CEA Comité Español de Automática
XLVI JORNADAS AUTOMÁTICA CARTAGENA 2025
UNIVERSIDADE DA CORUÑA

Jornadas de Automática

Manipulación robótica mediante aprendizaje por refuerzo inverso con características basadas en trayectorias expertas

Naranjo-Campos, Francisco J.*; Victores, Juan G.; Calzada-Garcia, Ana; Balaguer, Carlos

Robotics Lab, Departamento de Ingeniería de Sistemas y Automática, Universidad Carlos III de Madrid, C/ Butarque, nº15, Leganés, 28911, Madrid, España.

To cite this article: Naranjo-Campos, F. J., Victores, J. G., Calzada-Garcia, A., Balaguer, C. 2025. Robotic Manipulation Using Inverse Reinforcement Learning with Expert-Trajectory-Based Features. Jornadas de Automática, 46. https://doi.org/10.17979/ja-cea.2025.46.12175

Resumen

Los algoritmos de aprendizaje para manipulación robótica aún presentan desafíos en tareas con alta variabilidad y dimensionalidad. Entre ellos, el Aprendizaje por Refuerzo ha mostrado buenos resultados, pero está limitado por la definición de la función de recompensa. Por ello surgen los algoritmos de Aprendizaje por Refuerzo Inverso (IRL), que estiman la recompensa a partir de demostraciones de un experto. En este trabajo se propone y valida un enfoque de IRL basado en características extraídas de trayectorias expertas, aplicado a tareas de manipulación con el robot TIAGo++. Este método aprovecha las demostraciones para centrar la definición de las características en la zona del espacio de estados relevante para el experto, así como priorizar los estados finales cercanos al objetivo. Las tareas de manipulación seleccionadas fueron apilar bloques y abrir un armario. Se entrenaron en simulación y se transfirieron al robot real, demostrando la viabilidad y eficacia del enfoque tanto en la ejecución exitosa como en métricas de distancia respecto al experto.

Palabras clave: Aprendizaje Automático, Manipulación Robótica, Aprendizaje por Refuerzo Profundo, Aprendizaje por Refuerzo Inverso, Inteligencia Artificial.

Robotic Manipulation Using Inverse Reinforcement Learning with Expert-Trajectory-Based Features

Abstract

Algorithms for robotic manipulation learning still face challenges in tasks with high variability and dimensionality. Among them, Reinforcement Learning has shown good results but is limited by the definition of the reward function. This has led to the development of Inverse Reinforcement Learning (IRL) algorithms, which estimate the reward from expert demonstrations. This work proposes and validates an IRL approach based on expert-trayectory based features, applied to manipulation tasks with the TIAGo++ robot. The method leverages demonstrations to focus the feature definition on the relevant regions of the state space for the expert, as well as to prioritize final states close to the goal. The selected manipulation tasks were block stacking and cabinet opening. They were trained in simulation and transferred to the real robot, demonstrating the viability and effectiveness of the approach both in successful execution and in distance-based metrics relative to the expert.

Keywords: Machine Learning, Robotic Manipulation, Deep Reinforcement Learning, Inverse Reinforcement Learning, Artificial Intelligence

1. Introducción

La manipulación robótica es fundamental y está bien consolidada en entornos industriales. Sin embargo, en entornos domésticos sigue planteando desafíos, ya que son espacios dinámicos donde resulta difícil implementar soluciones robustas para tareas complejas.

*Autor para correspondencia: frnaranj@ing.uc3m.es

En este contexto, el Aprendizaje por Refuerzo (RL, Reinforcement Learning) (Sutton and Barto, 2018) ha demostrado ser útil en aplicaciones de control robótico (Kober et al., 2013). No obstante, requiere diseñar una función de refuerzo adecuada que defina el objetivo de la tarea y guíe al agente durante el aprendizaje (Eschmann, 2021), lo cual suele demandar conocimiento experto y numerosos ajustes.

Como alternativa, el Aprendizaje por Refuerzo Inverso (IRL, Inverse Reinforcement Learning) (Ng and Russell, 2000) utiliza demostraciones expertas para estimar la función de refuerzo, capturando aspectos críticos de la tarea. Esto puede traducirse en movimientos más naturales y eficientes del robot (Ziebart et al., 2008).

Desde su formalización, se han propuesto distintos enfoques de IRL. Los primeros utilizaban características manuales, como *Apprenticeship Learning via IRL* (AL-IRL) (Abbeel and Ng, 2004), y técnicas basadas en entropía para reducir ambigüedad y sesgos (Ziebart et al., 2008). Más adelante, se incorporaron técnicas de aprendizaje profundo (Wulfmeier et al., 2015) para evitar la ingeniería manual, así como enfoques probabilísticos, como procesos gaussianos (Jin et al., 2017) y estimación por máxima aproximación (Zeng et al., 2023). También destacan métodos basados en redes generativas adversarias (Fu et al., 2017; Sun et al., 2021), con excelentes resultados en manipulación robótica.

Otro avance importante es cómo manejar las demostraciones de expertos para extraer información útil. Algunos desafíos incluyen aprender de múltiples expertos (Likmeta et al., 2021), manejar expertos subóptimos (Poiani et al., 2024), lo que ayuda a reducir ambigüedad, y estimar modelos del proceso de decisión del experto para generar datos adicionales (Hoshino et al., 2022; Beliaev and Pedarsani, 2024). Además, en tareas con decisiones dependientes del tiempo, se ha usado IRL con recompensas temporales (Ashwood et al., 2022), permitiendo modelar entornos complejos de forma más precisa.

A pesar de la variedad de métodos disponibles, AL-IRL sigue siendo valioso por su marco claro y explicable, clave en robótica para garantizar transparencia y control, y porque se integra bien con técnicas modernas de optimización. Sin embargo, su desempeño depende críticamente de la elección y escalado adecuado de las características, lo que puede reintroducir desafíos manuales. Además, métodos comunes para generalizar características, como la discretización o el agrupamiento aleatorio (Ng and Russell, 2000; Abbeel and Ng, 2004; Neu and Szepesvári, 2012), enfrentan limitaciones en espacios de alta dimensión (Bellman, 1961). Finalmente, el uso de descuentos puede reducir la relevancia de las características cercanas al objetivo, especialmente en tareas con estados iniciales concentrados o metas escasas.

En este contexto surge la propuesta de características basadas en trayectorias expertas para AL-IRL (*Expert-trajectory-Based Features for AL via IRL*, EXBAL-IRL) (Naranjo-Campos et al., 2024), donde el espacio de características se define a partir del muestreo de las trayectorias del experto y del entorno circundante. Este enfoque permite centrar la representación en las regiones del espacio de estados más relevantes para el comportamiento experto, capturando las características esenciales de la tarea y dejando de lado detalles irrelevantes. Para evitar la pérdida de información sobre los estados objetivo, también se propone una aplicación inversa del factor de descuento, que da mayor peso a las expectativas de características en los estados finales, preservando así los elementos clave asociados a la meta. Además, se combina este esquema con *Proximal Policy Optimization* (PPO) (Schulman et al., 2017), aprovechando su estabilidad y eficiencia en espacios de acción continuos.

En este trabajo se presenta la aplicación de EXBAL-IRL a tareas de manipulación robótica, demostrando su implementación y validación en el robot real TIAGo++. Esta propuesta busca mostrar la viabilidad y eficacia del método en escenarios reales, destacando su potencial para resolver tareas complejas en entornos físicos más allá de la simulación. El modelo es entrenado inicialmente en simulación y posteriormente transferido al robot real. En la Figura 1 se muestra un esquema de la implementación desarrollada en este trabajo.

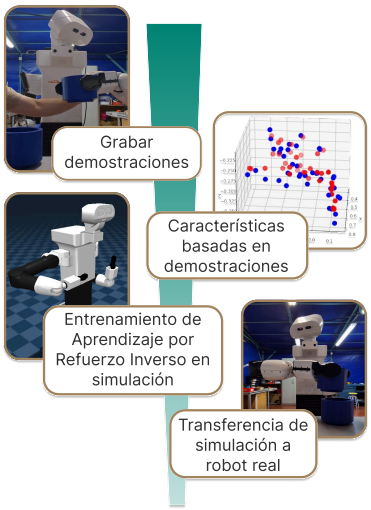


Figura 1: Esquema de la implementación de EXBAL-IRL con el robot TIAGo++. Primero, se graban las demostraciones del experto, luego se definen las características a partir de ellas y se entrena el modelo en simulación. Finalmente, el modelo se transfiere al robot real.

2. Materiales y métodos

En esta sección se presenta el marco teórico de este trabajo, la descripción del algoritmo implementado, así como su aplicación a un robot real.

2.1. *Marco teórico*

El Aprendizaje por Refuerzo (RL) es un tipo de aprendizaje automático en el que un agente aprende a tomar decisiones mediante la interacción con su entorno. El problema puede formularse como un Proceso de Decisión de Markov (MDP, *Markov Decision Process*), definido por la tupla $\mathcal{M} = \{\mathcal{S}, \mathcal{A}, p(s,a), \rho_0, r\}$, que incluye el espacio de estados $\mathcal{S}$, el espacio de acciones $\mathcal{A}$, la función de probabilidad de transición $p(s,a)$, la distribución inicial ρ_0 y la función de refuerzo r. Un algoritmo de RL busca optimizar una política $\pi(a|s)$ que maximice la obtención de refuerzos.

Por su parte, el Aprendizaje por Refuerzo Inverso (IRL) busca inferir la función de refuerzo a partir de demostraciones expertas, con el objetivo de optimizar una política. Puede definirse como la tupla $\mathcal{M}\backslash\mathcal{R} = \{\mathcal{S}, \mathcal{A}, p(s,a), \rho_0, \pi_E\}$, donde π_E representa la política del experto.

La implementación *Apprenticeship Learning via IRL* (AL-IRL) es el algoritmo base de este trabajo. En este enfoque, se asume que es posible definir un vector de características ϕ, tal que la función de refuerzo se expresa como:

$$R(s) = w^T \phi(s), \tag{1}$$

donde w representa los pesos asociados a cada característica. A continuación, se define la expectativa acumulada sobre una serie de m trayectorias como:

$$\hat{\mu}_E = \frac{1}{m} \sum_{i=1}^{m} \sum_{t=0}^{T} \gamma^t \phi_t^{(i)}, \tag{2}$$

donde $\gamma \in [0, 1]$ es el descuento, T es la longitud de cada trayectoria, y ϕ_t^i representa el vector de características en el paso t de la trayectoria i.

El objetivo de AL-IRL es encontrar una política cuya expectativa acumulada se aproxime a la obtenida a partir de las trayectorias del experto, definida en la expresión 3.

$$\|\mu(\tilde{\pi}) - \mu_E\|_2 \leq \epsilon, \tag{3}$$

donde ϵ es el umbral de error permitido, μ_E es la expectativa acumulada del experto, y $\mu(\tilde{\pi})$ la correspondiente a la política $\tilde{\pi}$.

En cada iteración, se optimiza la política utilizando un método de RL y la función de refuerzo con los pesos actualizados en esa iteración.

2.2. *Aprendizaje por refuerzo inverso usando características basadas en trayectorias expertas*

El aprendizaje por refuerzo inverso usando características basadas en trayectorias expertas (EXBAL-IRL) define el espacio de características identificando las zonas del espacio de estados más relevantes.

Primero, dadas las trayectorias del experto $\mathcal{T}_E$, se realiza una agrupación mediante el algoritmo K-means (Jain, 2010), obteniendo C centros de *cluster*. A continuación, se calcula el radio mínimo promedio r_m entre estos centros, y se definen L capas concéntricas alrededor de cada centro, cada una con un ancho de $2r_m$. Luego, se generan K puntos aleatorios por centro y por capa, lo que da lugar a un total de:

$$F = C + C \cdot K \cdot L \tag{4}$$

De esta forma, los F centros de característica, representados por el vector ς, codifican tanto la información central del experto como su entorno.

La relación entre un estado s y su vector de características se calcula mediante una *Radial Basis Function* (RBF) gaussiana (Arora et al., 2023):

$$\phi(s \mid \mathcal{T}_E, \varsigma) = \exp\left(-\frac{\|s - \varsigma\|_2^2}{2r_m^2}\right), \tag{5}$$

donde r_m es el radio mínimo promedio entre los centros.

En la Figura 2 se ilustra como ejemplo el espacio de características que se genera en un espacio de estados 2D con $C = 2$, $K = 2$, y $L = 2$.

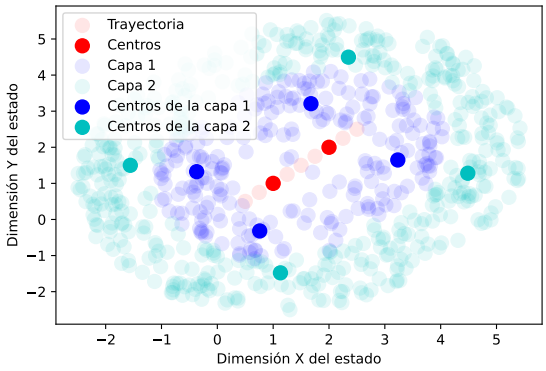


Figura 2: Ejemplo de los centros que representan el espacio de características en un espacio de estados 2D. La trayectoria experta se muestra en rojo claro; los $C = 2$ centros agrupados, en rojo; los puntos aleatorios en las $L = 2$ capas, en azul claro y cian claro; y los $C \cdot K \cdot L = 8$ centros de capa, en azul y cian.

Por otro lado, se aplica un descuento reverso en la expectativa acumulada, con el objetivo de dar mayor relevancia a las características de las zonas finales asociadas al objetivo. La expectativa acumulada queda entonces definida en la Ecuación 6:

$$\hat{\mu}^r = \frac{1}{m} \sum_{i=1}^{m} \sum_{t=0}^{T} \gamma^{T-t} \phi_t^{(i)}, \tag{6}$$

Finalmente, se utiliza el método PPO como algoritmo de RL. Como resultado, se obtiene el Algoritmo 1.

Algorithm 1 Aprendizaje por refuerzo inverso con características basadas en trayectorias expertas (EXBAL-IRL)

1: Genera F centros de características a partir del experto siguiendo los descrito en la sección 2.2.
2: Escoge aleatoriamente $\pi^{(0)}$.
3: Calcula $\mu^{(0)} = \mu(\pi^{(0)})$ con descuento reverso, usando (6).
4: Calcula μ_E con descuento reverso, usando (6).
5: Establece $i = 0$.
6: Establece $\overline{\mu}^{(0)} = \mu^{(0)}$ y $w^{(1)} = \mu_E - \mu^{(0)}$.
7: **Repetir**
8: Computa política óptima $\pi^{(i)}$ con PPO y $r = (w^{(i)})^T \phi$.
9: Calcula $\mu^{(i)} = \mu(\pi^{(i)})$ con descuento reverso con (6).
10: Establece $i = i + 1$.
11: Optimiza $\overline{\mu}^{(i-1)}$ con la función de pérdida (3).
12: Establece $w^{(i)} = \mu_E - \overline{\mu}^{(i-1)}$.
13: Establece $t^{(i)} = \|w^{(i)}\|_2$.
14: **Hasta que** $t^{(i)} < \epsilon$

2.3. *Implementación en robot real TIAGo++*

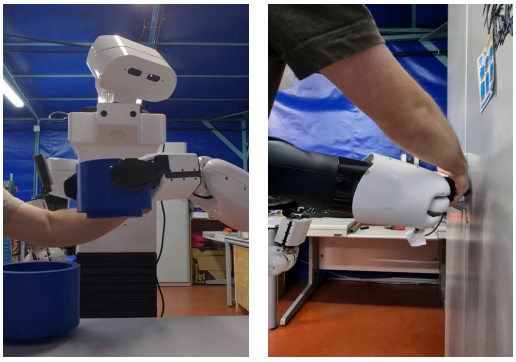
En este trabajo se ha aplicado el algoritmo EXBAL-IRL en tareas de manipulación con el robot móvil bi-manipulador TIAGo++[1] de PAL Robotics. Presenta un diseño modular que permite adaptarse a los requisitos específicos de la tarea de investigación. En su configuración por defecto, cuenta con una base móvil sensorizada para navegación autónoma, un torso prismático, dos brazos de siete grados de libertad, dos pinzas como efector final, sensores de fuerza en los efectores finales y una cámara RGB-D montada sobre un sistema de giro e inclinación.

[1] Ver `https://blog.pal-robotics.com/tiago-bi-manual-robot-research/`, consultado el 19 de marzo de 2025.

La cadena cinemática en formato URDF permite calcular cinemática directa e inversa para alcanzar posiciones del efector final cuando es factible. Estos cálculos se realizan con PyKDL[2], una herramienta para cálculos cinemáticos eficientes.

Para el aprendizaje se desarrolló un entorno usando Gymnasium[3], conectado a una simulación con MuJoCo[4] y el modelo del TIAGo++[5]. Los movimientos se ejecutan mediante comandos de posición articular enviados por ROS al brazo derecho. El estado incluye los valores articulares y el cuaternión de la pose del efector final; las acciones son incrementos articulares deseados.

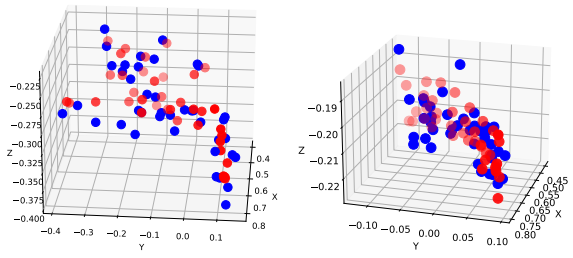
Las tareas de manipulación seleccionadas para el robot son apilar bloques y abrir un armario. Para ello, se grabaron 50 trayectorias de demostraciones expertas por tarea, guiando el robot mediante compensación de gravedad, es decir, moviendo manualmente el brazo, como se muestra en la Figura 3.



(a) (b)

Figura 3: Grabación manual de trayectorias expertas empleando compensación de gravedad: (a) apilado de bloques y (b) apertura de armario.

Para aplicar EXBAL-IRL, el espacio de características se definió a partir de la pose del efector final. Tras un ajuste iterativo, se seleccionaron los valores $C = 40$, $K = 1$ y $L = 1$. La Figura 4 muestra las posiciones de los centros generados (sin orientación, para facilitar la visualización).



(a) (b)

Figura 4: Visualización de las posiciones de los centros que definen los espacios de características de (a) la tarea de apilar bloques y (b) la tarea de abrir un armario. En rojo los centros extraídos de las trayectorias y en azul los que rodean a estos.

3. Resultados

En esta sección se presentan los resultados obtenidos. La tarea de manipulación ha sido entrenada con el algoritmo EXBAL-IRL en simulación y luego se ha transferido la política obtenida al robot real.

3.1. *Entrenamiento en simulación*

Para cada tarea, se realizaron un total de diez entrenamientos, cada uno compuesto por un millón de pasos de interacción. Los principales hiperparámetros empleados durante este proceso se recogen en la Tabla 1, los cuales fueron seleccionados tras un proceso de ajuste empírico basado en múltiples pruebas preliminares.

Tabla 1: Principales hiperparámetros utilizados

Parámetro	**Valor**
α (learning_rate)	0.0001
γ (factor de descuento)	0.99
λ (GAE)	0.95
Número de entornos paralelos	10
Pasos por entorno (T)	128
Número de minibatches	32
Épocas por actualización	3
Coeficiente de recorte (clip)	0.2

Para evaluar los resultados, se emplea el valor t, que representa la distancia entre la expectación acumulada del experto y la del agente, es decir, cuán similares son las características obtenidas por el agente respecto al experto. En las Figuras 5 y 6 se muestran las gráficas de esta distancia para ambas tareas de apilar bloques y abrir armario, indicando la media y la desviación típica. Se observa una disminución progresiva a lo largo del entrenamiento.

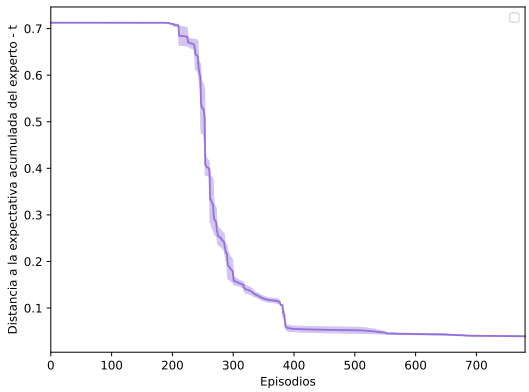


Figura 5: Distancia t entre la expectativa acumulada del experto y la del agente, media y desviación estándar sobre 10 entrenamientos en simulación de la tarea de apilar bloques.

[2] `https://docs.ros.org/en/diamondback/api/kdl/html/python/`, consultado el 19 de marzo de 2025
[3] `https://gymnasium.farama.org/`, consultado el 19 de marzo de 2025
[4] `https://mujoco.readthedocs.io/en/stable/overview.html`, consultado el 19 de marzo de 2025
[5] `https://github.com/pal-robotics-forks/mujoco_menagerie/tree/140ae8d30b430d9d8d8f0c42e031b93b59cb2968/pal_tiago_dual`, consultado el 19 de marzo de 2025

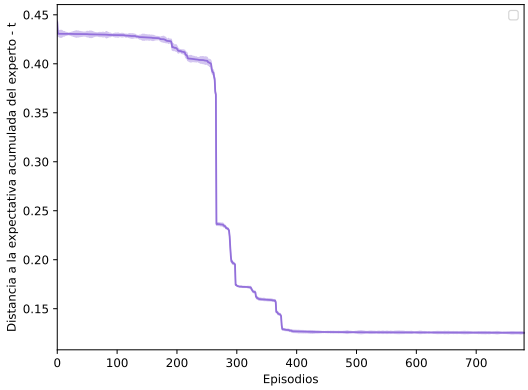
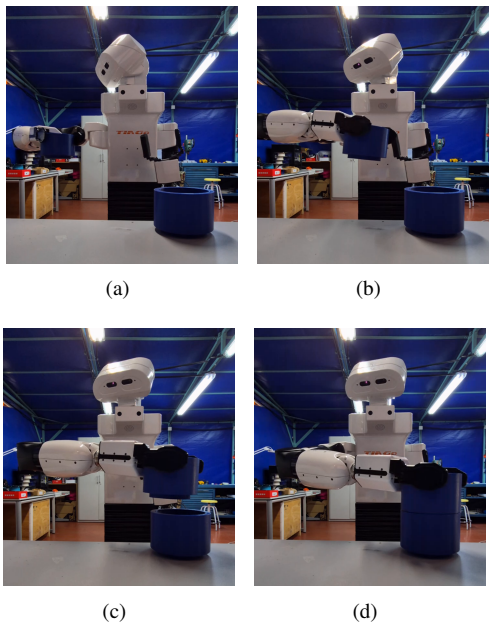


Figura 6: Distancia t entre la expectativa acumulada del experto y la del agente, media y desviación estándar sobre 10 entrenamientos en simulación de la tarea de abrir armario.

3.2. *Trasferencia a robot real*

Para aplicar la política aprendida al robot real, se ha optado por ejecutar en paralelo el entorno de simulación y el robot físico. La política opera directamente sobre la simulación, generando comandos de posición que son ejecutados en simulación. Luego, la posición alcanzada en simulación es comandada al robot real para su ejecución. De esta forma se evita comandar al robot real con acciones que resulten en configuraciones no seguras o singularidades. A su vez, las posiciones alcanzadas por el robot real son retroalimentadas a la simulación, manteniendo así la coherencia entre ambos entornos.

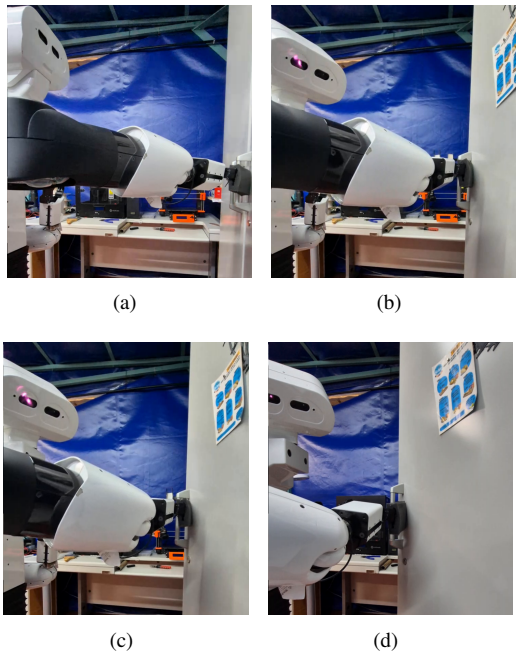
En las Figuras 7 y 8 se muestran secuencias de imágenes del robot TIAGo++ realizando las tareas de apilar bloques y abrir un armario, respectivamente, evidenciando su desempeño. Estos experimentos pueden verse en un video en línea[6].



(a) (b)

(c) (d)

Figura 7: Secuencia de ejecución de la tarea de apilado de bloques mediante la política entrenada con EXBAL-IRL: desde el fotograma inicial (a) hasta la finalización de la tarea en (d).



(a) (b)

(c) (d)

Figura 8: Secuencia de ejecución de la tarea de abrir armario mediante la política entrenada con EXBAL-IRL: desde el fotograma inicial (a) hasta la finalización de la tarea en (d).

Por otro lado, para evaluar la similitud entre el experto y la política aprendida, se calculan también distancias acumuladas entre trayectorias (Bigham and Mazaheri, 2020):

- *Dynamic Time Warping* (DTW): mide la similitud entre secuencias temporales alineándolas para minimizar la distancia acumulada, permitiendo comparar trayectorias con distintas velocidades.
- Distancia de Fréchet: conocida como distancia de la "correa del perro", evalúa cuán lejos deben recorrer dos curvas para mantenerse alineadas, capturando similitudes en la forma general de las trayectorias.
- Distancia de Hausdorff: mide la mayor distancia mínima entre puntos de dos conjuntos, proporcionando una noción de discrepancia posicional máxima entre trayectorias.

En la Tabla 2 se presenta la media y desviación típica de estas 3 distancias entre las trayectorias ejecutadas por la política con el robot real y el experto.

Tabla 2: Distancias acumulativas entre 50 trayectorias expertas y 10 trayectorias del agente por cada tarea, media y desviación estándar.

Distancia	Apilar bloques	Abrir armario
Fréchet	1,3376 ± 0,0002	1,0228 ± 0,00005
Hausdorff	1,3376 ± 0,0002	1,0077 ± 0,0027
DTW	14,3800 ± 0,1988	8,6568 ± 0,0336

Como se observa en la Tabla 2, las tareas presentan valores de distancia acumulativa notablemente bajos en las métricas

[6] Video de los experimentos disponible en `https://youtube.com/shorts/iKuQVFtCcL8?feature=share`, consultado el 29 de junio de 2025.

de Fréchet y Hausdorff, lo que indica una alta similitud espacial entre las trayectorias del agente y del experto. En particular, la tarea de abrir el armario muestra menores distancias medias en ambas métricas en comparación con la tarea de apilar bloques, reflejando un desempeño más preciso respecto al experto. En cuanto a DTW, se obtienen valores más elevados debido a su naturaleza acumulativa y dependiente del alineamiento temporal, aunque igualmente se aprecia un desempeño considerablemente mejor en la tarea de abrir armario. Estos resultados sugieren que el método propuesto logra generar políticas que reproducen trayectorias próximas a las expertas en términos espaciales y temporales.

4. Conclusiones

En este trabajo se ha logrado implementar el algoritmo *Expert-trajectory-Based features for Aprenticeship Learning via Inverse Reinforcement Learning* (EXBAL-IRL) en tareas de manipulación empleando el robot real TIAGo++. La propuesta se ha validado en dos tareas representativas, apilar bloques y abrir un armario, mostrando que el enfoque permite aprender políticas que reproducen de forma precisa el comportamiento experto a partir de demostraciones.

La combinación de características basadas en trayectorias expertas y el uso de un factor de descuento invertido ha permitido centrar el aprendizaje en las regiones más relevantes del espacio de estados, mejorando la calidad de las políticas obtenidas. Junto con el optimizador PPO, EXBAL-IRL ha demostrado ser efectivo para aprender comportamientos similares al experto, mostrando buenos resultados en tareas de manipulación evaluadas mediante métricas de distancia acumulativa.

Agradecimientos

Esta investigación ha sido financiada mediante el programa de actividades de I+D con referencia TEC-2024/TEC-62 y acrónimo iRoboCity2030-CM, concedido por la Comunidad de Madrid a través de la Dirección General de Investigación e Innovación Tecnológica a través de la Orden 5696/2024; "iREHAB" (DTS22/00105), financiado por el Instituto de Salud Carlos III; y fondos estructurales de la EU.

Referencias

Abbeel, P., Ng, A. Y., 2004. Apprenticeship learning via inverse reinforcement learning. Proceedings, Twenty-First International Conference on Machine Learning, ICML 2004, 1–8.
DOI: 10.1145/1015330.1015430

Arora, G., KiranBala, Emadifar, H., Khademi, M., 11 2023. A review of radial basis function with applications explored. Journal of the Egyptian Mathematical Society 2023 31:1 31, 1–14.
DOI: 10.1186/S42787-023-00164-3

Ashwood, Z. C., Jha, A., Pillow, J. W., 12 2022. Dynamic inverse reinforcement learning for characterizing animal behavior. Advances in Neural Information Processing Systems 35, 29663–29676.

Beliaev, M., Pedarsani, R., 2 2024. Inverse reinforcement learning by estimating expertise of demonstrators.
DOI: 10.1609/aaai.v39i15.33705

Bellman, R., 1961. Adaptive Control Processes: A Guided Tour. Princeton University Press.

Bigham, B. S., Mazaheri, S., 2020. A survey on measurement metrics for shape matching based on similarity, scaling and spatial distance. Lecture Notes on Data Engineering and Communications Technologies 45, 13–23.
DOI: 10.1007/978-3-030-37309-2$_2$

Eschmann, J., 2021. Reward function design in reinforcement learning. Studies in Computational Intelligence 883, 25–33.
DOI: 10.1007/978-3-030-41188-6$_3$/*FIGURES*/3

Fu, J., Luo, K., Levine, S., 10 2017. Learning robust rewards with adversarial inverse reinforcement learning. 6th International Conference on Learning Representations, ICLR 2018 - Conference Track Proceedings.
DOI: 10.48550/arXiv.1710.11248

Hoshino, H., Ota, K., Kanezaki, A., Yokota, R., 2022. Opirl: Sample efficient off-policy inverse reinforcement learning via distribution matching. Proceedings - IEEE International Conference on Robotics and Automation, 448–454.
DOI: 10.1109/ICRA46639.2022.9811660

Jain, A. K., 6 2010. Data clustering: 50 years beyond k-means. Pattern Recognition Letters 31, 651–666.
DOI: 10.1016/J.PATREC.2009.09.011

Jin, M., Damianouy, A., Abbeel, P., Spanos, C., 12 2017. Inverse reinforcement learning via deep gaussian process. Uncertainty in Artificial Intelligence - Proceedings of the 33rd Conference, UAI 2017.
DOI: 10.48550/arXiv.1512.08065

Kober, J., Bagnell, J. A., Peters, J., 9 2013. Reinforcement learning in robotics: A survey. International Journal of Robotics Research 32, 1238–1274.
DOI: 10.1177/0278364913495721

Likmeta, A., Metelli, A. M., Ramponi, G., Tirinzoni, A., Giuliani, M., Restelli, M., 9 2021. Dealing with multiple experts and non-stationarity in inverse reinforcement learning: an application to real-life problems. Machine Learning 110, 2541–2576.
DOI: 10.1007/S10994-020-05939-8/FIGURES/20

Naranjo-Campos, F. J., Victores, J. G., Balaguer, C., 2024. Expert-trajectory-based features for apprenticeship learning via inverse reinforcement learning for robotic manipulation. Applied Sciences 14 (23), 11131.
DOI: 10.3390/app142311131

Neu, G., Szepesvári, C., 6 2012. Apprenticeship learning using inverse reinforcement learning and gradient methods. Proceedings of the 23rd Conference on Uncertainty in Artificial Intelligence, UAI 2007, 295–302.
DOI: 10.48550/arXiv.1206.5264

Ng, A. Y., Russell, S., 2000. Algorithms for inverse reinforcement learning.

Poiani, R., Curti, G., Metelli, A. M., Restelli, M., 2024. Inverse reinforcement learning with sub-optimal experts.
DOI: 10.48550/arXiv.2401.03857

Schulman, J., Wolski, F., Dhariwal, P., Radford, A., Openai, O. K., 7 2017. Proximal policy optimization algorithms.
DOI: 10.48550/arXiv.1707.06347

Sun, J., Yu, L., Dong, P., Lu, B., Zhou, B., 4 2021. Adversarial inverse reinforcement learning with self-attention dynamics model. IEEE Robotics and Automation Letters 6, 1880–1886.
DOI: 10.1109/LRA.2021.3061397

Sutton, R. S., Barto, A. G., 2018. Reinforcement Learning: An Introduction, 2nd Edition. MIT press.

Wulfmeier, M., Ondrúška, P., Ondrúška, O., Posner, I., 7 2015. Maximum entropy deep inverse reinforcement learning.
DOI: 10.48550/arXiv.1507.04888

Zeng, S., Li, C., Garcia, A., Hong, M., 2 2023. When demonstrations meet generative world models: A maximum likelihood framework for offline inverse reinforcement learning.

Ziebart, B. D., Maas, A., Bagnell, J. A., Dey, A. K., 2008. Maximum entropy inverse reinforcement learning. In: Proceedings of the Twenty-Third AAAI Conference on Artificial Intelligence.