Reporte proyecto final: Recomendación de variables manipulables en molinos SAG con Deep Reinforcement Learning Offline

Nombre: José Luis Cádiz.

Curso: Seminario de robótica y sistemas autónomos - EL7021 otoño 2023.

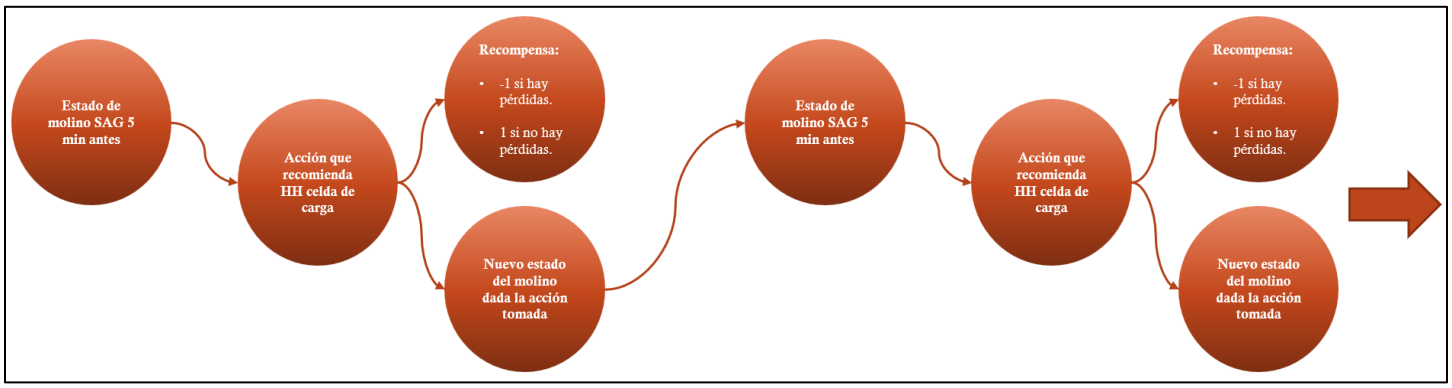
Profesor: Javier Ruiz del Solar.

Introducción: Para abordar el proyecto en cuestión se hizo uso de la librería de Offline Reinforcement Learning, **d3rlpy**, esto con el objetivo de centrar los esfuerzos en el modelamiento del proceso que se busca optimizar. A continuación, se muestran los resultados del modelamiento **"Estabilización de TPH mediante recomendación de su nivel de carga"**. El algoritmo utilizado para aprender las políticas optimas fue **Advantage Weighted Actor-Critic.**

Para verificar que las políticas aprendidas son consistentes, se compararan las distribuciones de lo que indican las políticas aprendidas vs la política adoptada según la data histórica, adicionalmente se analizaran algunas series de tiempo con el objetivo de chequear la consistencia de las políticas.

Modelamiento:

- **Estados**: Agua, porcentaje de sólidos, rpm, HH TPH, granulometría, celda de carga y Setpoint HH celda de carga 5 minutos antes.
- **Acciones**: Setpoint HH celda de carga.
- **Reward**:
  - 1 si no hay perdidas → Pérdidas menores a 100.
  - -1 si hay perdida → Pérdidas mayores a 100.
- **Modelo**: **Advantage Weighted Actor-Critic**.
- **Esquema:**



Resultados:

1. **Estabilización de TPH mediante recomendación de su nivel de carga**:

   - **Visualizaciones**:

Figura 1: Comparación de distribución de políticas.

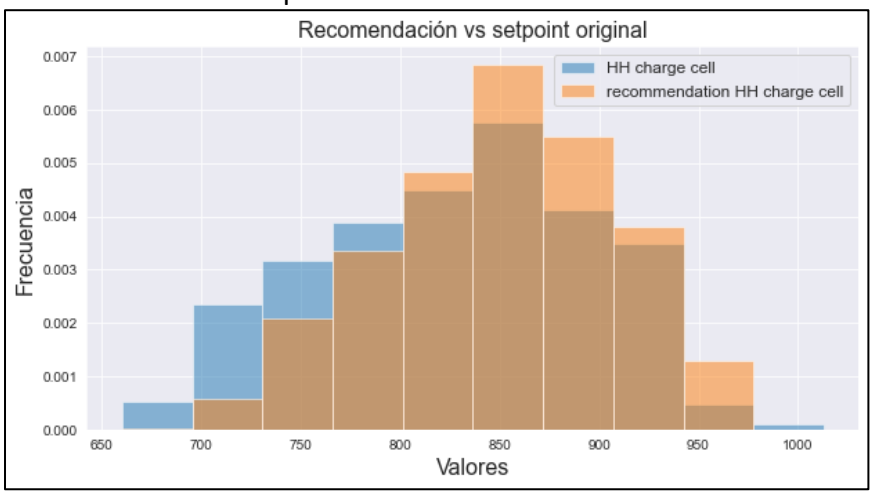


Figura 2: Estado de pérdida, la recomendación sugiere subir el límite de carga, se observa que TPH cae al no seguir recomendación.

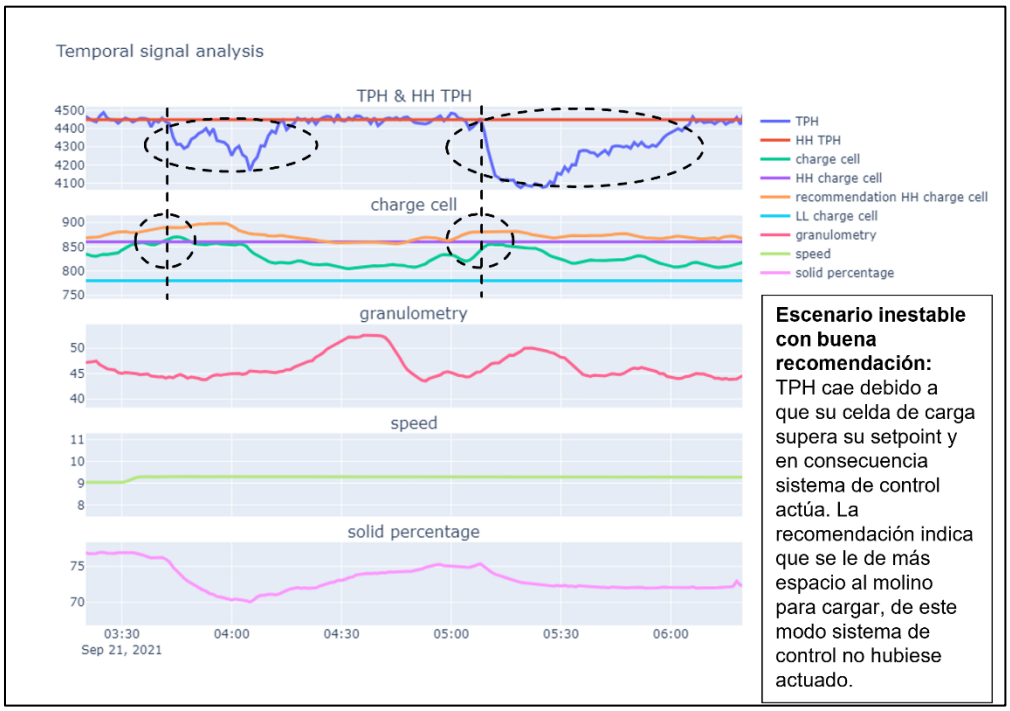
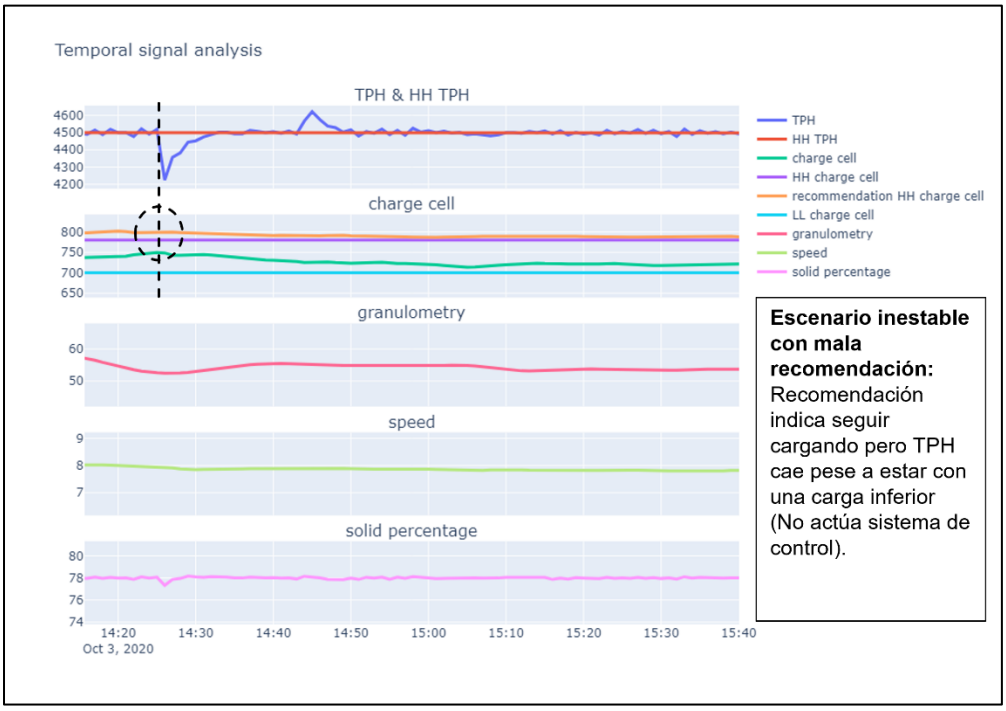


Figura 3: Estado de no pérdida, caso en que recomendación está en torno a la decisión que tomo la operación.

Figura 4: Estado de pérdida, caso en que la recomendación no es buena debido a que TPH cae pese a que celda de carga no supera su setpoint.



- **Análisis**:
  A partir de la figura 1, se observa que la distribución de recomendaciones generadas por el algoritmo AWAC esta dentro de los rangos admisibles al compararlo con la distribución histórica del setpoint de celda de carga.

  En la figura 2, se observa como el modelo es capaz de anticipar los eventos de caída de TPH, recomendando que el setpoint de celda de carga sea aumentado para que el sistema de control no actúe.

  Por otro lado, en la figura 3 se observa como la recomendación del modelo es muy similar al setpoint real en el caso en que el TPH se encuentra estable.

  Finalmente, a partir de la figura 4, se evidencia un caso en que la recomendación del modelo no es correcta, ya que pese a que el sistema de control no actúa (su celda de carga no supera su setpoint) pero el molino SAG se embanca.

Conclusiones: Si bien el enfoque Offline del Reinforcement Learning permite aprender políticas optimas a partir de data histórica sin necesidad de experimentar con la planta, de igual manera surge la necesidad de poder testear las políticas aprendidas y para esto es necesario tener al menos un modelo de la planta que permita simular la incorporación de las políticas aprendidas dentro del sistema.

A partir de la necesidad de tener un modelo para poder testear las políticas aprendidas, se genera una cierta contradicción, ya que, si se tuviese un modelo de la planta, lo podríamos utilizar como ambiente y aprender políticas optimas a partir de métodos convencionales de Reinforcement Learning.

Reinforcement Learning Offline es método muy interesante para aprender políticas optimas solamente a partir de data histórica, sin embargo, es necesario diseñar metodologías para poder testear de manera robusta las políticas aprendidas a partir del mismo tipo de estructura de información con la cual se entrena el algoritmo.

También es importante mencionar que para obtener resultados satisfactorios utilizando este enfoque de aprendizaje, es muy importante entender la dinámica del sistema que se busca optimizar, ya que es de vital importancia modelar de manera correcta dichas dinámicas como un proceso de decisión de Márkov.

Se concluye que los resultados han sido parcialmente satisfactorios ya que, si bien las políticas aprendidas son consistentes con los resultados esperados a partir de conocimiento previo, no se puede decir de manera categórica que las políticas aprendidas son óptimas o seguras debido a que no es posible la experimentación.

Finalmente, a partir de todo el proceso de experimentación con la data, entrenamiento e intento de testear, se concluye que, si se llega a tener un buen modelo de simulación de la planta junto con data histórica del comportamiento de esta, debería ser posible alcanzar muy buenos resultados en el proceso de encontrar políticas optimas de una manera robusta y segura.