
Instituto Tecnológico de Costa Rica**Escuela de Ingeniería Electrónica****Trabajo Final de Graduación****Proyecto:** Método basado en aprendizaje reforzado para el control automático de una planta no lineal.**Estudiante:** Oscar Andrés Rojas Fonseca

I Semestre 2024

Firma del asesor

Bitácora de trabajo

Fecha	Actividad	Anotaciones	Horas dedicadas
11/03/2024	1. Pruebas con las versiones de las clases <i>CartPoleSystem</i> y <i>CartPoleMpc</i> .	a) Luego del nuevo estudio del código original de ejemplo [1], se evidencia que las condiciones o parámetros definidos no son adecuados.	3 horas
12/03/2024	2. Continuación de pruebas para la definición de las clases referentes a <i>CartPole</i> .	a) Se definió una nueva versión de la clase <i>CartPoleSystem</i> utilizando las funciones <i>_init_()</i> , <i>reset()</i> , <i>step()</i> , <i>render()</i> , <i>close()</i> y <i>get_stage_cost()</i> . b) Revisión de parámetros funcionales de los ejemplos con la adaptación a <i>CartPole</i> .	6 horas
13/03/2024	3. Reunión de seguimiento con el asesor del proyecto.	a) Revisión de avance en el código y resaltado de errores de forma. b) Se indicó los puntos importantes por replantear.	2 horas
13/03/2024	4. Continuación de pruebas para la definición de las clases referentes a <i>CartPole</i> .	a) Prueba con resultado incorrecto de las matrices que definen la dinámica del sistema.	4 horas

14/03/2024	5. Reunión de seguimiento con coordinador del TFG a las 15 : 00.	a) No asistí a la reunión por motivos personales. b) Se contactó al coordinador mediante correo electrónico para solicitar información.	1 horas
15/03/2024	6. Hora consulta con coordinador del TFG 08 : 00.	a) Se consultó mediante llamada telefónica para solicitud de información respecto a la reunión. b) Redacción de correo electrónico con datos solicitados por el coordinador.	1 horas
15/03/2024	7. Continuación de pruebas para la definición de las clases referentes a CartPole.	a) Los parámetros del ejemplo <i>dpg.py</i> son diferentes al caso del <i>CartPole</i> por el objetivo de aprendizaje del modelo. b) Se redefinieron las funciones <i>_init_()</i> , <i>dynamics()</i> , <i>plan()</i> y <i>execute()</i> de la clase <i>CartPoleMpc</i> . c) Nuevamente se tienen errores de definición, se debe replantear la clase <i>CartPoleMpc</i> .	6 horas
Total de horas de trabajo:			23 horas

Contenidos de actividades

Al observar los ejemplos del repositorio *MPC–RL* [1] y al replantear las clases *CartPoleSystem* y *CartPoleMpc*, se definió nuevamente la primera con la estructura siguiente:

CartPoleSystem(gym.Env):

- *_init_(self)*: Define el tipo de *environment* virtual de *Gymnasium* y guarda las observaciones del espacio de estados y las acciones.
- *reset(self)*: Reinicia el proceso del ambiente virtual.
- *step(self, action)*: Recibe la acción, la aplica al ambiente y da un paso en el proceso.

- *render(self, mode)*: Plantea el tipo de proceso a observar entre gráfico (*human*) y numérico (*rgb_array*).
- *close(self)*: Termina el proceso del ambiente virtual.
- *get_stage_cost(self, state)*: Define un primer comportamiento del ambiente virtual para pantearlo a la recompensa.

Por otro lado, la clase *CartPoleMpc* presenta muchos errores, por lo que se debe analizar cuidadosamente y replantear por completo.

Referencias

- [1] F. Airaldi, A. Bietti, A. Casagrande, and A. Bemporad, “Learning model predictive control with policy gradients,” *IEEE Transactions on Automatic Control*, 2023.