carro.
4. velocidad del carro.
5. aceleración del carro.

Estas variables se discretizan para facilitar el aprendizaje, considerando rangos y divisiones específicas para cada una:

* se discretiza en 10 valores entre y .
* se discretiza en 10 valores.
* se discretiza en 10 valores entre los límites del riel.
* se discretiza en 10 valores.
* se discretiza en 5 valores según las posibles aceleraciones.

El número total de estados es el producto de las discretizaciones de cada variable:

## Acciones

El agente tiene tres acciones posibles que puede realizar:

* **0**: Mover el carro hacia la izquierda.
* **1**: Mantener el carro en su posición actual (acción "quieto").
* **2**: Mover el carro hacia la derecha.

Cada acción impacta directamente en las dinámicas de y , y se utiliza para explorar el espacio de soluciones posibles.

## Políticas

Para balancear la exploración y explotación, se utiliza una política *epsilon-greedy*:

Donde disminuye gradualmente a medida que el agente gana experiencia, permitiendo una mayor explotación de la política aprendida.

## Selección de Hiperparámetros

Los hiperparámetros fueron seleccionados basándose en pruebas previas y teorías estándar (Sutton & Barto, 2018):

* **Tasa de aprendizaje** : Permite que el agente actualice los valores de de manera gradual.
* **Factor de descuento** : Permite valorar las recompensas futuras.
* **Tasa de exploración inicial** : Fomenta la exploración del espacio de acciones.

## Parámetros del Entorno y Fuerza

Los parámetros del entorno y la fuerza aplicada son los siguientes:

* Masa del carro:
* Masa del péndulo:
* Longitud del péndulo:
* Fuerza máxima aplicada:

La fuerza es discreta y puede tomar valores dentro de con incrementos de 20 N.

## Exploración y Explotación

El balance entre exploración y explotación es fundamental para lograr un aprendizaje eficiente. En las primeras etapas, la exploración predomina, con alto, permitiendo al agente probar diversas acciones. A medida que avanza el proceso de aprendizaje, la explotación aumenta, favoreciendo las acciones con mayor valor esperado en la tabla .

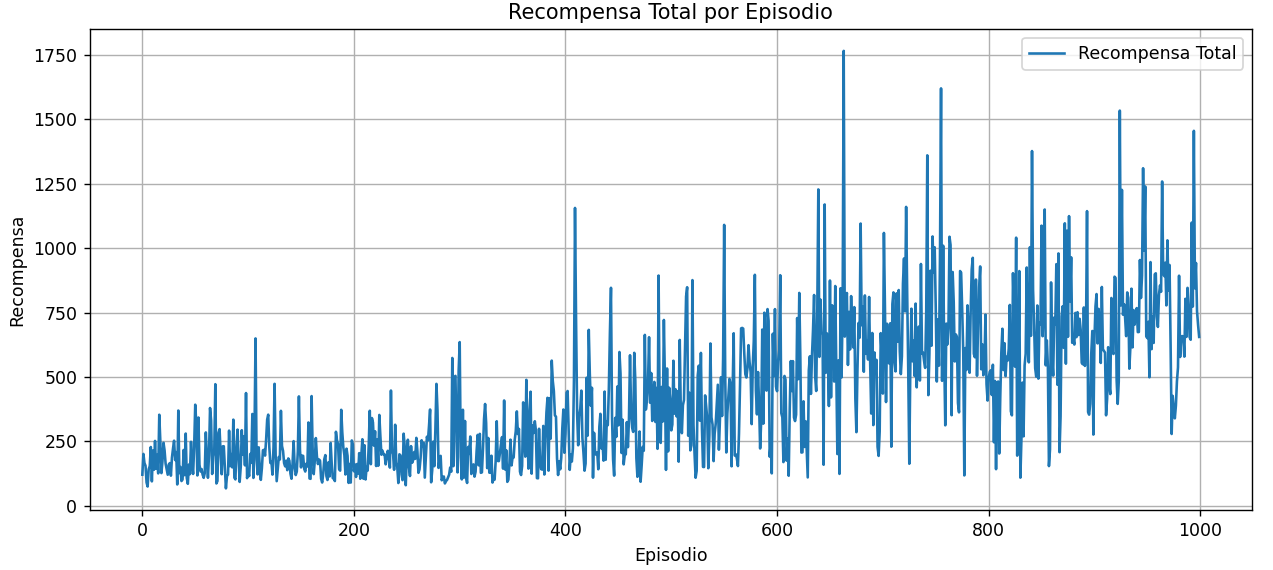
Capítulo 5: Validación y resultados

1. **Análisis de la Influencia de la Longitud del Péndulo y la Fuerza Máxima en el Rendimiento del Agente**

Se evaluaron tres configuraciones experimentales para determinar la influencia de la longitud del péndulo (LENGTH) y la fuerza máxima (MAX\_FORCE) en el rendimiento de un agente. Los resultados se presentan a continuación.

* Caso 1: **LENGTH = 100, MAX\_FORCE = 180.** Esta configuración produjo una recompensa promedio de 767.82 (Figura 1). Este resultado indica un rendimiento estable y eficiente, sugiriendo un balance óptimo entre la longitud del péndulo y la fuerza aplicada. La longitud permitió movimientos correctivos eficientes, mientras que la fuerza fue suficiente para mantener el equilibrio del péndulo.

Figura 1: asas



Nota. *Hasasjas*

Describir el resultado de la imagen

* Caso 2: **LENGTH = 50, MAX\_FORCE = 360.** En este caso, se observó una disminución significativa en el rendimiento, con una recompensa promedio de 408.28 (Figura 2). La reducción en la longitud del péndulo incrementó su velocidad angular, haciendo el sistema más susceptible a errores. A pesar de la mayor fuerza máxima aplicada, no se compensó adecuadamente la alta velocidad angular, resultando en una inestabilidad y menor capacidad para mantener el equilibrio.

Figura 2: asas

Gráfico

Descripción generada automáticamente

Nota: *asdsd*

Descripción de la figura

* Caso 3: **LENGTH = 150, MAX\_FORCE = 90.** Con un péndulo más largo y una fuerza máxima reducida, la recompensa promedio fue de 443.76 (Figura 3). Si bien la mayor longitud facilitó el control al disminuir la velocidad angular, la limitación en la fuerza máxima dificultó la ejecución de correcciones rápidas y precisas, llevando a un rendimiento subóptimo.

Figura 3: assdda

Gráfico

Descripción generada automáticamente

Nota: *asasa*

Descripción de la imagen

Los resultados obtenidos indican la existencia de una relación crucial entre la longitud del péndulo y la fuerza máxima para optimizar el rendimiento del agente. Una longitud mayor proporciona mayor tolerancia a los errores debido a la menor velocidad angular; sin embargo, requiere una fuerza suficiente para realizar ajustes rápidos. Por el contrario, un péndulo corto es más difícil de controlar debido a su alta velocidad angular, y tanto el exceso como la insuficiencia de fuerza pueden generar inestabilidad. Por lo tanto, la selección cuidadosa de ambos parámetros es fundamental para lograr un desempeño óptimo y estable.

1. **Influencia de la Estructura de la Función de Recompensa en el Desempeño del Agente**

Se investigó el impacto de diferentes estructuras de la función de recompensa en el desempeño de un agente, utilizando tres configuraciones que variaban la penalización por el movimiento del carro. El entrenamiento se realizó durante 1000 episodios en cada configuración.

* Caso 1: **Penalización Estándar (k = 0.01).** La función de recompensa se definió como: . Este enfoque produjo una recompensa promedio de 607.16 (Figura 4), representando un balance adecuado entre la estabilidad del péndulo y el movimiento del carro.

Figura 4: titulo

Nota: *asasa*

Descripción de la imagen

* Caso 2: **Penalización Alta (k = 0.05).** Se incrementó la penalización por el movimiento del carro utilizando la función: . La recompensa promedio resultante fue de 598.27 (Figura 5), indicando una ligera disminución en el desempeño en comparación con el caso estándar. Este resultado sugiere que una penalización excesiva restringió innecesariamente el movimiento del carro, afectando negativamente el rendimiento general.

Figura 5: asasa

Nota: *asas*

Descripción de la imagen

* Caso 3: **Penalización Baja (k = 0.001).** Con una penalización reducida , la recompensa promedio obtenida fue de 357.60 (Figura 6). Este desempeño significativamente inferior al de los casos anteriores se atribuye a movimientos descontrolados del carro, comprometiendo la estabilidad del péndulo.

Figura 6: asasa

Nota: asas

Descripción de la imagen

Los resultados muestran una clara dependencia del desempeño del agente en la magnitud de la penalización por el movimiento del carro. La configuración estándar (k = 0.01) ofreció el mejor rendimiento al lograr un equilibrio entre la estabilidad del péndulo y la libertad de movimiento del carro. Aumentar la penalización (k = 0.05) resultó en una ligera disminución del rendimiento, mientras que reducirla (k = 0.001) provocó un desempeño significativamente inferior. Estos hallazgos destacan la importancia de un ajuste preciso de la función de recompensa para optimizar el aprendizaje del agente y lograr un rendimiento satisfactorio.

1. **Tiempo de Convergencia del Agente y Efecto de los Hiperparámetros**

Se analizaron tres configuraciones de hiperparámetros para evaluar su impacto en el rendimiento de un agente de aprendizaje por refuerzo, enfocado en la estabilización de un sistema. Se compararon diferentes tasas de aprendizaje (learning\_rate, ), factores de descuento (gamma, ) y tasas de decaimiento de exploración (epsilon\_decay, ).

* Caso 1: **Configuración Estándar (, , ).** Esta configuración representó un balance entre velocidad de aprendizaje y estabilidad. Se obtuvo una recompensa promedio de 592.95 (Figura 7).

Figura 7: asasa

Nota: asas

Descripción de la imagen

* Caso 2: **Aprendizaje Rápido, Exploración Intensa (, , ).** Esta configuración priorizó la experiencia reciente (α alto) y las recompensas inmediatas (γ bajo). Se observó una recompensa promedio significativamente menor de 227.63 (Figura 8), indicando que, a pesar de la convergencia potencialmente más rápida, la mayor impulsividad del agente afectó negativamente la estabilidad del sistema.

Figura 8: asasa

Nota: asas

Descripción de la imagen

* Caso 3: **Aprendizaje Lento, Exploración Reducida (, , ).** En este caso, se favoreció la estabilidad y un aprendizaje más gradual. La recompensa promedio obtenida fue de 230.67 (Figura 9), similar al caso de aprendizaje rápido. El bajo rendimiento se atribuye a la lentitud en la adaptación de las políticas del agente, priorizando la estabilidad a expensas de la eficiencia.

Figura 9: asasa

Nota: asas

Descripción de la imagen

Los resultados indican que la elección de los hiperparámetros es crucial para el rendimiento del agente. La configuración estándar demostró un óptimo balance, ofreciendo el mejor rendimiento. Los casos de aprendizaje rápido y lento, aunque con objetivos contrastantes, resultaron en un desempeño significativamente inferior, destacando la importancia de una cuidadosa selección y ajuste de los parámetros α, γ y ε para lograr un equilibrio adecuado entre la velocidad de aprendizaje y la estabilidad del sistema.

1. **Respuesta del Agente a Estados Iniciales Desfavorables**

Se evaluó la capacidad del agente para recuperarse de estados iniciales desfavorables, utilizando tres configuraciones que simulaban diferentes situaciones desafiantes. En cada caso, se realizaron 1000 episodios.

* Caso 1: **Ángulo Extremo Cercano al Límite de Caída.** Se inicializó el sistema con un ángulo aleatorio entre y radianes (ángulo muy inclinado) y el carro en la posición central (). La recompensa promedio obtenida fue de 57.61 (Figura 10), indicando una dificultad significativa del agente para estabilizar el péndulo desde esta condición inicial. Se observaron fuertes oscilaciones, reflejando una baja capacidad de recuperación.

Figura 10: asasa

Nota: asas

Descripción de la imagen

* Caso 2: **Carro en Posición Extrema Izquierda.** Se inició con un ángulo θ cercano a la vertical (aleatorio entre -0.05 y 0.05 radianes) y el carro en una posición aleatoria entre 0 y 50 unidades de ancho (cerca del borde izquierdo). La recompensa promedio en este caso fue de 377.47 (Figura 11), mostrando una mejora considerable en comparación con el caso 1. La tendencia al alza sugiere una mayor capacidad del agente para recuperar el equilibrio desde esta posición.

Figura 11: asasa

Nota: asas

Descripción de la imagen

* Caso 3: **Carro en Posición Extrema Derecha con Ángulo Inclinado.** Se simuló un estado inicial con un ángulo θ aleatorio entre y radianes (ángulo inclinado) y el carro en una posición aleatoria entre WIDTH - 50 y WIDTH (cerca del borde derecho). La recompensa promedio obtenida fue extremadamente baja, de 3.79 (Figura 12), indicando un desempeño deficiente del agente en esta situación. La tendencia cercana a cero sugiere una incapacidad para estabilizar el sistema.

Figura 12: asasa

Nota: asas

Descripción de la imagen

Los resultados muestran que el agente presenta dificultades significativas para manejar estados iniciales desfavorables, particularmente cuando se combinan ángulos de inclinación extremos con posiciones extremas del carro. El espacio disponible para maniobras parece ser un factor determinante en la capacidad del agente para estabilizar el sistema. El caso 2, con un ángulo inicial cercano a la vertical y el carro desplazado a la izquierda, demostró un desempeño considerablemente mejor, mientras que las situaciones que involucran ángulos pronunciados y posiciones extremas del carro resultaron en un rendimiento muy pobre. Esto indica áreas de mejora en el algoritmo de aprendizaje o en la función de recompensa para fortalecer la capacidad de recuperación del agente ante condiciones iniciales desafiantes.

1. **Evolución de la Recompensa Acumulada por Episodio Durante el Entrenamiento**

Se analizó la evolución de la recompensa acumulada por episodio durante el entrenamiento del agente, utilizando diferentes números de episodios. Se presentan los resultados para 500, 1000, 10000, 20000 y 40000 episodios.

* Caso 1: 500 Episodios. La Figura 13 muestra la evolución de la recompensa acumulada durante los primeros 500 episodios de entrenamiento. [Se debe agregar aquí una descripción cualitativa de la gráfica, por ejemplo: "Se observa una tendencia ascendente con fluctuaciones significativas, indicando un aprendizaje inicial inestable."]

Figura 13: asasa

Nota: asas

Descripción de la imagen

* Caso 2: 1000 Episodios. La Figura 14 ilustra la evolución de la recompensa durante 1000 episodios. [Se debe agregar aquí una descripción cualitativa de la gráfica, por ejemplo: "La tendencia ascendente se mantiene, pero las fluctuaciones son menores que en el caso anterior, sugiriendo una mayor estabilidad en el aprendizaje."]

Figura 14: asasa

Nota: asas

Descripción de la imagen

* Caso 3: 10000 Episodios. La Figura 15 presenta la recompensa acumulada para 10000 episodios. [Se debe agregar aquí una descripción cualitativa de la gráfica, por ejemplo: "Se observa una convergencia clara hacia un valor de recompensa estable, con fluctuaciones mínimas, indicando un aprendizaje robusto."]

Figura 15: asasa

Nota: asas

Descripción de la imagen

* Caso 4: 20000 Episodios. La Figura 16 muestra la evolución de la recompensa para 20000 episodios. [Se debe agregar aquí una descripción cualitativa de la gráfica, por ejemplo: "La recompensa se mantiene estable alrededor de un valor óptimo, con fluctuaciones insignificantes, confirmando la convergencia del aprendizaje."]

Figura 16: asasa

Nota: asas

Descripción de la imagen

* Caso 5: 40000 Episodios. La Figura 17 presenta la recompensa acumulada después de 40000 episodios. [Se debe agregar aquí una descripción cualitativa de la gráfica, por ejemplo: "El comportamiento es similar al caso anterior, con una recompensa estable y fluctuaciones mínimas, indicando que el aprendizaje ha llegado a un punto de saturación."]

Figura 17: asasa

Nota: asas

Descripción de la imagen

El análisis de las figuras muestra una clara relación entre el número de episodios de entrenamiento y la estabilidad de la recompensa acumulada. Con un número bajo de episodios (500 y 1000), el aprendizaje es inestable, con fuertes fluctuaciones. A medida que aumenta el número de episodios, la recompensa se estabiliza, convergiendo hacia un valor óptimo. A partir de 10000 episodios, se observa una convergencia casi completa, con fluctuaciones mínimas, indicando que el agente ha aprendido eficazmente a estabilizar el sistema. El incremento de episodios más allá de 10000 no produce una mejora significativa en el rendimiento, sugiriendo que se ha alcanzado un punto de saturación en el aprendizaje. Estos resultados resaltan la importancia de un entrenamiento adecuado para lograr un rendimiento óptimo en tareas de aprendizaje por refuerzo.

Capítulo 6: Conclusiones y trabajos futuros

Referencias

Apellidos, Nombre (Año). Título del artículo. *Título del diario*, páginas desde - hasta.

Apellidos, Nombre (Año). *Título del libro.* Nombre de la ciudad: Nombre del editor