

---

# Ampliación funcional del robot Pololu 3Pi+ mediante la integración de un efector final de pinza para tareas de recolección y desplazamiento de objetos

---

Diana Lucia Abrego Barrera





UNIVERSIDAD DEL VALLE DE GUATEMALA  
Facultad de Ingeniería



**Ampliación funcional del robot Pololu 3Pi+ mediante la  
integración de un efector final de pinza para tareas de  
recolección y desplazamiento de objetos**

Trabajo de graduación presentado por Diana Lucia Abrego Barrera para  
optar al grado académico de Licenciada en Ingeniería Mecatrónica

Guatemala,

2025

Vo.Bo.:

(f)   
M.Sc. Miguel Enrique Zea Arenales

(f)   
M.Sc. Carlos Alberto Esquit Hernández

Fecha de aprobación: Guatemala, 28 de noviembre de 2025.

---

## Prefacio

---

En primer lugar, agradezco a Dios por permitirme vivir esta oportunidad, por protegerme en todo momento y ser mi apoyo. Su guía me dio la confianza para seguir adelante y creer que todo este proceso forma parte de sus planes.

Agradezco profundamente a mis padres por su apoyo incondicional a lo largo de este proceso. Gracias a ellos conté con las herramientas y el respaldo financiero necesarios para estudiar esta carrera, así como con una motivación constante que fue esencial para continuar y concluir este trabajo.

A mis amigos, por no dejarme solo en los momentos difíciles y por su paciencia y ánimo continuo por todo su apoyo en los momentos más difíciles y por ayudarme a sacar adelante todos los proyectos. Especialmente agradezco a Diego por todo este tiempo. Gracias a su apoyo salí adelante. Me motivó y dio confianza en mí, por escuchar todas mis quejas y penas, y por ser este año uno de los mejores. A Angi, por ser mi amiga de confianza por siempre darme palabras alentadoras y estar ahí para mí pasamos momentos de risa y estrés, pero logramos superar todo. A Cruz, por su paciencia para explicarme las cosas y aguantar mi estrés y hacerme reír en momentos de bajón. A Campos por brindarme sus consejos y sus explicaciones y darme de su tiempo para la ayudar que necesitara. A Gio, porque desde un inicio pasamos por todo esto juntos, por su apoyo en los momentos difíciles y su motivación constante para no rendirme y pensar siempre optimista. Por último, pero no menos importante, a las dos niñas Cam y Marre, que sin ellas, los días no hubieran sido tan divertidos, por incluirme en sus planes y darme su cariño las aprecio mucho. Gracias a todas las personas que no menciono pero siempre estuvieron conmigo, ya que sin ustedes mi experiencia estudiantil no hubiera sido la misma sin su compañía.

Finalmente, agradezco a los catedráticos que me brindaron su tiempo, sus consejos y su guía. A Fredy España, por sus valiosos consejos y su motivación durante todo el proyecto, me dio aliento para poder realizar mi mayor esfuerzo y se lo agradezco mucho. A mi asesor, Miguel, por apoyarme en todas las dudas y en los errores que cometí. A Luis, por su apoyo moral y el tiempo que puso para poder trabajar en el laboratorio. Sin su acompañamiento y disposición, este trabajo no habría sido posible.

<b>Prefacio</b>	<b>I</b>
<b>Índice de figuras</b>	<b>V</b>
<b>Índice de cuadros</b>	<b>VI</b>
<b>Resumen</b>	<b>VII</b>
<b>Abstract</b>	<b>VIII</b>
<b>1. Introducción</b>	<b>1</b>
<b>2. Antecedentes</b>	<b>3</b>
<b>3. Justificación</b>	<b>6</b>
<b>4. Objetivos</b>	<b>7</b>
4.1. Objetivo general . . . . .	7
4.2. Objetivos específicos . . . . .	7
<b>5. Alcance</b>	<b>8</b>
<b>6. Marco teórico</b>	<b>9</b>
6.1. Mecanismo de 4 barras . . . . .	9
6.2. Pinzas robóticas . . . . .	12
6.3. Efecto FinRay <sup>®</sup> en pinzas flexibles . . . . .	13
6.4. Robot diferencial Pololu 3Pi+ . . . . .	15
6.5. Modelando un robot diferencial . . . . .	16
6.6. Control robot diferencial . . . . .	17
<b>7. Diseño de pinza y figuras geométricas</b>	<b>20</b>
7.1. Metodología . . . . .	20
7.2. Iteraciones realizadas . . . . .	23
7.3. Resultados . . . . .	29

7.4. Discusión de resultados . . . . .	34
<b>8. Cooperación de robots para formación de letra l</b>	<b>35</b>
8.1. Flujo de información y comunicación . . . . .	36
8.2. Primera etapa de control: PID para capturar al objeto . . . . .	36
8.3. Segunda etapa: control no Lineal + planificación de trayectoria para atravesar mapa con obstáculos . . . . .	40
8.4. Tercera etapa y final: controlador PID . . . . .	43
<b>9. Conclusiones</b>	<b>47</b>
<b>10.Recomendaciones</b>	<b>49</b>
<b>11.Referencias</b>	<b>51</b>
<b>12.Anexos</b>	<b>53</b>

---

## Índice de figuras

---

1.	Robot realizado por Jose Álvarez . . . . .	3
2.	Robot de rescate T.R.U.D.I . . . . .	4
3.	Estudio de colaboración de robots móviles realizados . . . . .	5
4.	Cadena cinemática con partes principales . . . . .	10
5.	Mecanismo 4 barras y sus partes . . . . .	10
6.	Clase I de Grashof . . . . .	11
7.	Clase II y III de Grashof . . . . .	12
8.	Diferentes tipo de garras utilizados en la industria . . . . .	13
9.	Estructura del efecto FinRay <sup>®</sup> . . . . .	13
10.	Pinza robótica flexible . . . . .	14
11.	Robot diferencial Pololu 3pi+ . . . . .	15
12.	Robot diferencial Pololu 3pi+ y marco inercial . . . . .	16
13.	Modelo unicycle . . . . .	17
14.	Control no lineal de pose . . . . .	19
15.	Garra genérica . . . . .	20
16.	Síntesis gráfica mecanismo Grashof especial . . . . .	21
17.	Figuras para medir efectividad de la pinza . . . . .	22
18.	Diseño y dimensiones finales de las distintas iteraciones realizadas . . . . .	23
19.	Síntesis gráfica mecanismo iteración 1 . . . . .	24
20.	Simulación de pinza en MotionGen . . . . .	25
21.	Engranajes utilizados en la segunda pinza . . . . .	25
22.	Configuración para engranajes . . . . .	26
23.	Brazo de pinza (eslabón $L_3$ y $L_{3'}$ ) segunda iteración . . . . .	26
24.	Síntesis gráfica final para mecanismo iteración 2 . . . . .	27
25.	Desempeño físico capturando diferentes geometrías de la segunda iteración de diseño para la pinza . . . . .	27
26.	Resultado de pruebas realizadas con el cuadrado . . . . .	30
27.	Resultado de pruebas realizadas con la pirámide . . . . .	30
28.	Resultado de pruebas realizadas con la cilindro . . . . .	31
29.	Resultado de pruebas realizadas con el pentágono . . . . .	32
30.	Demostración de agarre efectivo . . . . .	32

31.	Pinza final física . . . . .	33
32.	Demostración de agarre con diferentes figuras . . . . .	34
33.	Etapas de control para la formación de la letra <i>l</i> . . . . .	36
34.	Flujo de información entre OptiTrack, Matlab, ESP32 y Pololu 3Pi+ . . . . .	36
35.	Posicionamiento de obstaculos metas y robots . . . . .	37
36.	Medición de la distancia centro del Pololu a agarre efectivo . . . . .	38
37.	Trayectoria seguida por el robot a cubo . . . . .	39
38.	Diagrama de flujo del control no lineal + planificación de trayectoria . . . . .	41
39.	Trayectoria seguida por el robot a meta final de trayectoria . . . . .	42
40.	Trayectoria a punto intermedio y formación . . . . .	44
41.	Formaciones iniciales y finales de cubos . . . . .	46

---

## Índice de cuadros

---

1.	Resultados de error de posición para distintos robots y metas . . . . .	40
2.	Errores de orientación final ( $e_{ONL}$ ) para cada robot e iteración en radianes y grados . . . . .	43
3.	Errores de posición en la formación final . . . . .	45

Guatemala es un país altamente vulnerable a desastres naturales, donde la intervención humana en escenarios de escombros puede resultar peligrosa. En este contexto, los manipuladores móviles permiten realizar tareas de inspección y transporte de objetos de forma más segura. El robot diferencial Pololu 3Pi+, utilizado dentro del ecosistema Robotat de la Universidad del Valle de Guatemala, ha sido empleado principalmente para experimentos de navegación, pero carece de un efector final que le permita manipular objetos, lo que limita su aplicación en tareas de recolección y desplazamiento. Este trabajo tiene como objetivo ampliar la funcionalidad del Pololu 3Pi+ mediante el diseño e integración de una pinza robótica ligera y compacta, capaz de sujetar distintas figuras geométricas durante el movimiento, y la implementación de una arquitectura de control para coordinar varios robots en una tarea cooperativa. La metodología incluye el diseño iterativo de la pinza, desde configuraciones rígidas en PLA hasta una geometría final en TPU, la selección de piezas de prueba y la evaluación experimental de la efectividad de agarre. Para validar la tarea cooperativa, varios robots transportan cubos y forman una figura con forma de letra "l". La navegación se organiza en tres etapas: aproximación al objeto con control PID, desplazamiento con un controlador no lineal y planificación de trayectoria, y posicionamiento final usando nuevamente control PID. Los resultados muestran una efectividad de agarre cercana al 87.5% y errores de posición y orientación que permiten formar la letra "l" de manera precisa.

**Palabras clave:** robótica móvil, manipulador móvil, pinza robótica, Pololu 3pi+, cooperación entre robots, controladores.

Guatemala is a country highly vulnerable to natural disasters, where human intervention in rubble scenarios can be hazardous. In this context, mobile manipulators enable safer execution of inspection tasks and object transportation. The Pololu 3Pi+ differential-drive robot, used within the Robotat ecosystem at Universidad del Valle de Guatemala, has been employed mainly for navigation experiments but lacks an end-effector capable of manipulating objects, which limits its application in collection and transport tasks.

This work aims to expand the functionality of the Pololu 3Pi+ through the design and integration of a lightweight and compact robotic gripper capable of grasping different geometric shapes while in motion, as well as the implementation of a control architecture to coordinate multiple robots in a cooperative task. The methodology includes an iterative design process for the gripper, progressing from rigid PLA configurations to a final TPU geometry, the selection of test objects, and the experimental evaluation of gripping effectiveness.

To validate the cooperative task, several robots transport cubes and assemble a figure in the shape of the letter “L”. Navigation is organized into three stages: object approach using PID control, displacement with a nonlinear controller and trajectory planning, and final positioning again using PID control. The results show a gripping effectiveness of approximately 87.5 % and position and orientation errors that still allow the robots to accurately form the letter “l”.

**Keywords:** mobile robotics, mobile manipulator, robotic gripper, Pololu 3Pi+, multi-robot cooperation, controllers.

En las últimas décadas, el desarrollo de la robótica móvil ha tenido un impacto significativo en áreas como la exploración, la logística y la asistencia en situaciones de emergencia. En entornos donde las condiciones son peligrosas o de difícil acceso para los seres humanos, los robots representan una alternativa segura y eficiente para realizar tareas de inspección, transporte e interacción con objetos. En países como Guatemala, caracterizados por una alta vulnerabilidad frente a desastres naturales, especialmente sismos, resulta pertinente explorar soluciones tecnológicas que contribuyan a la mitigación de riesgos y al apoyo en los procesos de búsqueda, rescate y remoción de escombros. En este contexto han surgido robots removedores de escombros, diseñados para manipular y transportar objetos pesados de forma autónoma, reduciendo la exposición del personal humano a situaciones de alto riesgo y ofreciendo una solución segura y eficiente.

Dentro de este contexto, el ecosistema Robotat de la Universidad del Valle de Guatemala proporciona una plataforma experimental para el estudio de navegación, control y localización de agentes móviles. Entre estos agentes destaca el Pololu 3Pi+, un robot diferencial compacto que ha sido utilizado principalmente para tareas de seguimiento de trayectorias y experimentos de control de movimiento. Sin embargo, en su configuración actual, el Pololu 3Pi+ es incapaz de realizar manipulación activa de objetos: carece de un efector final que le permita sujetar, trasladar y colocar piezas de manera controlada.

El objetivo de este trabajo es extender la funcionalidad del Pololu 3Pi+ mediante la integración de un efector final en forma de pinza robótica y el desarrollo de un esquema de control que permita ejecutar tareas de recolección y desplazamiento de objetos, tomando como inspiración las aplicaciones de los robots removedores de escombros que realizan tareas similares. En particular, se busca diseñar una pinza ligera y compacta, compatible con las restricciones geométricas del robot, capaz de sujetar objetos con geometrías variadas sin pérdida durante movimientos dinámicos, y evaluar su desempeño en un escenario cooperativo de formación de figuras. El éxito del diseño se medirá a partir de métricas cuantitativas de desempeño, tales como la efectividad del agarre sobre diferentes piezas (porcentaje de intentos exitosos sin pérdida del objeto) y la precisión en la colocación final de los cubos

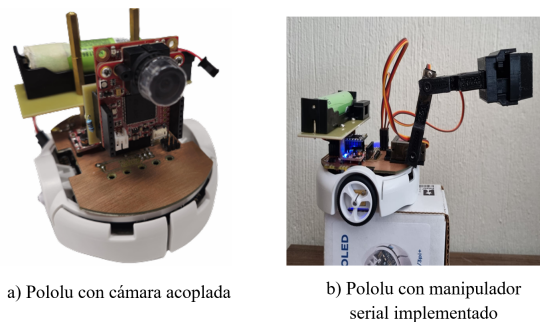
dentro de una configuración geométrica objetivo.

Para alcanzar este propósito, se propone un enfoque que combina el diseño mecánico de la pinza con el desarrollo de estrategias de control para el robot diferencial. Desde el punto de vista mecánico, se plantea un mecanismo de cuatro barras accionado por un servomotor y se exploran distintas geometrías de dedos, incluyendo variantes inspiradas en el efecto FinRay<sup>®</sup>, con el fin de mejorar la adaptación a diferentes formas. En paralelo, se integran algoritmos de control de movimiento, como control PID y control no lineal con planificación de trayectoria, aprovechando la información de posición proporcionada por el sistema de captura de movimiento del Robotat. Finalmente, se implementan pruebas de manipulación y cooperación entre varios robots para validar experimentalmente la solución propuesta.

En este informe se presenta, en primer lugar, el contexto en el que se enmarca el proyecto, así como los antecedentes y trabajos relacionados que motivan el desarrollo de una pinza para el Pololu 3Pi+. A continuación, se describen los objetivos planteados, el alcance y las principales restricciones del sistema propuesto. Posteriormente, se introduce el marco teórico necesario para comprender tanto el diseño mecánico de la pinza como los algoritmos de control utilizados, y se detalla el proceso de diseño, construcción y validación experimental del prototipo. Finalmente, se discuten los resultados obtenidos, se extraen las conclusiones más relevantes y se plantean posibles líneas de trabajo futuro que permitan ampliar las capacidades de manipulación y navegación del robot.

El Robotat de la Universidad del Valle de Guatemala (UVG) es un ecosistema avanzado para la experimentación y validación de prototipos y algoritmos de robótica, ubicado en el laboratorio CIT-116. Esta plataforma opera mediante un sistema de captura de movimiento denominado OptiTrack, que permite la comunicación con agentes robóticos a través de una red inalámbrica *wifi* basada en el protocolo TCP/IP, facilitando así la adquisición de datos y la interacción en tiempo real con los agentes [1]. Dentro del sistema Robotat se pueden emplear diversos agentes robóticos, entre los que destaca el Pololu 3Pi+, un robot diferencial compacto diseñado para la exploración, navegación y resolución de problemas, mediante sensores como giroscopios y capacidades de comunicación inalámbrica. Su diseño ligero y su capacidad para desplazarse con precisión lo convierten en una herramienta ideal para pruebas y desarrollos dentro del ecosistema Robotat.

**Figura 1.** Robot realizado por Jose Álvarez



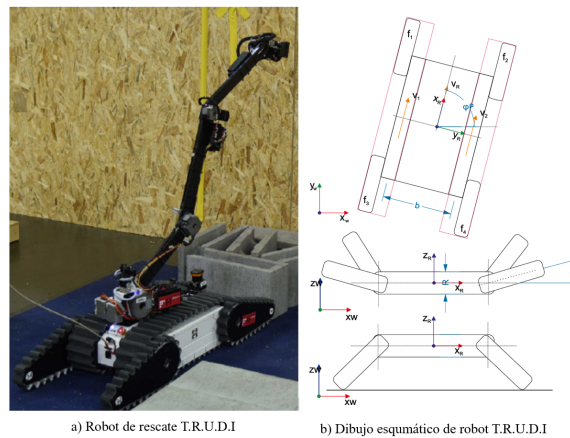
Nota. Diseño final de robot Pololu 3pi+ realizado por José Álvarez equipado con la cámara OPENCam H7 (a) y Manipulador serial de tres grados de libertad (b)

Con el objetivo de expandir sus aplicaciones y capacidades, en 2024 José Álvarez desarrolló una investigación [2] que proponía una mejora funcional al Pololu 3Pi+ mediante una

placa de expansión que integra el TinyS3, la OpenMV Cam H7 y un manipulador serial. Uno de los resultados obtenidos fue que el manipulador serial, dotado de tres grados de libertad, alcanzó una precisión de movimiento de  $0.1^\circ$ . Sin embargo, una limitación identificada fue que se debe remover el módulo de la cámara OpenMV para ensamblar el manipulador. Como se puede observar en la Figura 1, no se puede hacer uso simultáneo de ambos componentes ya que el manipulador es grande y obstaculiza la visión de la cámara.

Un referente en la implementación de manipuladores móviles es T.R.U.D.I. [3], un robot desarrollado para la competencia RoboCup y diseñado específicamente para la remoción de escombros en entornos adversos. Este robot combina una plataforma móvil con un manipulador serial de 12 grados de libertad, diseñado para interactuar con su entorno con alta precisión. Su estructura robusta le permite manipular objetos pesados y voluminosos, y cuenta con dos orugas principales más cuatro extremidades adicionales como se muestra en la Figura 2. Estas le otorgan estabilidad en terrenos irregulares, como en edificios colapsados o en algún tipo de desastre natural. No obstante, esta robustez conlleva limitaciones adicionales en términos de velocidad, fuerza y par del manipulador, además de que se trata de un sistema único que debe realizar todas las tareas de remoción por sí solo, sin considerar opciones de cooperación entre múltiples robots lo que incrementa la complejidad y el costo del sistema.

**Figura 2.** Robot de rescate T.R.U.D.I



Nota. Robot de rescate T.R.U.D.I y su esquemático que ejemplifica dónde se encuentra su centro de masa y fuerzas y velocidades involucradas

Como un ejemplo de manipuladores móviles que trabajan en colaboración se encuentra el estudio realizado por [4], que muestra una alternativa prometedora basada en la cooperación entre manipuladores móviles. En esta investigación, dos robots colaboran para trasladar un objeto dentro de un entorno con obstáculos (Figura 3). Se diseñó un algoritmo capaz de calcular trayectorias factibles que, en caso de no existir, detienen el movimiento, garantizando así la seguridad del sistema y la convergencia hacia la configuración deseada.

La comunicación entre robots se realiza mediante *lean communication*, una ley de control de bajo nivel que minimiza la necesidad de intercambio constante de información, reduciendo así la carga de procesamiento y aumentando la eficiencia del sistema. Cabe resaltar que este estudio fue validado únicamente mediante simulaciones. Los autores proponen implementarlo



Guatemala es un país sísmicamente activo debido a su ubicación geográfica, lo cual genera desastres naturales que presentan escenarios peligrosos y complejos para los equipos de rescate. En estas situaciones, los escombros voluminosos y pesados dificultan las labores de auxilio, por lo que se requiere maquinaria especializada para su remoción. Una alternativa prometedora frente a este desafío es el uso de manipuladores móviles, capaces de remover escombros de forma controlada.

En una investigación anterior, José Álvarez integró un micro brazo robótico al Pololu 3Pi+ con el objetivo de crear un manipulador móvil capaz de recoger objetos. No obstante, el sistema desarrollado, con tres grados de libertad, resultó ser demasiado voluminoso y complejo para las tareas planteadas. Esto evidenció la necesidad de diseñar un manipulador más ligero y simple, que se ajustara mejor al robot.

La propuesta actual plantea la incorporación de una garra robótica con un solo grado de libertad al Pololu 3Pi+, creando un manipulador móvil que pueda ejecutar tareas de recolección y ubicación de objetos en puntos específicos con precisión. Una ventaja considerable de este proyecto es la posibilidad de realizar pruebas dentro del ecosistema Robotat. Esto posibilitará la creación de entornos físicos personalizados que simule escenarios de desastre, los cuales serían muy difíciles de replicar en condiciones reales, permitiendo así examinar, validar y perfeccionar el desempeño del manipulador móvil en las diferentes situaciones propuestas. Adicionalmente, se considerará la cooperación de manipulación de objetos entre varios manipuladores móviles, permitiendo distribuir tareas y aumentar la eficiencia operativa. Al combinar esfuerzos, será posible llevar a cabo operaciones más complejas sin depender de un solo robot robusto y complejo.

Con estas aplicaciones, se busca ampliar las capacidades del Pololu 3Pi+ dentro del ecosistema Robotat, brindándole un nuevo enfoque como manipulador móvil orientado a la remoción de escombros, y fortaleciendo su utilidad en tareas de manipulación de objetos en entornos físicos.

### 4.1. Objetivo general

Diseñar un robot manipulador móvil que utilice el Pololu 3Pi+ como base móvil y una garra como efector final, capaz de recoger y trasladar objetos de un lugar a otro con precisión dentro del ecosistema Robotat.

### 4.2. Objetivos específicos

- Diseñar y construir un efector final en forma de pinza con un grado de libertad, para su integración en el robot Pololu 3Pi+.
- Integrar el sistema de captura de movimiento OptiTrack del ecosistema Robotat para poder ejecutar tareas de manipulación móvil punto a punto.
- Validar la aplicación de múltiples manipuladores móviles robóticos para poder realizar tareas cooperativas.

El presente proyecto se centra en el diseño, construcción y validación de una pinza robótica adaptable al robot Pololu 3Pi+ y en su aplicación a tareas de manipulación y cooperación entre múltiples agentes móviles. El alcance del trabajo se limita a la manipulación de objetos con dimensiones y geometrías específicas, de modo que puedan ser tomados de forma fiable por la pinza propuesta; no se contempla la manipulación de objetos fuera de este rango dimensional. La pinza fue diseñada para asegurar la compatibilidad mecánica con el robot, lo que incluye el uso del servomotor FT90M, ya que la geometría de la pinza está optimizada para acoplarse a las dimensiones de dicho servomotor. Asimismo, se consideró un único método de sujeción al robot, mediante la placa pcb que se encuentra conectada al mismo, por lo que, en caso de requerir su montaje en otra zona del robot o en otra plataforma, sería necesario un rediseño mecánico. La garra cuenta únicamente con un grado de libertad, por lo que su funcionalidad se restringe a las acciones de capturar y soltar objetos, sin considerar tareas de orientación más complejas.

Todos los experimentos se realizaron dentro del laboratorio Robotat, bajo disponibilidad de tiempo y configuraciones predeterminadas de la red. Como consecuencia de ello, el número de pruebas se vio limitado, así como los escenarios que pudieron evaluarse. Debido a estas restricciones, en este proyecto no se incluyó la evaluación de controladores más complejos ni la optimización de la velocidad de manipulación; en su lugar, se consideró únicamente el desempeño de un controlador PID y de un controlador no lineal, priorizando la estabilidad y la convergencia por encima de la rapidez o la eficiencia extrema, con el objetivo de garantizar un desempeño confiable en la manipulación del cubo dentro de las condiciones de prueba. De igual forma, el uso de los cubos se restringió a la formación de figuras compuestas, sin explorar otras formas. Finalmente, la cooperación entre robots se planteó de manera no simultánea; los tres robots no operan coordinadamente al mismo tiempo, sino que se emplean como múltiples agentes para demostrar el desempeño de la pinza y del esquema de control en distintos robots, sin abordar estrategias avanzadas de coordinación cooperativa entre ellos.

En este capítulo se presentan los fundamentos teóricos necesarios para el desarrollo del proyecto. En primer lugar, se describen los conceptos básicos de mecanismos, con énfasis en el mecanismo de cuatro barras, abordando la síntesis gráfica, la determinación de grados de libertad y la clasificación según la condición de Grashof, con el fin de sustentar el diseño cinemático de la garra. Posteriormente, se revisan los tipos de pinzas robóticas y se introduce el efecto FinRay<sup>®</sup> como principio de diseño para pinzas flexibles. Se destacan sus ventajas en términos de adaptación geométrica y aumento del área de contacto. A continuación, se presenta el robot diferencial Pololu 3Pi+ y su modelado cinemático. Iniciando desde la perspectiva del robot diferencial y luego mediante el modelo de unicycle. Con ello se puede formular las leyes de control del robot. Se incluye la cinemática diferencial inversa y el controlador no lineal para el modelo de unicycle, que serán empleados en las etapas de implementación y validación experimental del sistema.

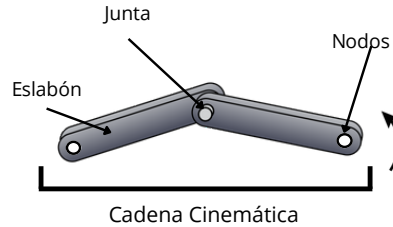
## 6.1. Mecanismo de 4 barras

Un mecanismo es un conjunto de elementos conectados entre sí con el objetivo de transmitir movimiento de forma controlada y predecible. Estos elementos incluyen los eslabones, que son cuerpos rígidos con al menos dos nodos de conexión hacia otros eslabones; las juntas, que permiten el movimiento relativo entre dos o más eslabones; y las cadenas cinemáticas, entendidas como el ensamble de eslabones y juntas dispuesto para producir una salida en respuesta a una entrada. Cuando al menos uno de los eslabones de una cadena cinemática se mantiene fijo respecto al sistema de referencia, dicha estructura recibe el nombre de mecanismo [5]. La Figura 4 ilustra visualmente estos elementos y muestra cómo se organizan para conformar una cadena cinemática básica.

Dentro de esta clasificación, uno de los mecanismos más estudiados es el mecanismo de cuatro barras, el cual está formado por cuatro eslabones conectados mediante juntas rotacionales. Entre ellos destacan tres componentes fundamentales para su comportamiento

cinemático: la manivela, que ejecuta una rotación completa y se encuentra unida al eslabón fijo; el balancín, que realiza un movimiento oscilatorio; y la tierra, que corresponde al eslabón inmóvil del sistema y sirve como referencia para el movimiento de los demás. En la Figura 5 se presentan estas partes principales, junto con una representación gráfica de la síntesis del mecanismo de cuatro barras.

**Figura 4.** Cadena cinemática con partes principales

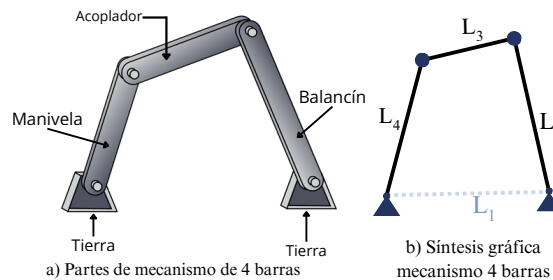


Nota. La imagen muestra las partes principales que conforman una cadena cinemática. Un mecanismo está conformado por un conjunto de cadenas cinemáticas. Elaboración propia.

### 6.1.1. Síntesis gráfica

La síntesis gráfica es una técnica empleada para representar y analizar el comportamiento de un mecanismo mediante la disposición geométrica de sus eslabones y juntas. En este tipo de representación, los eslabones se trazan como líneas y las juntas como puntos, lo que facilita visualizar la estructura y el movimiento relativo entre cada componente. Para su identificación, los eslabones se nombran convencionalmente como  $L_1$ ,  $L_2$ ,  $L_3$  y  $L_4$ : la tierra corresponde a  $L_1$ , la manivela a  $L_2$ , el acoplador a  $L_3$  y el balancín a  $L_4$ . El eslabón fijo  $L_1$  es aquel al que se encuentran conectados los demás, y en él puede instalarse un motor u otro actuador encargado de accionar el mecanismo [5]. Esta representación puede observarse en la Figura 5.

**Figura 5.** Mecanismo 4 barras y sus partes



Nota. En a) se muestra las partes principales que conforman un mecanismo de 4 barras(eslabones). En b) la síntesis gráfica del mismo, se puede observar que  $L_1$  está representado con una línea punteada con fin ilustrativo para mostrar que la tierra es la misma en los dos eslabones. Elaboración propia.

### 6.1.2. Determinación de los grados de libertad

Los grados de libertad (GDL) representan el número de entradas independientes necesarias para lograr el movimiento deseado de un mecanismo. Se calcula mediante la ecuación de Kutzbach (una modificación de la fórmula de Gruebler) [5]:

$$M = 3(L - 1) - 2J_1 - J_2, \quad (1)$$

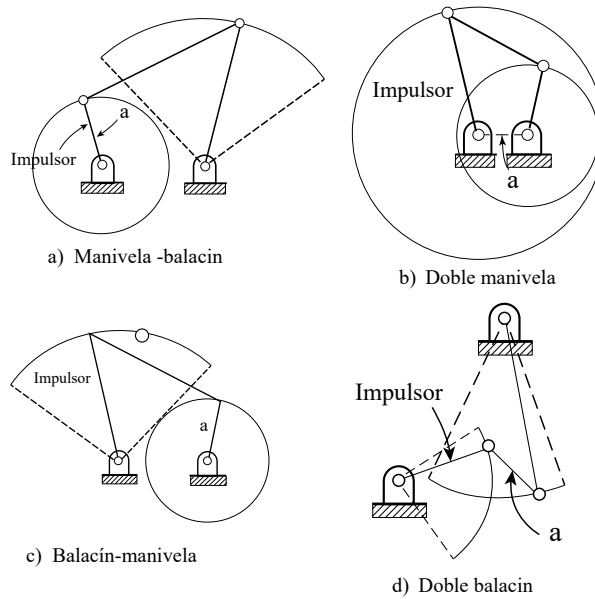
donde  $M$  es el grado de libertad,  $L$  es el número de eslabones,  $J_1$  es el número de juntas con un GDL (pares cinemáticos completos) y  $J_2$  es el número de juntas con dos GDL (pares cinemáticos parciales).

La condición de Grashof permite predecir si un mecanismo de cuatro barras puede realizar rotaciones completas, dependiendo únicamente de las longitudes de sus eslabones. Esta condición se expresa mediante:

$$S + L \leq P + Q, \quad (2)$$

donde  $S$  es el eslabón más corto,  $L$  es el eslabón más largo, y  $P$  y  $Q$  son los dos eslabones restantes.

**Figura 6.** Clase I de Grashof

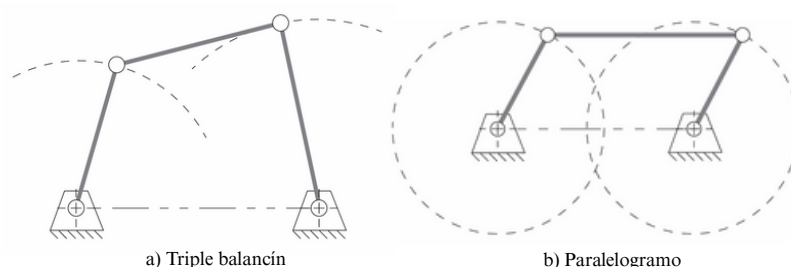


Nota. Se muestran las diferentes configuraciones fijando las tierras en diferentes eslabones, se indica cuál es el que impulsa el mecanismo que actúa como manivela. Elaboración propia.

Según esta condición:

- **Clase I** ( $S + L < P + Q$ ): dependiendo del eslabón que se fije a la tierra será el tipo. Si se fija cualquier eslabón adyacente al más corto se obtiene un mecanismo doble manivela. Si se fija el eslabón más corto se obtiene un manivela-balancín, o si se fija el eslabón opuesto al más corto se obtiene un doble balancín. En la Figura 6 se muestra la síntesis gráfica de todas las configuraciones posibles. Entonces, si esta condición se cumple el acoplador realiza una revolución completa. Si esta no se cumple entonces el eslabonamiento no es de Grashoff [5].
- **Clase II** ( $S + L > P + Q$ ): no importa cuál eslabón se fije a la tierra todas serán balancines triples, en los que ningún eslabón puede girar por completo como se muestra en la Figura 7 a) [5].
- **Clase III** ( $S + L = P + Q$ ): también conocido como caso especial de Grashof o cadena cinemática Clase III, no importa cuál eslabón se fije todas las configuraciones serán dobles-manivelas o manivela-balancín. Como ejemplo están la configuración de paralelogramo el cuál se muestra en la Figura 7 b). Este es útil ya que duplica con exactitud el movimiento rotatorio de la manivela impulsora a la impulsada [5].

**Figura 7.** Clase II y III de Grashof



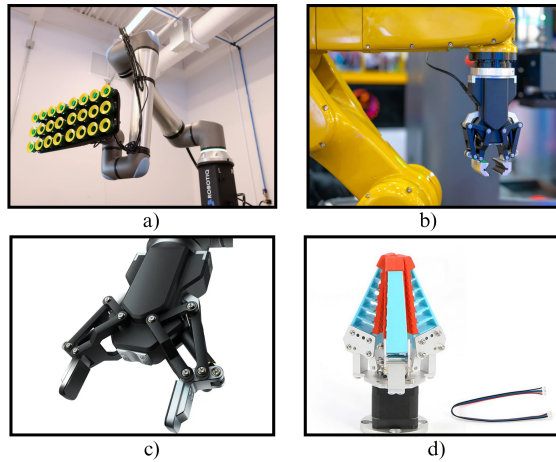
Nota. En a) se muestra un ejemplo de un triple balancín de clase II, en b) el ejemplo de la configuración de paralelogramo de clase III. Elaboración propia.

## 6.2. Pinzas robóticas

Una pinza robótica o garra es un efector final diseñado para sujetar, manipular o transportar objetos. Este elemento se acopla al extremo de cualquier agente robótico y constituye el punto de interacción con el entorno físico [6].

Existen diversos tipos de pinzas, entre ellas: de ventosas, neumáticas o hidráulicas, servo eléctricas (*Servo-Electric Gripper*) y flexibles [7].

**Figura 8.** Diferentes tipo de garras utilizados en la industria

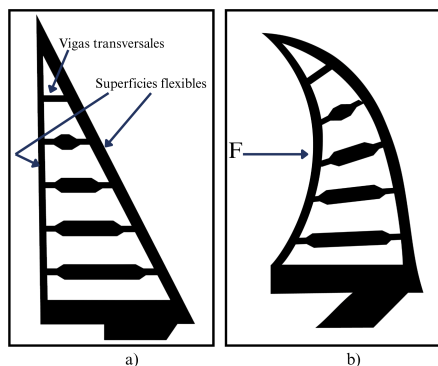


Nota. a) garra con ventosas, b) garras neumática c) garra servo eléctrica ,d) garra flexible. Elaboración propia.

### 6.3. Efecto FinRay<sup>®</sup> en pinzas flexibles

El efecto FinRay<sup>®</sup> se inspira en el comportamiento biomecánico de las aletas de los peces. Fue descrito por Leif Kniese, quien observó que, al aplicar una fuerza sobre una aleta, esta no colapsa hacia atrás, sino que se deforma hacia la fuente de la fuerza, lo cual permite un contacto eficiente y estable [8]. Mecánicamente, una estructura FinRay<sup>®</sup> está compuesta por dos superficies flexibles dispuestas en forma de “V” o “A” y vigas transversales que conectan ambos lados.

**Figura 9.** Estructura del efecto FinRay<sup>®</sup>



Nota. Efecto FinRay<sup>®</sup> y deformación al aplicarse una fuerza. Elaboración propia.

Esta configuración provoca una deformación en la dirección de la fuerza aplicada, lo que permite aumentar el área de contacto sin colapsar la estructura. En la Figura 9 se observa que, al aplicar una fuerza, la estructura se deforma adaptándose a la geometría del objeto.

Desde el punto de vista del agarre, la fuerza máxima que puede soportar una pinza antes de que ocurra el deslizamiento puede expresarse, de forma simplificada, como

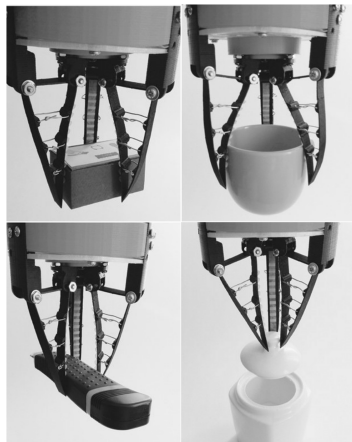
$$F_g \approx \mu p_{eq} A_c, \quad (3)$$

donde  $F_g$  es la fuerza de agarre,  $\mu$  es el coeficiente de fricción entre el dedo y el objeto,  $p_{eq}$  es una presión de contacto equivalente y  $A_c$  es el área efectiva de contacto. En una pinza rígida convencional,  $A_c$  suele ser reducido porque los dedos solo apoyan en unas pocas zonas. En cambio, en un dedo tipo FinRay, la deformación inducida por las vigas transversales, especialmente cuando estas se inclinan hacia la superficie de contacto [9], hace que el dedo envuelva el objeto, incrementando  $A_c$  para la misma fuerza de actuación. De este modo, el efecto FinRay<sup>®</sup> permite generar mayores fuerzas de agarre  $F_g$  sin incrementar de forma significativa la presión local sobre el objeto. Lo que se traduce en una sujeción más estable y en una menor probabilidad de dañar objetos delicados en comparación con pinzas rígidas o con dedos de geometría no adaptativa [8], [10].

En las pinzas flexibles, este efecto ha demostrado ser una solución efectiva frente a las limitaciones de las pinzas tradicionales. Estas no suelen adaptarse adecuadamente a geometrías variadas ni resultan apropiadas para manipular elementos pequeños o frágiles. La estructura FinRay<sup>®</sup>, al ser adaptativa y aumentar  $A_c$  de forma pasiva, permite una interacción más segura y eficaz en aplicaciones de manipulación robótica. El desempeño de la pinza frente a diferentes figuras se puede observar en la Figura 10.

Estas pinzas suelen fabricarse con materiales flexibles como el poliuretano termoplástico (TPU), el cual posee una dureza *Shore* de 95A y una alta capacidad de recuperación elástica, haciéndolo ideal para estructuras que deben deformarse sin perder su funcionalidad. El TPU, utilizado en procesos de impresión 3D, proporciona un equilibrio entre suavidad y resistencia mecánica. Por ello, resulta una opción adecuada para dedos robóticos que requieren adaptabilidad y durabilidad [11].

**Figura 10.** Pinza robótica flexible



Nota. Se muestra una garra flexible que fue diseñada utilizando el efecto FinRay<sup>®</sup>, demuestra la versatilidad para tomar diferentes objetos. Elaborado por [11].

## 6.4. Robot diferencial Pololu 3Pi+

Un robot diferencial es un tipo de robot móvil que se desplaza utilizando dos ruedas. Cada rueda puede ser controlada de forma independiente, permitiendo que el robot se desplace en línea recta, gire o incluso rote sobre su propio eje. Para mejorar la estabilidad, estos robots suelen incorporar una o más ruedas locas o de arrastre [12].

El Pololu 3pi+ es un robot diferencial compacto de aproximadamente 9.7 cm de diámetro, programable por el usuario y basado en un microcontrolador ATmega32U4 compatible con Arduino. Este robot implementa una arquitectura de tracción diferencial: cuenta con dos ruedas laterales, cada una impulsada por un motor DC independiente, lo que le permite ejecutar maniobras de giro mediante la diferencia de velocidades entre ambas ruedas [13].

Los robots diferenciales presentan un comportamiento no holonómico, es decir, su movimiento está sujeto a dos restricciones que se expresan directamente sobre las velocidades. Estas son:

- **Restricción de no deslizamiento lateral.** Un robot diferencial no puede desplazarse lateralmente de manera directa. Su movimiento se limita a avanzar o retroceder en la dirección de su orientación, junto con el cambio de orientación angular alrededor de su propio eje. Esto implica que su velocidad lateral instantánea es siempre cero [14]. Matemáticamente, esta restricción se expresa como:

$$\dot{x} \sin \theta - \dot{y} \cos \theta = 0, \quad (4)$$

donde  $\dot{x}$  y  $\dot{y}$  son las componentes de la velocidad del robot en el marco inercial y  $\theta$  es el ángulo de orientación del robot.

- **Restricción no integrable (no holonómica).** La restricción anterior depende de las velocidades  $\dot{x}$ ,  $\dot{y}$  y  $\dot{\theta}$ , y no puede integrarse para obtener una relación algebraica que involucre solo las coordenadas de posición  $(x, y, \theta)$ . Es decir, no existe una función  $f(x, y, \theta) = 0$  que describa dicha restricción de forma exacta. Por ello, el sistema se clasifica como no holonómico y su análisis se realiza mediante cinemática diferencial.

**Figura 11.** Robot diferencial Pololu 3pi+

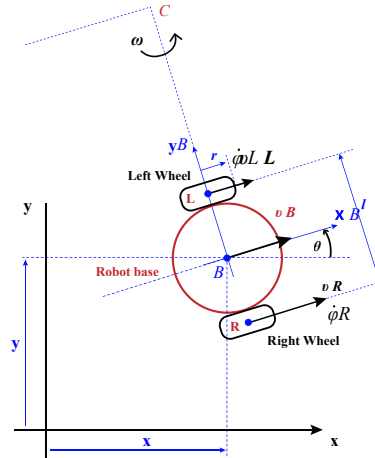


Nota. Vista frontal del robot Pololu 3pi+ elaborada por [15].

## 6.5. Modelando un robot diferencial

El comportamiento de un robot diferencial puede describirse mediante un modelo cinemático, el cual relaciona la configuración del robot en el espacio con las variables que se pueden medir o accionar directamente. En particular, la pose del robot en el plano se describe mediante las coordenadas  $(x, y)$  de su posición y el ángulo  $\theta$  de orientación respecto al marco inercial. En la Figura 12 se muestran los marcos de referencia global y del robot, así como las principales variables utilizadas para el análisis cinemático.

**Figura 12.** Robot diferencial Pololu 3pi+ y marco inercial



Nota. Se muestran el marco de referencia inercial y el marco de referencia del robot diferencial, indicando las velocidades y la posición. Elaboración propia.

### 6.5.1. Cinemática diferencial

La cinemática diferencial describe cómo las velocidades de las ruedas del robot se relacionan con la variación de su pose en el marco inercial, es decir, con sus velocidades lineales y angular  $(\dot{x}, \dot{y}, \dot{\theta})$ . En el caso de un robot diferencial, este modelo permite calcular el movimiento del robot a partir de las velocidades angulares de las ruedas  $(\dot{\varphi}_R, \dot{\varphi}_L)$ , estableciendo la relación entre las acciones de los actuadores y el comportamiento global del robot en el espacio  $(x, y, \theta)$  [14]. El modelo matemático se describe como:

$$\begin{aligned}\dot{x} &= \frac{r}{2}(\dot{\varphi}_R + \dot{\varphi}_L) \cos \theta, \\ \dot{y} &= \frac{r}{2}(\dot{\varphi}_R + \dot{\varphi}_L) \sin \theta, \\ \dot{\theta} &= \frac{r}{2l}(\dot{\varphi}_R - \dot{\varphi}_L),\end{aligned}\tag{5}$$

donde  $r$  es el radio de las ruedas y  $l$  es la distancia entre ellas.

## 6.6. Control robot diferencial

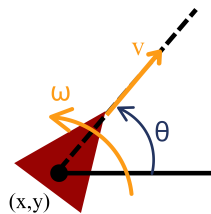
Para diseñar el control de un robot diferencial resulta conveniente utilizar el modelo de unicycle como representación intermedia. En este esquema, el controlador se diseña para generar comandos de velocidad lineal y angular ( $v_{ctrl}, \omega_{ctrl}$ ) sobre el modelo de unicycle, y posteriormente estos comandos se transforman mediante la cinemática inversa en velocidades angulares de las ruedas  $\dot{\varphi}_R$  y  $\dot{\varphi}_L$ . De este modo se separa el problema de control en dos niveles: uno asociado al movimiento del robot en el plano y otro relacionado con su implementación física mediante las ruedas.

### 6.6.1. Modelo unicycle

El modelo de unicycle es una representación simplificada de un robot diferencial que captura su movimiento en el plano. En este modelo, el estado del robot se describe por su posición  $(x, y)$  y su orientación  $\theta$  respecto al marco inercial, mientras que las entradas de control son la velocidad lineal  $v$  y la velocidad angular  $\omega$ . La velocidad lineal  $v$  representa la capacidad del robot para avanzar o retroceder, y la velocidad angular  $\omega$  describe su capacidad para girar sobre su eje, tal como se ilustra en la Figura 13. Este modelo comparte las mismas restricciones de rodamiento sin deslizamiento que el robot diferencial [14].

Trabajar con el modelo de unicycle ofrece una ventaja importante desde el punto de vista del control: permite diseñar leyes de control en términos de las variables  $v$  y  $\omega$ , independientes de la configuración física específica del robot. De esta forma, el controlador se formula a un nivel más abstracto, operando sobre el movimiento global de avance y giro del robot. Posteriormente, las velocidades de control ( $v_{ctrl}, \omega_{ctrl}$ ) se traducen mediante las ecuaciones de cinemática inversa a las velocidades de las ruedas ( $\dot{\varphi}_R, \dot{\varphi}_L$ ), que son las señales finalmente aplicadas a los actuadores [16].

**Figura 13.** Modelo unicycle



Nota. Modelo unicycle: se muestran la velocidad angular  $\omega$ , la velocidad lineal  $v$ , la posición  $x, y$  y la orientación  $\theta$ . Elaborado por [17].

El modelo cinemático del unicycle se expresa como

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \end{bmatrix} = \begin{bmatrix} \cos \theta & 0 \\ \sin \theta & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} v \\ \omega \end{bmatrix}.$$

## Cinemática diferencial inversa

Se utiliza la inversa de la cinemática diferencial del robot diferencial y así obtener velocidades angulares controladas :

$$\begin{aligned}\dot{\varphi}_{R,\text{ctrl}} &= \frac{v_{\text{ctrl}} + l\omega_{\text{ctrl}}}{r} \\ \dot{\varphi}_{L,\text{ctrl}} &= \frac{v_{\text{ctrl}} - l\omega_{\text{ctrl}}}{r}\end{aligned}$$

dónde:

- $\dot{\varphi}_{R,\text{ctrl}}$ ; velocidad angular controlada de la rueda derecha [rad/s]
- $\dot{\varphi}_{L,\text{ctrl}}$ ; velocidad angular controlada de la rueda izquierda [rad/s]
- $v_{\text{ctrl}}$ ; velocidad lineal deseada del robot [m/s]
- $\omega_{\text{ctrl}}$ ; velocidad angular deseada del robot [rad/s]
- $r$ ; radio de las ruedas [m]
- $l$ ; semidistancia entre ruedas (desde el centro a cada rueda) [m]

## Controlador no lineal para el unicycle

Esta técnica de control aborda directamente las no linealidades del sistema, sin recurrir a linealizaciones aproximadas. Para ello, se introduce un cambio de coordenadas a un sistema polar relativo, en el cual el error de pose se expresa mediante las variables  $\rho$ ,  $\alpha$  y  $\beta$ . En la Figura 14 se ilustra el significado geométrico de estas coordenadas y el origen de cada una de ellas. Las transformaciones se definen como:

$$\rho = \sqrt{\Delta x^2 + \Delta y^2}, \quad (6)$$

$$\alpha = \text{atan2}(\Delta y, \Delta x) - \theta, \quad (7)$$

$$\beta = -\theta - \alpha. \quad (8)$$

En este marco, el objetivo de control consiste en llevar al robot desde su pose inicial hasta la pose objetivo, lo cual equivale a hacer que el estado del sistema converja al origen  $(\rho, \alpha, \beta) = (0, 0, 0)$ . Para lograrlo, se busca definir  $\nu$  y  $\omega$  de manera que la distancia al objetivo  $\rho$  disminuya y, al mismo tiempo, la orientación del robot se alinee con la orientación deseada, descrita por los ángulos  $\alpha$  y  $\beta$ .

Si se restringe el ángulo  $\alpha$  al intervalo  $\alpha \in (-\frac{\pi}{2}, \frac{\pi}{2})$  y se redefine la velocidad lineal como  $v = -\nu$ , es posible reescribir la cinemática del modelo de unicycle en las coordenadas  $(\rho, \alpha, \beta)$ . La dinámica del error adopta entonces la forma:

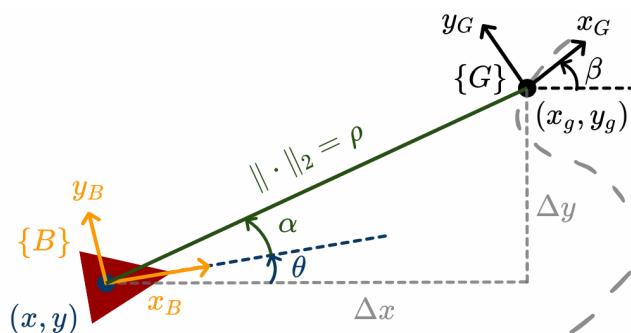
$$\begin{bmatrix} \dot{\rho} \\ \dot{\alpha} \\ \dot{\beta} \end{bmatrix} = \begin{bmatrix} \cos \alpha & 0 \\ -\frac{\sin \alpha}{\rho} & 1 \\ \frac{\sin \alpha}{\rho} & 0 \end{bmatrix} \begin{bmatrix} \nu \\ \omega \end{bmatrix}. \quad (9)$$

A partir de este modelo en coordenadas  $(\rho, \alpha, \beta)$ , se diseña una ley de control que garantice la convergencia del robot a la pose objetivo. En lugar de linealizar el sistema alrededor de un punto de operación, se emplea un enfoque no lineal basado en la teoría de estabilidad de Lyapunov. Mediante una función de Lyapunov candidata expresada en términos de  $\rho$ ,  $\alpha$  y  $\beta$ , se obtienen condiciones sobre las velocidades de control  $\nu$  y  $\omega$  que aseguran que el error de pose disminuya con el tiempo [14], [16]. Este procedimiento conduce a una ley de control de la forma:

$$\begin{aligned} \nu &= k_\rho \rho & \omega &= k_\alpha \alpha + k_\beta \beta \\ k_\rho &> 0, \quad k_\beta < 0 & k_\alpha + \frac{5}{3}k_\beta - \frac{2}{\pi}k_\rho &> 0, \end{aligned} \quad (10)$$

donde  $\nu$  y  $\omega$  representan la velocidad lineal y angular de control, respectivamente, y  $k_\rho$ ,  $k_\alpha$  y  $k_\beta$  son ganancias de control. Las restricciones sobre estas ganancias se derivan directamente del análisis de estabilidad y garantizan que la pose del robot converja asintóticamente al objetivo.

**Figura 14.** Control no lineal de pose



Nota. Diagrama que muestra el marco de referencia  $G$  y la descomposición de la posición para definir el vector  $\rho$ , la orientación actual  $\theta$  y el ángulo asociado a la pose objetivo  $\beta$ . Elaborado por [17].

## 7.1. Metodología

Para diseñar la pinza, primero se analizaron diseños preexistentes con el fin de identificar características que pudieran implementarse en este proyecto. De estos diseños, las características más útiles para nuestra pinza fueron las siguientes:

**Figura 15.** Garra genérica



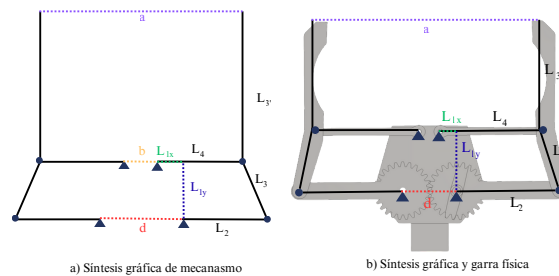
Nota. Esta pinza recopila la mayoría de características útiles que ayudaron a inspirar este diseño. Obtenida de [18]

1. El servo se acopla al mecanismo mediante la hélice del mismo, actuando como método de sujeción.
2. Mediante los engranajes se permite transmitir el movimiento de un brazo a otro.

3. Las pinzas emplean un agarre paralelo, evitando que las piezas se deslicen o se desorienten al ser sujetadas.
4. Cada brazo está compuesto por un mecanismo de cuatro barras como se puede observar en la Figura 16.

A partir de estos puntos, se inició el diseño de la pinza con las dimensiones deseadas. El proceso fue iterativo, utilizando la herramienta MotionGen para observar el movimiento de la pinza y Autodesk Inventor para obtener las medidas precisas del mecanismo. Cada vez que se modificaba algún parámetro, se simulaba primero y posteriormente se fabricaba mediante impresión 3D.

**Figura 16.** Síntesis gráfica mecanismo Grashof especial



Nota. Síntesis gráfica de dos mecanismos de Grashof especial, mostrando el nombre de cada eslabón y se muestra la síntesis gráfica sobre el mecanismo para mostrar cada parte de los eslabones y sus respectivas dimensiones. Elaboración propia.

Como se mencionó anteriormente, la pinza está compuesta por dos mecanismos de cuatro barras de tipo Grashof especial, uno en cada brazo. Tanto el brazo derecho como el izquierdo poseen las mismas dimensiones, lo que representa una ventaja de diseño, ya que al dimensionar un brazo, el otro se define automáticamente.

Las partes que componen todas las iteraciones de pinza son: los eslabones que componen la pinza son  $L_2$  (manivela),  $L_3$  (acoplador) y  $L_4$  (balancín). El eslabón  $L_2$  incluye el engranaje donde se acopla el servomotor en el brazo derecho, mientras que en el brazo izquierdo se sujeta mediante un tornillo.  $L_3$  es el eslabón que extiende la pinza para sujetar los objetos, y su extensión se denomina  $L_{3'}$ . Finalmente,  $L_4$  actúa como balancín, completando el mecanismo y permitiendo el movimiento coordinado de la pinza.

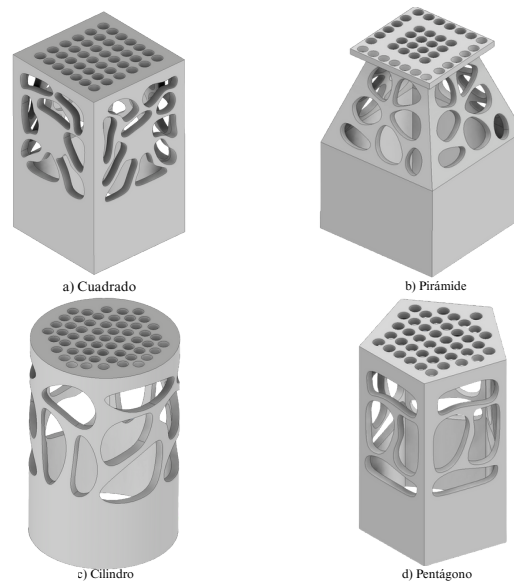
Se realizaron múltiples iteraciones de diseño con el fin de determinar qué configuración de la pinza permitía manipular mejor los diferentes objetos y generaba un agarre más firme. Como métrica de evaluación cualitativa, se observó el comportamiento de la pieza sujeta mientras el robot realizaba distintos movimientos: si, durante el desplazamiento, la figura se soltaba, era expulsada ante maniobras rápidas o interfería con la movilidad del robot, se consideraba que el agarre no era adecuado y se identificaban puntos de mejora en el diseño.

Una vez satisfechos con el diseño de la pinza se llevaron a cabo una serie de pruebas específicas para evaluar la efectividad de la pinza y, en etapas posteriores, cuantificar su desempeño de manera más sistemática.

### 7.1.1. Diseño de figuras

Se definieron 4 figuras para poder evaluar el comportamiento de la pinza con diferentes objetos: cuadrado, pentágono, pirámide y cilindro. Todos cuentan con un diseño llamado voronoi, implementado en las figuras para mejorar el agarre de la pinza, ya que los espacios permiten engancharse a los brazos. Además, este diseño disminuye el peso de las piezas y facilita su desplazamiento. En la base, las piezas son lisas para reducir el centro de masa y evitar que se volteen con facilidad. En la parte superior cuentan con orificios para colocar el marcador reflectante y permitir su identificación dentro del Robotat. El cuadrado incluye espacios internos para la colocar imanes, ya que será utilizado para formar una figura específica. Todas las piezas fueron fabricadas en filamento PLA mediante impresión 3D.

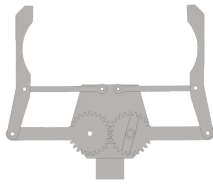
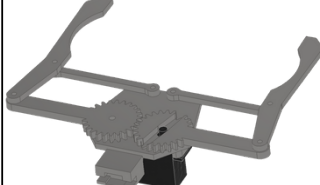
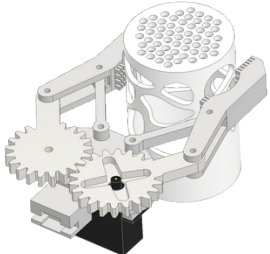
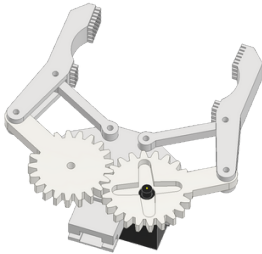
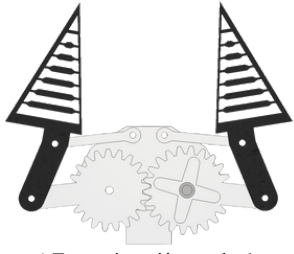
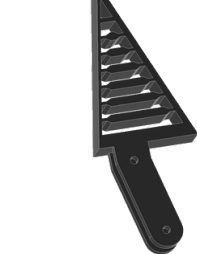
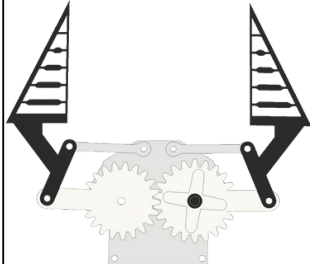
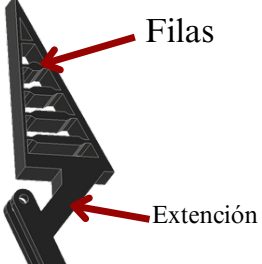
**Figura 17.** Figuras para medir efectividad de la pinza



Nota. Figuras que se utilizarán para medir la capacidad de la pinza.  
Elaboración propia.

## 7.2. Iteraciones realizadas

Figura 18. Diseño y dimensiones finales de las distintas iteraciones realizadas

Primera iteración		Dimensiones finales
		$L_1 = 28.504mm$ $L_2 = 48.000mm$ $L_3 = 28.504mm$ $L_{3'} = 52.000mm$ $L_4 = 48.000mm$ $a = 106.000mm$
a) Vista superior de primera iteración	b) Vista lateral de primera iteración	
Segunda iteración		Dimensiones finales
		$L_1 = 22.472mm$ $L_2 = 30.000mm$ $L_3 = 22.472mm$ $L_{3'} = 50.000mm$ $L_4 = 30.000mm$ $a = 74.000mm$
c) Segunda iteración pinza cerrada	d) Tercera iteración pinza abierta	
Tercera iteración: prueba 1		Dimensiones finales
		$L_1 = 22.847mm$ $L_2 = 30.000mm$ $L_1 = 28.847mm$ $L_{3'} = 50.000mm$ $L_4 = 30.000mm$ $a = 50.000mm$
e) Tercera iteración prueba 1	f) Brazo ( $L_3$ ) con efecto FinRay	
Tercera iteración: prueba 2		Dimensiones finales
		$L_1 = 22.847mm$ $L_2 = 30.000mm$ $L_1 = 28.847mm$ $L_{3'} = 50.000mm$ $L_4 = 30.000mm$ $a = 72.669mm$
g) Resultado tercera iteración	h) Brazo con efecto FinRay®	

Nota. Se muestra los resultados finales de cada iteración realizada y las dimensiones respectivas de cada eslabón y la apertura (a) de la pinza. Elaboración propia.

### 7.2.1. Primera iteración

Al iniciar el diseño en Autodesk Inventor se contaba con dos medidas conocidas:  $d$  que es la distancia entre los centros de los engranajes ilustrado en la Figura 16. La segunda es  $L_1$  esta medida se obtuvo calculandola con la siguiente fórmula:

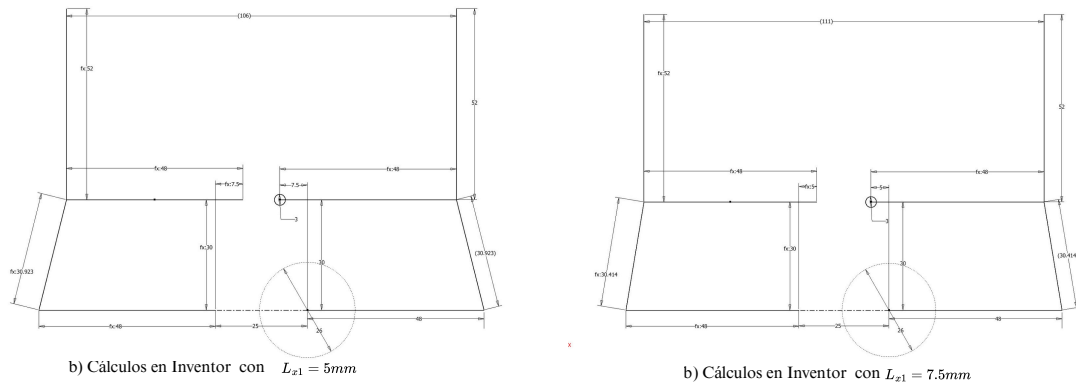
$$L_1 > \text{radio del engranaje} + \frac{\text{grosor del balancín}}{2}. \quad (11)$$

Esta condición garantiza que el balancín y el engranaje no se interfieran entre sí durante el movimiento. Al tratarse de un mecanismo de tipo Grashof especial, los eslabones  $L_2$  y  $L_4$  poseen la misma longitud. Esto permite modificar libremente las dimensiones de  $L_1$  (específicamente  $L_{x1}$  y  $L_{y1}$ ) y  $L_2$  para realizar las pruebas. En este caso,  $L_2$  se definió con un valor de 48 mm, mientras que solo se ajustaron los valores de  $L_1$ .

Al modificar la longitud  $L_{x1}$  se observa que la amplitud del movimiento varía ligeramente. Para  $L_{x1} = 5$  mm, la amplitud es de 111 mm, mientras que para  $L_{x1} = 7.5$  mm, la amplitud se reduce a 106 mm. Esta variación de 5 mm en la amplitud indica que el cambio en  $L_{x1}$  tiene un efecto poco significativo sobre el recorrido total.

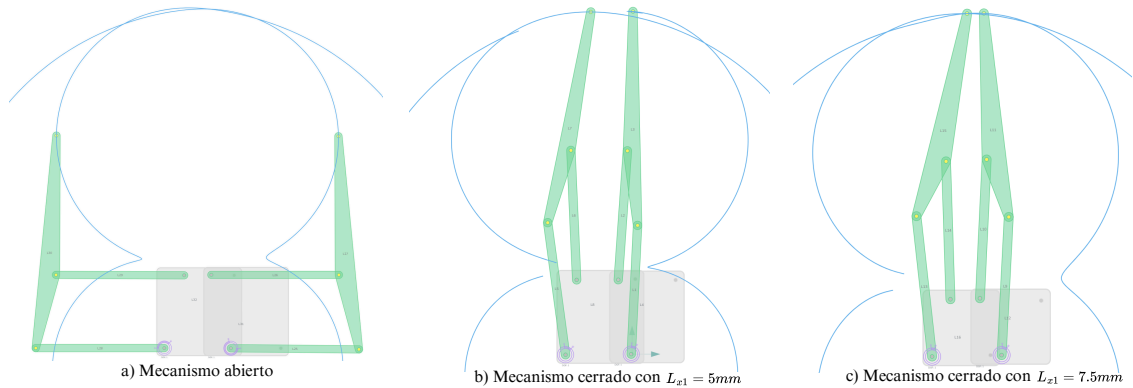
Sin embargo, si  $L_{x1}$  era muy pequeña, al cerrar el mecanismo se produce una singularidad. Como se observa en la Figura 20 b), el mecanismo no termina de cerrar y el brazo alcanza una posición singular. Por el contrario, en la Figura 20 c), el mecanismo cierra correctamente sin presentar problemas. Este comportamiento puede afectar el agarre de la pinza. Por lo tanto, se concluye que un mayor valor de  $L_{x1}$  es más favorable.

**Figura 19.** Síntesis gráfica mecanismo iteración 1



Nota. Iteraciones realizadas en inventor para conocer las medidas de  $L_3$  y la amplitud  $a$ . Elaboración propia.

**Figura 20.** Simulación de pinza en MotionGen

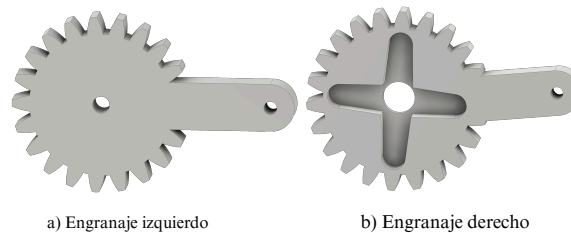


Nota. Simulaciones realizadas en MotionGen utilizando dos diferentes dimensiones de  $L_{x1}$ . Elaboración propia.

Se realizaron varias iteraciones modificando la altura  $L_{y1}$ , pero no se observó un cambio significativo en la amplitud. Por ello, el diseño final adoptó  $L_{y1} = 27.5$  mm y  $L_{x1} = 7.5$  mm, ya que estos valores mostraron un mejor comportamiento en la simulación del mecanismo obteniendo la máxima amplitud posible. Las dimensiones finales de cada eslabón se pueden observar en la Figura 18 a). Durante las pruebas físicas ensamblada al Pololu, se identificaron aspectos a mejorar. En primer lugar, aunque la apertura era correcta, se evidenció que la pinza era relativamente grande comparado al tamaño del robot, lo que podría dificultar el control dentro del Robotat debido al área ocupada por el robot. En segundo lugar, se observó que la hélice, al contar únicamente con dos alas, podría requerir un mayor esfuerzo del servomotor para mover el engranaje así que se consideró cambiar a otro tipo. Todas las piezas del mecanismo fueron fabricadas mediante impresión 3D en material PLA.

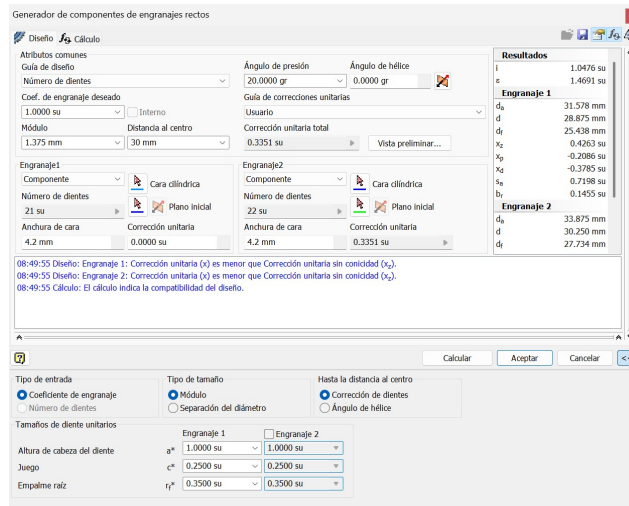
### 7.2.2. Segunda iteración

**Figura 21.** Engranajes utilizados en la segunda pinza



Nota. Para engranaje derecho se utilizó otro diseño de hélice y se redimensionó  $L_2$ . Elaboración propia.

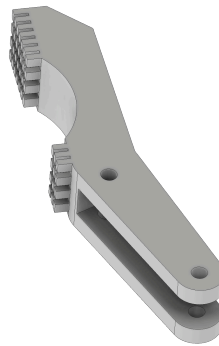
**Figura 22.** Configuración para engranajes



Nota. Se muestra el diámetro ( $d_a$ ) del engranaje 1 (engranaje derecho de la pinza), y engranaje 2 (engranaje izquierdo de la pinza). Elaboración propia.

Tomando en cuenta las mejoras obtenidas en la primera iteración, se realizaron los siguientes cambios en el diseño. En primer lugar, se empleó una hélice en forma de estrella para facilitar su acoplamiento. Esto implicó la necesidad de redimensionar el engranaje, cuyo diseño final se muestra en la Figura 21 b). Dado que el engranaje no estaría sometido a grandes fuerzas tangenciales, el criterio de diseño no se centró en la resistencia mecánica. El enfoque principal consistió en garantizar un diámetro suficiente para permitir el acoplamiento de la hélice. Asimismo, se estableció una relación de transmisión uno a uno, procurando que ambos engranajes tuvieran diámetros lo más similares posible. El engranaje 1 corresponde a  $L_2$  del brazo derecho y tiene un diámetro de 31.578 mm, mientras que el engranaje 2 corresponde a  $L_2$  del brazo izquierdo y presenta un diámetro de 33.875 mm, estas medidas se pueden observar en la Figura 22. Por otra parte, el eslabón  $L_1$  se definió con una longitud de 30 mm, con el objetivo de obtener un diseño más compacto.

**Figura 23.** Brazo de pinza (eslabón  $L_3$  y  $L_3'$ ) segunda iteración

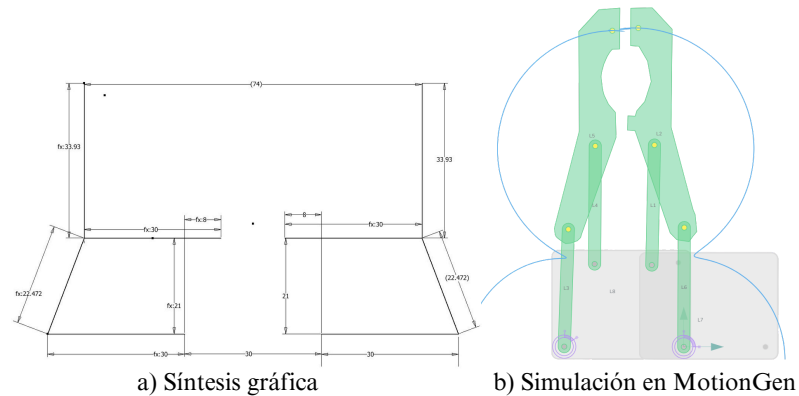


Nota. Diseño nuevo para brazo de pinza tomando en cuenta mejoras pasadas. Elaboración propia.

El eslabón  $L_{3'}$  también presentó cambios significativos en su geometría. En primer lugar, su dimensión se redujo respecto al diseño inicial, pasando de 28.50 mm a 22.47mm. Además, se aumentó su grosor a 10 mm con el propósito de realizar un vaciado en el centro, lo cual otorga una mayor estabilidad en el acoplamiento con los eslabones  $L_2$  y  $L_4$ .

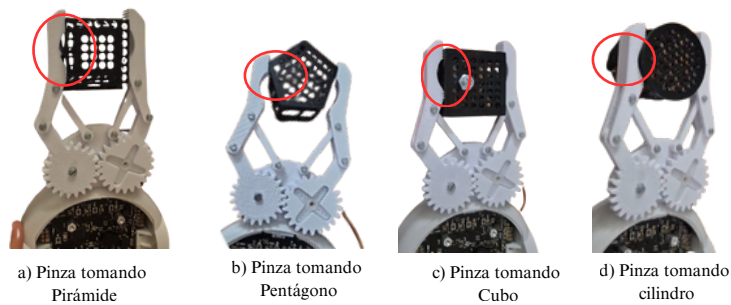
En cuanto a la extensión de  $L_{3'}$  su largo anterior era de 52 mm y ahora es de 33.91 mm. Al incrementar su grosor se mejoró la capacidad de sujeción de los objetos al poseer más superficie de sujeción. Adicionalmente, se incorporaron dientes en su superficie para evitar que esta fuera completamente lisa, ya que al estar fabricada en material PLA, al igual que los objetos a manipular se deslizaba más fácil.

**Figura 24.** Síntesis gráfica final para mecanismo iteración 2



Nota. Se muestra en a) las medidas de todos los eslabones para la pinza final y b) simulación de movimiento realizada en *MotionGen*. Elaboración propia

**Figura 25.** Desempeño físico capturando diferentes geometrías de la segunda iteración de diseño para la pinza



Nota. Pruebas de sujeción realizadas con las 4 figuras geométricas

Con todas estas modificaciones se obtuvo la pinza mostrada en la Figura 18 c). Al ensamblarla en el robot Pololu, las dimensiones resultaron más adecuadas, como se aprecia en la Figura 18. La simulación realizada en MotionGen permitió confirmar que el mecanismo no presentaría singularidades a lo largo de su recorrido y que su comportamiento sería el esperado, tal como se muestra en la Figura 24 b). En cuanto a las mejoras identificadas,

se observó que el eslabón  $L_{3'}$  cumplía con sujetar adecuadamente las figuras; sin embargo, la fuerza de sujeción no era suficiente, como se aprecia en la Figura 25, ya que en giros rápidos algunas piezas llegaban a soltarse. En el caso de la figura cuadrada no se presentaron inconvenientes, pero con otras geometrías este problema fue evidente.

### 7.2.3. Tercera iteración

El análisis evidenció que el brazo  $L_{3'}$ , fabricado en PLA, presentaba el mismo material que las piezas a manipular, lo cual disminuía la eficiencia del agarre debido a la limitada fricción superficial. Como medida correctiva, se incorporó el efecto *Fin-Ray*<sup>®</sup> en el diseño de  $L_{3'}$  y se sustituyó el material por TPU. Esta modificación permitió reducir el deslizamiento y mejorar la estabilidad durante la manipulación de los objetos. También se rediseñó la base de la pinza debido a deficiencias en el sistema de fijación anterior. Inicialmente, la base se acoplaba al chasis del robot mediante presión, lo que resultaba en una sujeción insuficiente. En el nuevo diseño, la base se atornilla directamente a la placa pcb del ESP32, garantizando una fijación rígida y mejorando la estabilidad del conjunto.

#### Primera pinza

El diseño final para ésta iteración se puede observar en la Figura 18 e). En esta configuración no se logró la flexibilidad esperada debido al incremento en la rigidez del conjunto. El espesor de las paredes y la alta densidad de filas internas generaban mayor soporte estructural, impidiendo la correcta deformación del elemento triangular. Si bien el material permitía cierta deformación, esta no era suficiente para afirmar que la pinza se adaptaba efectivamente a la geometría de los objetos manipulados. Además, la apertura máxima (a) se redujo a 50 mm, lo que dejaba una holgura mínima considerando que las piezas tienen un diámetro de 40 mm. Esta limitación comprometía la maniobrabilidad y el margen de ajuste, por lo que fue necesario plantear una alternativa que incrementara la amplitud de apertura y, simultáneamente, mejorara la flexibilidad del sistema.

#### Segunda pinza

A partir del análisis de la iteración previa, se procedió al rediseño del eslabón  $L_{3'}$ . Se eliminaron varias filas internas de refuerzo para disminuir la rigidez del componente, tal como se observa en la Figura 18 g). Adicionalmente, se redujo el espesor de la pared lateral izquierda con el fin de mejorar la deformabilidad de la pinza durante la manipulación de objetos con geometrías variables. Como complemento, se añadió una extensión estructural que incrementó la amplitud de apertura hasta 72.669 mm, como se muestra en la Figura 18 h).

### 7.3. Resultados

Para medir la efectividad de la pinza se llevaron a cabo pruebas orientadas a evaluar la calidad del agarre, verificando que las piezas permanecieran sujetas durante todo el recorrido y no se soltaran durante la manipulación. Cada una de las ocho categorías evaluadas tuvo una ponderación de 12.5 puntos, de modo que la suma total alcanzara 100 puntos. Para cada figura se realizaron 20 iteraciones, cuyos resultados se presentan a continuación.

Con el propósito de analizar el desempeño en cada prueba individual, los puntajes obtenidos se agruparon en tres rangos. Se consideró un comportamiento excelente cuando la calificación se encontraba entre 95 y 100 puntos, ya que garantizaba que la pieza no se soltaba en ningún momento del recorrido. Un puntaje entre 85 y 94 puntos se clasificó como bueno, indicando que el agarre era generalmente efectivo, aunque podían presentarse fallas ocasionales sin comprometer completamente la tarea. Finalmente, valores iguales o inferiores a 84 puntos se catalogaron como deficientes, debido a la probabilidad significativa de pérdida de la pieza o posicionamiento incorrecto.

Para establecer una conclusión general sobre el funcionamiento de cada figura, se adoptó como criterio de aceptación un desempeño promedio igual o superior al 85 %. En consecuencia, cuando el resultado global se ubicaba dentro de los rangos excelente o bueno, se determinó que la pinza cumplía adecuadamente su función. Por el contrario, si el promedio se encontraba por debajo de este umbral, se consideró que el desempeño presentaba limitaciones relevantes en el agarre.

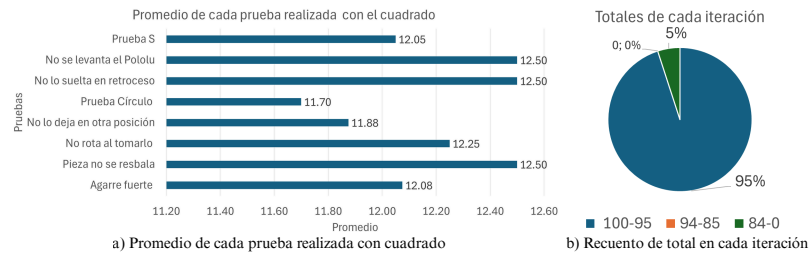
Para evaluar de manera estructurada el desempeño del agarre, se definieron ocho criterios específicos que permitieron analizar distintos escenarios dinámicos y estáticos. Estos criterios se describen a continuación:

1. **Agarre fuerte:** verificar que la pinza tome los objetos con suficiente fuerza desde el inicio.
2. **Toma sin resbalamiento:** al sujetar la pieza, esta no debe desplazarse de su posición inicial. En caso de un ligero movimiento, aún se considera un buen agarre.
3. **No rotación al tomar la pieza:** la pieza no debe girar sobre su eje al ser sujeta, de modo que conserve su posición inicial.
4. **No alterar la posición al soltarla:** al liberar la pieza, esta debe quedar en una orientación similar a la que tenía durante la sujeción.
5. **Prueba de círculo:** consiste en hacer girar al robot a alta velocidad; si la pieza no se desprende por la fuerza centrífuga, el agarre es considerado fuerte.
6. **No soltar en retroceso:** el robot retrocede a alta velocidad y la pieza no debe quedar en el trayecto.
7. **No levantamiento:** se verifica que el robot no pierda tracción ni se eleve al sujetar piezas de geometría particular.
8. **Prueba en "S":** el robot avanza con un movimiento serpenteante; si la pieza no se desprende durante estas maniobras, se confirma la solidez del agarre.

### 7.3.1. Resultados de las pruebas

#### Cuadrado:

**Figura 26.** Resultado de pruebas realizadas con el cuadrado



Nota. Se muestra en a) el promedio de 20 iteraciones de cada prueba realizada y b) diagrama de pie que recopila el resultado de cada iteración realizada. Elaboración propia.

Como se muestra en la Figura 26, las pruebas críticas en las que el cuadrado podía fallar correspondieron a los casos en que la pieza se dejaba en una posición diferente a la inicial o bien durante la prueba del círculo, donde ocasionalmente se soltaba. Sin embargo, el puntaje promedio no descendió de 11.7 puntos, lo que indica que estas situaciones se presentaron en muy pocas ocasiones. En la Figura 26 b) se observa que el 95 % de las veces la pinza operó con una efectividad dentro del rango de 100 a 95 puntos, lo que demuestra que la manipulación de la figura cuadrada no representa una dificultad significativa para la pinza y que los fallos son poco frecuentes. En consecuencia, la efectividad funcional de la pinza para el cuadrado se estima en un 95 %.

#### Pirámide:

**Figura 27.** Resultado de pruebas realizadas con la pirámide



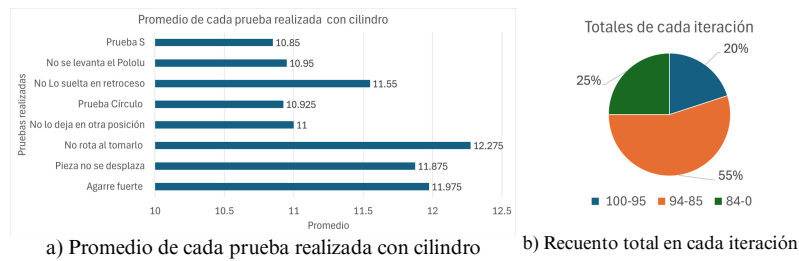
Nota. Se muestra en a) el promedio de 20 iteraciones de cada prueba realizada y b) diagrama de pie que recopila el resultado de cada iteración realizada. Elaboración propia.

Las pruebas críticas se presentaron principalmente en la prueba en “S”, donde, tras varios giros bruscos, la pieza tendía a soltarse. En algunas ocasiones, el robot Pololu se levantaba debido a que la pirámide posee lados inclinados y, si los brazos se sujetaban en una zona elevada, esto afectaba la estabilidad del robot. En la prueba del círculo, al girar por mucho

tiempo, la pieza se desplazaba durante el agarre y, al no mantener la misma orientación inicial, llegaba a soltarse. En la Figura 27 b) se observa que el 65 % de las pruebas alcanzó una efectividad en el rango de 100–95 puntos, mientras que un 20 % se ubicó entre 94–85, lo que indica que en la mayoría de los casos la pinza cumple su función, aunque en un 15 % de situaciones extremas puede no dar el resultado esperado. Considerando como efectivas únicamente las pruebas cuyo desempeño se encuentra en el rango de 100–95 puntos, la efectividad funcional de la pinza al manipular la pirámide se estima en un 65 %.

### Cilindro:

**Figura 28.** Resultado de pruebas realizadas con la cilindro



Nota. Se muestra en a) el promedio de 20 iteraciones de cada prueba realizada y b) diagrama de pie que recopila el resultado de cada iteración realizada. Elaboración propia.

Como se muestra en la Figura 28, las evaluaciones realizadas con el cilindro evidencian que las pruebas más exigentes fueron la trayectoria en “S”, la trayectoria circular, el reposicionamiento del objeto y el retroceso del robot.

La pinza con efecto *Fin-Ray*<sup>®</sup> no logra deformarse adecuadamente alrededor de la superficie curva del cilindro, lo que limita el contacto efectivo prácticamente a un punto entre una de sus caras planas y la geometría cilíndrica (Figura 30). Esta condición reduce la estabilidad del agarre y aumenta la susceptibilidad al deslizamiento. Durante el retroceso, el cilindro experimenta una fuerza de fricción opuesta al movimiento del robot, generando un arrastre en su base. Este fenómeno provoca un desplazamiento progresivo del objeto hasta su liberación, lo cual coincide con las fallas observadas en varias iteraciones experimentales.

Este comportamiento se refleja en los resultados de la Figura 28 b): únicamente el 25 % de las pruebas alcanza puntuaciones entre 100 y 95 puntos, mientras que el 55 % se ubica en el rango de 94 a 85. El 20 % restante presenta fallos significativos asociados a la pérdida de sujeción durante maniobras dinámicas, especialmente en movimientos de giro y retroceso. Dado que el cilindro constituye la pieza más compleja de manipular, al no disponer de aristas ni superficies planas amplias que favorezcan el contacto, se consideró aceptable un rango de efectividad más amplio, entre 85 y 100 puntos. Bajo este criterio, tanto el intervalo de 100–95 como el de 94–85 se clasifican como satisfactorios, pues representan un desempeño funcional adecuado pese a las limitaciones geométricas inherentes a la pieza.

## Pentágono:

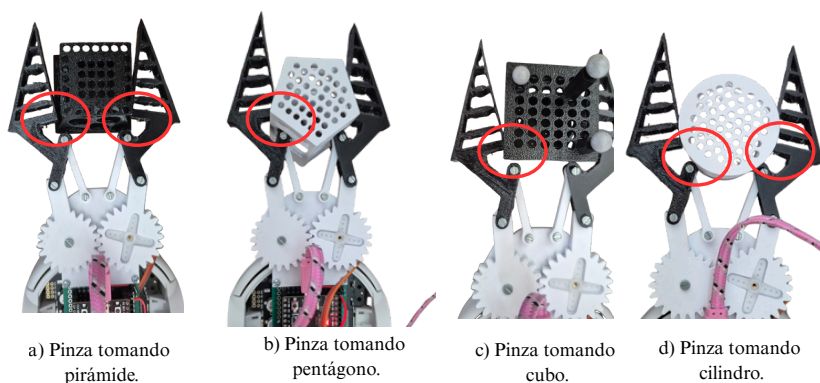
**Figura 29.** Resultado de pruebas realizadas con el pentágono



Nota. Se muestra en a) el promedio de 20 iteraciones de cada prueba realizada y b) diagrama de pie que recopila el resultado de cada iteración realizada. Elaboración propia.

Las pruebas críticas con el pentágono ocurrieron principalmente durante la liberación del objeto, ya que este no quedaba en la misma posición u orientación que la inicial. Esto se debe a que su geometría no ofrece dos superficies planas opuestas sobre las cuales la pinza pueda apoyarse de manera estable. Como se observa en la Figura 30, la pinza tiende a sujetar la pieza desde sus esquinas, generando dos fuerzas normales aplicadas en puntos no simétricos. Al abrirse la pinza, estas fuerzas producen un empuje desigual que desplaza la pieza o provoca que rote sobre su propio eje. En consecuencia, al ser depositado en un nuevo punto, el pentágono puede presentar una orientación diferente, dificultando el control preciso de su pose tras la liberación. A pesar de ello, los resultados mostrados en la Figura 29 b) indican que el 90 % de las pruebas alcanzó una puntuación entre 100–95. Este desempeño se considera aceptable, ya que demuestra que, aunque exista variación en la orientación, la pinza cumple adecuadamente con las tareas asignadas. Por lo tanto, la efectividad funcional de la pinza al manipular el pentágono se estima en un 90 %, considerando como efectivas las pruebas cuyo desempeño se encuentra en el rango de 100–95 puntos.

**Figura 30.** Demostración de agarre efectivo



Nota. La parte que está circunscrita en rojo es el área efectiva de la pinza, es decir esa es la parte que se utilizaba para tomar objetos. Elaboración propia.

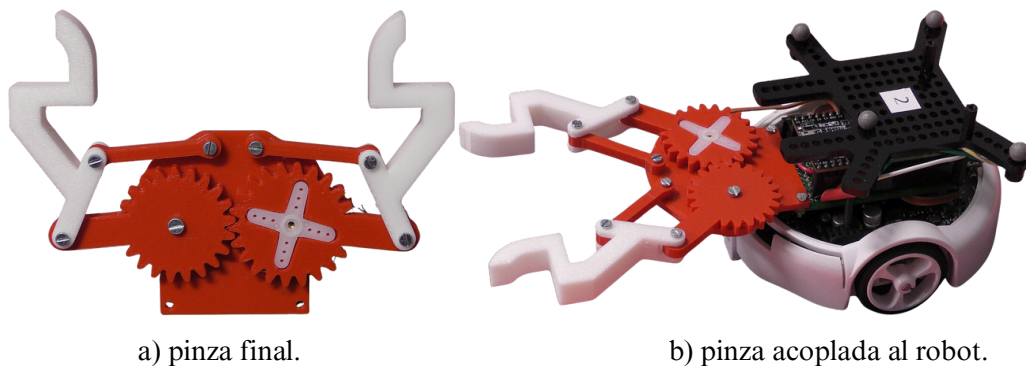
Luego de realizar todas las pruebas, se determinó que el agarre más efectivo de la pinza se obtenía en la zona inferior del triángulo que conforma la base del mecanismo, característica propia del efecto *Fin-Ray*<sup>®</sup> (Figura 30). Cuando el objeto se posicionaba ligeramente más adelantado, la estructura no se deformaba lo suficiente, impidiendo que la pinza envolviera adecuadamente la geometría de la pieza, tal como se esperaría teóricamente. Para lograr dicha adaptación habría sido necesario incrementar el ángulo de cierre del servomotor, aplicando mayor fuerza de compresión, además de plantear un rediseño que aumentara la flexibilidad del brazo.

No obstante, dado que dentro de una región específica la pinza alcanzó un agarre consistente y confiable, se estableció dicha zona como punto óptimo de contacto. Esta decisión se fundamenta en el desempeño observado experimentalmente, garantizando resultados satisfactorios durante la manipulación sin requerir modificaciones adicionales en la configuración actual del sistema.

### 7.3.2. Resultado final de la pinza

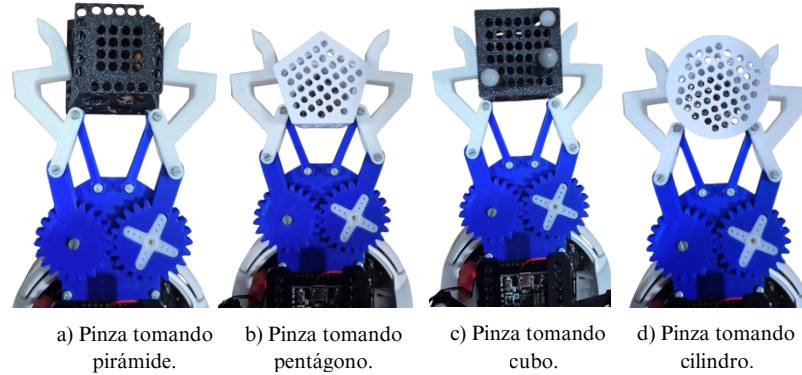
Para la iteración final, se realizó un rediseño en el cual únicamente se conservó la zona de trabajo efectiva identificada previamente, descartando así el uso completo del efecto *Fin-Ray*. Al tratarse de una simplificación de la iteración anterior, las pruebas iniciales confirmaron que el comportamiento con cada una de las figuras se mantenía prácticamente igual. Como se observa en la Figura 32, el agarre efectivo continúa ocurriendo en la misma región, incluso con este nuevo diseño. Por ello, se decidió trabajar directamente con esta versión final y evaluar su desempeño utilizando los resultados obtenidos en las pruebas previas.

**Figura 31.** Pinza final física



Nota. Se muestran dos vistas de la pinza, se puede observar que esta ya se adapta al tamaño del robot. Elaboración propia.

**Figura 32.** Demostración de agarre con diferentes figuras



Nota. Se muestra como la pinza si sujeta las piezas con toda el área efectiva. Elaboración propia.

## 7.4. Discusión de resultados

Las primeras iteraciones, basadas en una pinza convencional fabricada en PLA, evidenciaron varias limitaciones importantes. La rigidez del material y la baja fricción entre las superficies provocaban deslizamientos frecuentes, especialmente durante retrocesos o giros bruscos. Esto impedía que la pinza mantuviera un agarre estable y reducía considerablemente la confiabilidad del sistema. Posteriormente se intentó incorporar el efecto *Fin-Ray*<sup>®</sup> para aprovechar su potencial de adaptación a distintas geometrías. Aunque esta versión también fue fabricada en TPU, las dimensiones y distribución de las costillas no permitieron que la estructura se deformara de manera efectiva. La pinza resultó demasiado rígida para generar la flexión característica del *Fin-Ray*<sup>®</sup> y, además, la fuerza de agarre proporcionada por el servomotor no era suficiente para inducir la curvatura deseada. Incrementar los grados de cierre habría aumentado el consumo de corriente hasta niveles que podían comprometer la seguridad del ESP32, por lo que se descartó esta opción. Estas observaciones llevaron a descartar el uso completo del efecto *Fin-Ray*<sup>®</sup> y a concentrarse únicamente en la zona de sujeción que, durante las pruebas experimentales, demostró un rendimiento estable. Con base en esta zona efectiva se diseñó la última versión de la pinza, también fabricada en TPU, pero con una geometría simplificada y optimizada para maximizar el contacto y la fricción. Esto permitió obtener un agarre más fuerte y consistente sobre las diferentes piezas de prueba. Un elemento adicional que mejoró el desempeño general fue la incorporación del patrón voronoid en las figuras, ya que los orificios actuaban como puntos de apoyo que facilitaban el acoplamiento de la pinza y aumentaban la estabilidad durante la manipulación. Con base en las 20 iteraciones por figura de prueba realizadas, la pinza alcanzó una efectividad global del 87.5 %, obtenida como el promedio de las efectividades medidas para las cuatro figuras de prueba. Obteniendo resultados más altos en figuras con caras planas y mayor área de contacto. Los objetos curvos, como el cilindro, concentraron la mayoría de fallos debido a su falta de superficies amplias para apoyarse. Aun así, el diseño final demostró ser confiable frente a desplazamientos lineales, retrocesos y giros moderados.

---

## Cooperación de robots para formación de letra l

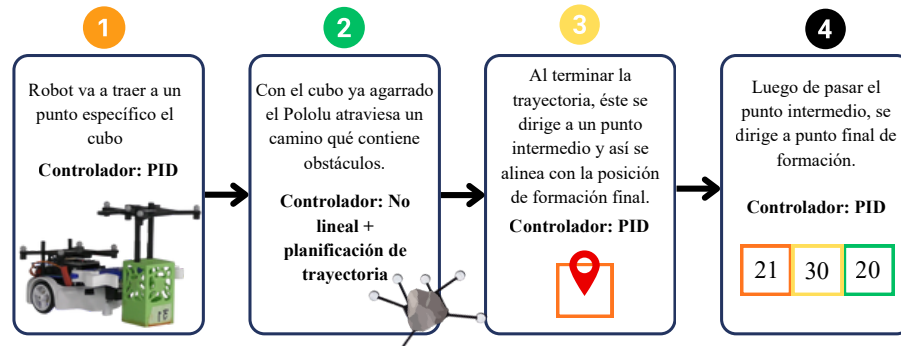
---

En este capítulo se describen los controladores implementados para alcanzar el objetivo de formar la letra *l* utilizando tres robots móviles. La tarea completa se divide en cuatro etapas principales. En la primera, cada robot se desplaza hacia un cubo y lo recoge en un punto específico. Una vez sujeta la pieza, la segunda etapa consiste en atravesar un camino con obstáculos previamente definidos en el entorno. En la tercera etapa, el robot se dirige a un punto intermedio con el fin de corregir su orientación y alinearse con el punto final de formación. Finalmente, en la cuarta etapa, el robot avanza hacia la posición asignada dentro de la formación, donde se unirán los tres cubos para construir la letra deseada.

Para su desarrollo se utilizó el sistema de captura de movimiento OptiTrack [1], el cual proporciona la posición y orientación tanto del robot como del objeto. Matlab fue empleado para el diseño e implementación de los controladores, así como para procesar la información proveniente del sistema de captura y calcular las velocidades de las ruedas y los comandos del servomotor de la pinza. La comunicación entre Matlab y el robot se realiza mediante el protocolo TCP hacia un ESP32 [2], que actúa como intermediario enviando los comandos a los motores DC y al servomotor, permitiendo ejecutar físicamente las órdenes de manipulación.

Como se muestra en la Figura 33, cada etapa emplea un controlador específico acorde con las necesidades de navegación, orientación o manipulación. El resto del capítulo detalla la implementación y resultados obtenidos con estos controladores.

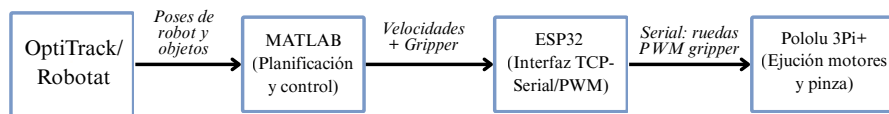
**Figura 33.** Etapas de control para la formación de la letra *l*



Nota. Diagrama de flujo de las etapas que conforman el control del robot para llevarlo al punto de formación. Elaboración propia.

## 8.1. Flujo de información y comunicación

**Figura 34.** Flujo de información entre OptiTrack, Matlab, ESP32 y Pololu 3Pi+



Nota. En las cajas se muestran los componentes que interviene en el flujo de información, en las flechas se indica los datos que se transmiten entre ellos. Elaboración propia.

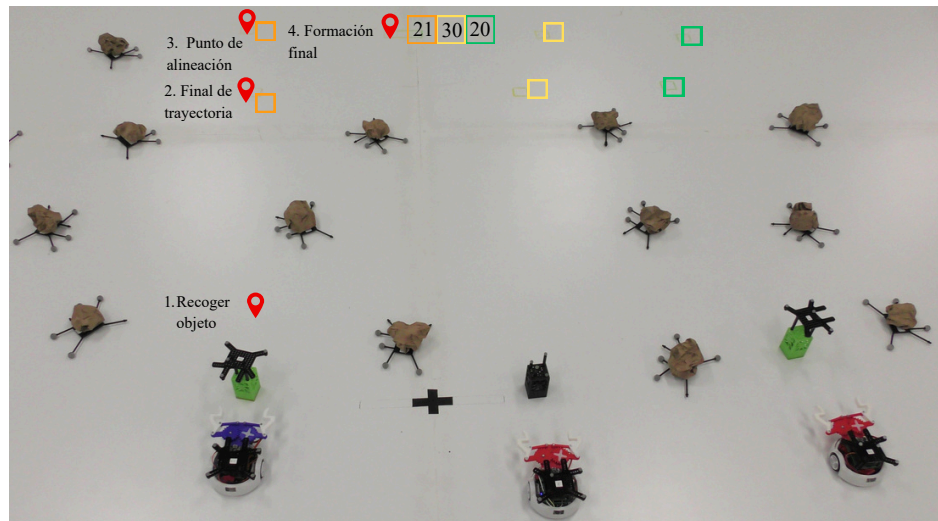
En la Figura 34 se muestra el flujo de información general del proyecto. El sistema OptiTrack captura la posición de los *markers*, que representan los obstáculos, así como la posición del robot y del objeto a manipular. Esta información es recibida por Matlab, encargado tanto de la planificación de trayectoria como del control, generando como salida las velocidades de las ruedas y los comandos del servomotor del gripper. Estos datos se envían mediante comunicación TCP al ESP32, el cual ejecuta directamente dos acciones: controlar los motores DC del robot y accionar la pinza para realizar la manipulación del objeto.

## 8.2. Primera etapa de control: PID para capturar al objeto

La primera tarea del robot consiste en recoger un cubo ubicado en una posición específica dentro del entorno de trabajo. Para ello, se implementó un controlador PID cuyo objetivo principal es regular la posición del robot hasta alcanzar el punto deseado. Es importante

señalar que, en esta etapa, únicamente se controló la posición del robot, sin considerar aún la orientación de la pieza.

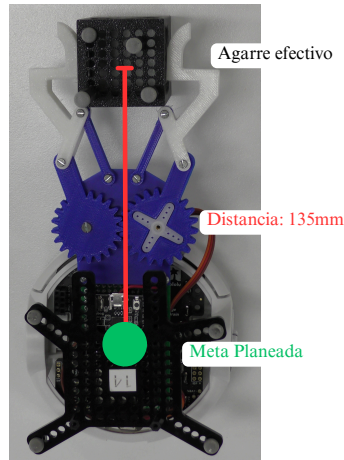
**Figura 35.** Posicionamiento de obstáculos metas y robots



Nota. Se puede observar la distribución de los obstáculos, los cuadrados anaranjados corresponden al cubo a las posiciones del 21, los cuadros amarillos corresponden al cubo 30 y los verdes al cubo 20. Elaboración propia

El proceso inicia tomando la posición inicial del robot y, dentro del ciclo de control, se estableció que el desplazamiento debe continuar hasta que el error de posición sea menor a 135 mm. Este valor corresponde al desfase existente entre el centro geométrico del robot Pololu 3Pi+ y el punto de agarre efectivo de la pinza, como se muestra en la Figura 36. De esta manera, cuando el robot alcanza dicho error, la pinza se encuentra correctamente alineada con el objeto, permitiendo que, al cerrarse, el cubo pueda ser tomado de forma precisa.

**Figura 36.** Medición de la distancia centro del Pololu a agarre efectivo



Nota. Esta medida indica la distancia que debe recorrer el robot para que el agarre de la pinza sea lo suficientemente fuerte. Elaboración propia.

Para garantizar un acercamiento suave y controlado hacia la meta, se implementó un perfil de aproximación exponencial que reduce progresivamente la velocidad de traslación a medida que el robot se aproxima al objetivo. Este esquema permite una llegada estable y sin oscilaciones significativas, favoreciendo una manipulación precisa en el momento del agarre.

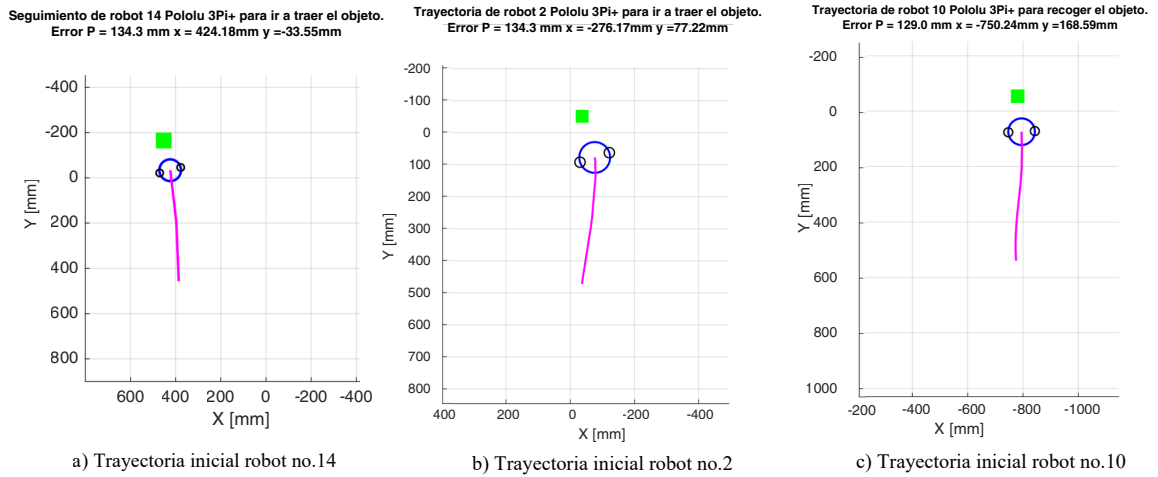
Una vez definida la estrategia de aproximación, se procedió a la sintonización experimental del controlador PID con el objetivo de obtener una respuesta rápida y estable, minimizando el sobrepaso y las oscilaciones. Tras diversas pruebas, se establecieron los siguientes parámetros: ganancia proporcional  $k_p = 8$ , ganancia derivativa  $k_d = 0.7$  y ganancia integral  $k_i = 0.0$ .

La ganancia proporcional ( $k_p$ ) determina la magnitud de la acción correctiva frente al error de posición, mientras que la ganancia derivativa ( $k_d$ ) introduce un efecto amortiguador que reduce la tendencia a la sobreoscilación y mejora la estabilidad transitoria del sistema. La acción integral ( $k_i$ ) se fijó en cero, dado que el sistema no presentaba error estacionario significativo que requiriera compensación acumulativa. De manera complementaria al controlador PID, el perfil de aproximación exponencial se parametrizó con una velocidad inicial de 1500 mm/s y un coeficiente de decaimiento  $\alpha = 0.000005$ . Estos valores permitieron disminuir progresivamente la velocidad conforme se reducía la distancia a la meta, sin comprometer el tiempo de respuesta del sistema.

### 8.2.1. Resultados

En la Figura 37 se observan las trayectorias obtenidas. Se aprecia que los movimientos son suaves, sin oscilaciones, y que los robots logran llegar al punto objetivo dentro del margen de error establecido (menor a 135 mm). En las gráficas también se observa un desfase entre el cubo (cuadro verde) y el robot, que corresponde a la distancia entre el centro del robot y el punto de agarre efectivo de la pinza. Este comportamiento confirma que el controlador PID,

**Figura 37.** Trayectoria seguida por el robot a cubo



Nota. El cuadro verde representa la meta dónde se encuentra el cubo, la línea fusia es la trayectoria seguida por el robot. Elaboración propia

en conjunto con la estrategia de acercamiento exponencial, permite alcanzar una posición adecuada para que la pinza pueda realizar el agarre del objeto de manera estable.

El Cuadro 1 permite evaluar la precisión del controlador PID implementado durante la etapa de acercamiento al objeto. Se observa que los errores finales de posición para los tres robots evaluados se mantienen consistentemente cercanos al valor de referencia de 135 mm, con variaciones mínimas entre iteraciones.

El robot 10 presenta la menor desviación estándar, con un valor de 0.98 mm, lo que evidencia un comportamiento altamente estable y repetible. Esto indica que el sistema de control logra converger hacia la meta de forma consistente, independientemente de las condiciones iniciales o pequeñas perturbaciones en la lectura de posición. El robot 2 muestra una desviación 1.43 mm, pero aún dentro de un margen muy reducido, lo que sugiere que las diferencias en error pueden atribuirse a la respuesta mecánica del robot. El robot 14 presenta la desviación estándar más alta 2.87 mm, aunque sigue siendo un valor pequeño que confirma la buena estabilidad del control.

**Cuadro 1.** Resultados de error de posición para distintos robots y metas

Robot	Iteración	Meta deseada (mm)		Posición final (mm)		Error (mm)
		$x$	$y$	$x$	$y$	
14	1	412.00	-66.40	376.32	62.33	133.60
	2	444.70	-78.52	470.66	45.30	126.50
	3	389.50	-76.65	387.95	58.38	134.89
	4	452.70	-164.80	424.18	-33.54	134.36
	5	460.80	-107.40	455.31	25.11	132.65
	6	452.70	-164.80	424.18	-33.54	134.36
<i>Desviación estándar</i>						<b>2.87</b>
2	1	-116.30	-252.30	-139.50	-123.08	131.29
	2	-33.00	116.40	-47.74	237.01	121.51
	3	-242.20	-54.40	-274.38	72.86	131.33
	4	-237.10	-51.30	-276.17	77.22	131.30
	5	-249.00	-51.90	-295.15	67.90	127.80
	6	-225.50	-113.10	-276.70	8.32	130.50
<i>Desviación estándar</i>						<b>1.43</b>
10	1	-724.30	-110.60	-720.19	19.01	128.70
	2	-738.00	40.10	-750.24	168.59	128.70
	3	-780.80	-54.10	-794.72	74.00	130.50
	4	-712.70	126.70	-741.14	-0.73	128.80
	5	-732.90	-46.90	-744.24	81.70	127.90
	6	-739.70	-80.10	-747.74	47.96	127.90
<i>Desviación estándar</i>						<b>0.98</b>

Nota. La desviación estándar se calcula sobre los errores de posición de cada robot, mostrando la variabilidad en la precisión de las corridas. Elaboración propia.

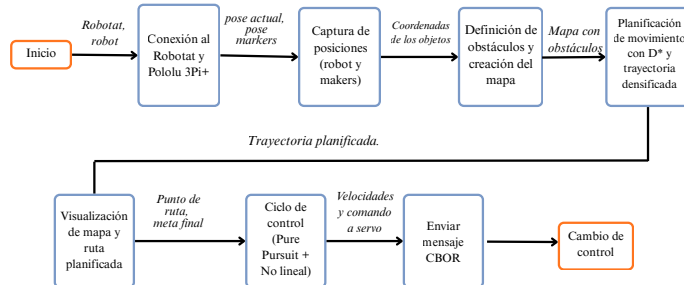
Los resultados demuestran que el controlador PID con acercamiento exponencial ofrece una excelente precisión y repetibilidad. Las pequeñas diferencias entre robots reflejan principalmente características individuales de sus sistemas mecánicos o de la comunicación con el entorno Robotat que pueden ya que el control depende del comportamiento que está teniendo el robot durante su ejecución. De este modo, se valida que el sistema logra posicionar el robot con el margen de error requerido para permitir el agarre fuerte y estable para las siguientes parte de control.

### 8.3. Segunda etapa: control no Lineal + planificación de trayectoria para atravesar mapa con obstáculos

Para esta segunda etapa, una vez que el robot recoge el objeto, se prepara para atravesar un camino con obstáculos. Para ello, se utiliza un controlador no lineal combinado con una planificación de trayectoria mediante el algoritmo D\* y el método de persecución pura (*Pure Pursuit*). Dado que el entorno contiene obstáculos, es necesario conocer su ubicación para

calcular una trayectoria que permita al robot desplazarse sin colisionar con los obstáculos. Para obtener los puntos finales de cada trayectoria, primero se coloca manualmente el cubo que usará cada robot en la posición deseada y se registra su posición en una variable. De esta forma, al calcular la trayectoria, se dispone de la meta fija, incluso cuando el cubo ya se encuentra sujeto por el robot y no es posible medir directamente su posición final durante la ejecución.

**Figura 38.** Diagrama de flujo del control no lineal + planificación de trayectoria

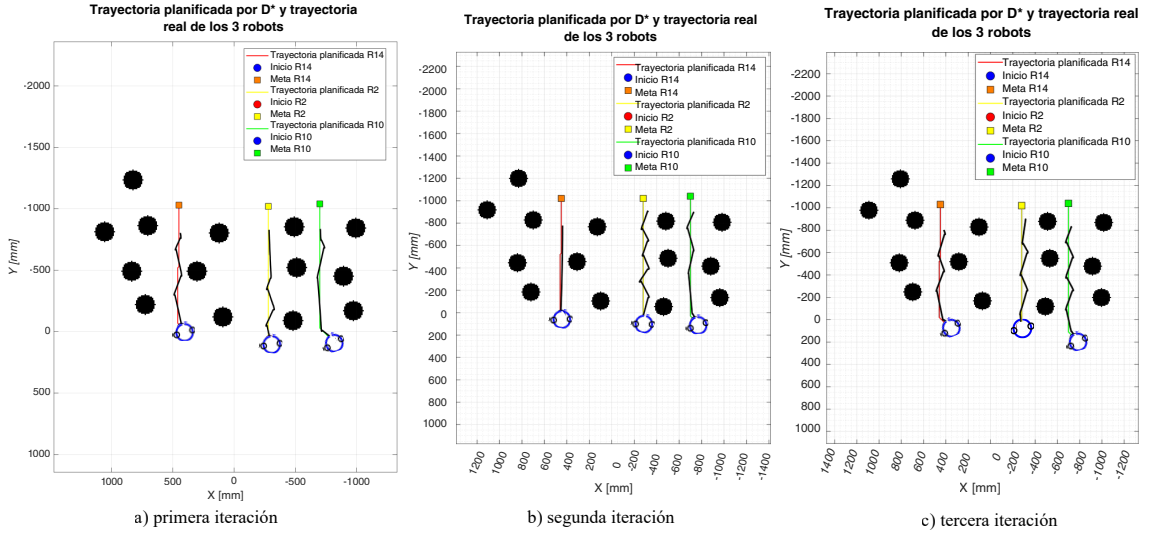


Nota. En los cuadros se muestran las principales etapas del código y en las flechas las variables que se generan en cada paso. Elaboración propia.

En la Figura 38 se presentan los pasos generales que sigue la programación del controlador. En la planificación de movimiento requiere definir dos puntos de referencia: inicio y fin. En este caso, el punto inicial correspondía a la posición del robot, y la meta es dónde terminan los obstáculos, estos puntos se puede observar en la Figura 35. Para cada robot se estableció éste punto. El cuadro amarillo representa el punto final de la trayectoria para el robot 2 y el cuadro verde representa el punto final para el robot 10. En este caso la posición como la orientación del robot son consideradas dentro de la ley de control. El error de posición se calcula con respecto a la meta, mientras que el error angular se define como la diferencia entre la orientación deseada y la orientación real del robot para mantenga sobre la trayectoria planificada, garantizando así un movimiento continuo y estable hacia la meta.

### 8.3.1. Resultados

Figura 39. Trayectoria seguida por el robot a meta final de trayectoria



Nota. Se muestran los tres robots y sus respectivas trayectorias. Elaboración propia.

Los parámetros para el controlador no lineal se seleccionaron los valores:  $k_p = 4$ ,  $k_\alpha = 40$  y  $k_\beta = -4$ , los cuales proporcionaron un comportamiento óptimo, logrando una aproximación suave y lenta al objetivo, con mínima oscilación del robot. Por otra parte, en el controlador persecución pura (*Pure Pursuit*) se configuraron los parámetros  $lookahead\_dist = 50$  mm para la distancia de *look-ahead* y  $idx\_goal = 1$  como índice de objetivo.

En la Figura 39 se muestran las trayectorias seguidas por los tres robots. La línea negra representa la trayectoria real seguida por el robot y las trayectorias muestran la ruta planificada, en las cuales se observa que el control mantiene el rumbo hacia la meta sin presentar oscilaciones altas. Cada robot inicia en una posición diferente y alcanza exitosamente el punto final indicado en su trayectoria. En las gráficas también se muestra que la línea negra se termina antes de llegar al punto final y esto es porque se implementó una condición que se basa en el cumplimiento simultáneo de un error de orientación menor al umbral  $e_{Otol2}$  y en que el robot haya recorrido al menos el 90% de los puntos planificados de la trayectoria esto es para cambiar automáticamente al siguiente error.

El desempeño del controlador se cuantificó mediante el error de orientación final alcanzado al llegar al punto final de la trayectoria. Los resultados se muestran en el Cuadro 2. Puede observarse que la mayoría de las iteraciones presentan valores pequeños (menores a 0.2 rad), lo cual indica una buena alineación durante toda su trayectoria.

**Cuadro 2.** Errores de orientación final ( $e_{ONL}$ ) para cada robot e iteración en radianes y grados

Robot	Iteración	$e_{ONL}$ (rad)	$e_{ONL}$ (°)
14	1	0.0581	3.33
	2	0.0843	4.83
	3	0.1428	8.18
	4	0.0116	0.66
	5	-0.1359	-7.79
	6	0.0724	4.15
2	1	0.0856	4.90
	2	0.1499	8.59
	3	0.1862	10.67
	4	0.0327	1.87
	5	0.1681	9.63
	6	0.1178	6.75
10	1	0.1421	8.14
	2	0.1686	9.66
	3	0.0620	3.55
	4	0.0863	4.94
	5	0.0901	5.16
	6	0.1486	8.51

Nota. El error de orientación se presenta en radianes y grados, calculado como la diferencia angular entre la dirección del robot y la orientación deseada al final de la trayectoria. Elaboración propia.

Se demuestra que el controlador no lineal logra un seguimiento estable y preciso de la trayectoria planificada, manteniendo bajos errores de orientación en la mayoría de las pruebas, siendo el más alto de  $10.67^\circ$  realizado por el robot 2 y esto es congruente con lo que se puede observar en la Figura 39 que éste robot fue el que más presentó oscilaciones entonces al controlador se le dificultó orientarse más a la posición final. Esto valida que la implementación es adecuada para tareas de transporte y manipulación dentro de un lugar con obstáculos, al permitir que el robot se desplace de forma controlada y mantenga la orientación necesaria para continuar hacia la siguiente etapa del proceso.

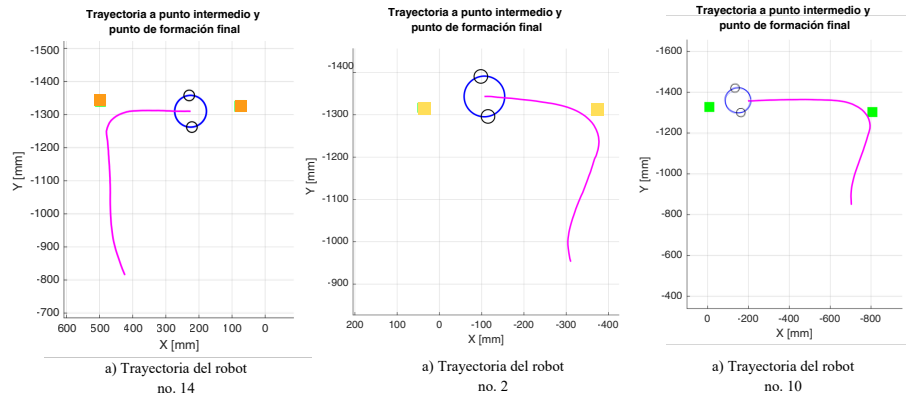
#### 8.4. Tercera etapa y final: controlador PID

Una vez terminada la planificación de trayectoria, el robot pasa a la tercera y última etapa de control. En esta fase se estableció un punto intermedio de alineación para que el robot ajuste su orientación de forma adecuada. Tras alcanzar este punto, el robot se dirige al punto de formación final. Ambos puntos pueden observarse en la Figura 35. Para definir estos puntos, primero se coloca cada cubo en la posición deseada de la formación y se registra su posición en una variable. Posteriormente, estas posiciones se utilizan como referencias durante la ejecución de los controladores. En esta etapa se emplearon los mismos parámetros de ajuste del PID y del acercamiento exponencial que en la primera etapa. La

formación final se ilustra en la Figura 35

### 8.4.1. Resultados

**Figura 40.** Trayectoria a punto intermedio y formación



Nota. Se muestra la trayectoria seguida por los tres robots, y se muestra el punto intermedio las metas finales el el punto de formación donde para el robot. Elaboración propia.

En la Figura 40 se observa que el robot atraviesa el punto intermedio y realiza un giro aproximado de  $90^\circ$  para dirigirse hacia la formación final. En todas las pruebas realizadas no se presentaron inconvenientes para alcanzar los puntos intermedios; sin embargo, es importante señalar que la orientación final con la que el robot deja el cubo depende directamente de la ubicación de dicho punto intermedio, ya que este determina el sentido del giro previo a la entrega. Dado que el controlador PID empleado no considera explícitamente la orientación dentro de su ley de control, la elección del punto intermedio cumple un papel fundamental para orientar al robot en la dirección correcta hacia la formación.

En la Cuadro 3 se presentan las posiciones iniciales y finales de los cubos durante seis iteraciones experimentales, junto con el error de posición calculado como la distancia euclidiana entre ambas posiciones. Este error permite cuantificar la precisión con la que los robots colocan los cubos respecto a la formación teórica. Si el error es mayor a 30 mm en cualquiera de los cubos, se considera una ejecución no satisfactoria, ya que esto indica que la formación se aleja de la configuración ideal: en lugar de que la letra *l* quede completamente alineada, puede aparecer una traslación en los ejes *x*, *y* o en ambos, produciendo una figura inclinada. Este comportamiento no es deseable, pues se busca la mayor precisión posible en la formación final.

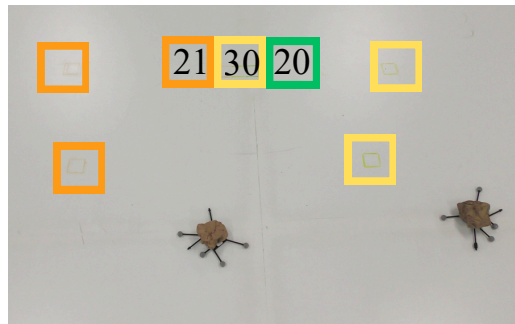
**Cuadro 3.** Errores de posición en la formación final

Iteración	Cubo	Formación inicial (mm)		Formación final (mm)		Error (mm)
		$x_i$	$y_i$	$x_f$	$y_f$	
1	20	916.44	-1326.57	908.60	-1314.72	14.21
2	20	74.79	-1328.17	80.85	-1314.03	15.38
3	20	77.33	-1325.84	78.99	-1329.69	4.19
4	20	74.63	-1328.30	77.92	-1327.36	3.42
5	20	70.11	-1319.15	73.55	-1318.71	3.47
6	20	85.00	-1314.90	99.00	-1311.90	14.32
<i>Error promedio</i>	20					9.33
1	30	580.39	-1321.34	587.22	-1325.52	8.01
2	30	425.58	-1304.75	427.19	-1302.72	2.59
3	30	36.07	-1332.41	38.22	-1326.79	6.02
4	30	36.27	-1323.34	29.57	-1325.14	6.94
5	30	512.86	-1304.69	512.86	-1304.96	0.27
6	30	-0.27	-1267.80	-273.20	-1265.80	2.283
<i>Error promedio</i>	30					4.35
1	21	97.68	-1329.99	98.43	-1331.48	1.67
2	21	-14.56	-1331.23	-6.88	-1342.74	13.84
3	21	11.12	-1330.60	-9.16	-1341.13	22.85
4	21	14.31	-1329.60	15.52	-1316.96	12.70
5	21	-13.84	-1304.69	-9.79	-1328.95	24.60
6	21	850.00	-1310.20	867.00	-1312.40	17.23
<i>Error promedio</i>	21					15.48

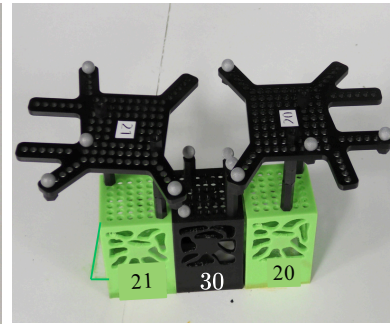
Nota. El error se calculó mediante  $E = \sqrt{(x_f - x_i)^2 + (y_f - y_i)^2}$ . Elaboración propia.

Los resultados muestran que la precisión en la colocación de los cubos fue, en general, adecuada. En todas las iteraciones el error de posición se mantuvo por debajo del umbral de 30 mm, por lo que, de acuerdo con el criterio definido, todas las ejecuciones pueden considerarse satisfactorias. Se observa, además, que los cubos 20 y 30 presentan errores relativamente pequeños, típicamente por debajo de 15 mm, lo cual indica que la referencia hacia sus puntos de formación es reproducible y poco sensible a variaciones en la trayectoria previa. De forma consistente con esto, el error promedio reportado en la Cuadro 3 es de aproximadamente 9.33 mm para el cubo 20 y de 4.35 mm para el cubo 30. En contraste, el cubo 21 tiende a mostrar los errores más elevados, con valores que alcanzan aproximadamente 25 mm y un error promedio cercano a 15.48 mm. Esto sugiere que su posición dentro de la formación es más crítica, ya sea por su ubicación en el extremo de la letra  $l$  o por la mayor dependencia de la orientación final del robot al momento de la entrega. A pesar de estas diferencias, los desplazamientos observados no alteran de forma significativa la forma global de la letra  $l$ , sino que se manifiestan principalmente como pequeñas traslaciones locales de cada cubo. En consecuencia, puede concluirse que la estrategia de control propuesta permite una formación cooperativa suficientemente precisa para la tarea planteada, dejando como posible trabajo futuro la reducción de estos errores mediante un control explícito de la orientación en la etapa final o mediante estrategias de corrección visual adicionales.

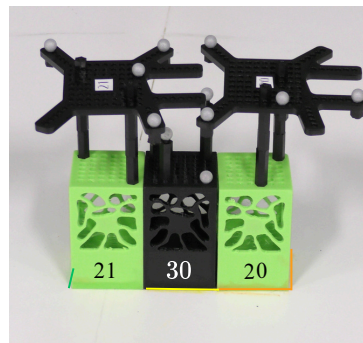
**Figura 41.** Formaciones iniciales y finales de cubos



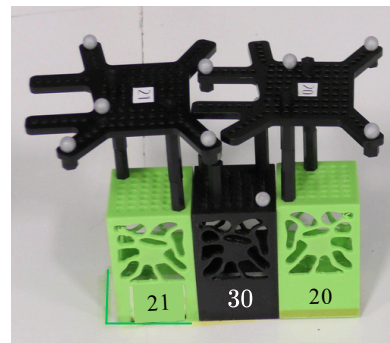
a) Formación inicial posición teórica



a) primera iteración de formación final



b) segunda iteración de formación final



c) tercera iteración de formación final

Nota. Se muestra la formación teórica que se debe alcanzar e iteraciones realizadas de cómo se formó la figura en “I”. Elaboración propia.

Finalmente, la Figura 41 presenta la disposición final de los cubos en la formación en “I”, confirmando que los robots lograron cumplir la tarea cooperativa de transporte y posicionamiento. Este resultado valida el correcto funcionamiento de la estrategia jerárquica de control implementada: el controlador PID inicial asegura la captura confiable del objeto, el controlador no lineal con planificación de trayectoria permite la navegación segura en presencia de obstáculos y, por último, el controlador PID final garantiza una entrega suficientemente precisa del objeto en la formación deseada.

- La definición de tres rangos de desempeño (100–95, 94–85 y 84–0 puntos) permitió vincular de forma directa el comportamiento observado con el riesgo operativo: en el rango 100–95 la probabilidad de falla es prácticamente nula; entre 94–85 aparece una probabilidad moderada de soldadura o cambio de orientación; y por debajo de 85 la manipulación puede presentar inconvenientes que comprometen el resultado. Bajo esta métrica, el cuadrado alcanzó una efectividad funcional del 95 %, con un 95 % de las iteraciones ubicadas en el rango 100–95 puntos, lo que confirma que para piezas con caras planas amplias el diseño final de la pinza garantiza un agarre muy estable y con fallos poco frecuentes. En contraste, las figuras pirámide y, sobre todo, cilindro se aproximan con mayor frecuencia a la zona de riesgo, evidenciando la sensibilidad del sistema ante geometrías menos favorables para el contacto.
- En la primera etapa de control, el uso de un controlador PID con acercamiento exponencial permitió posicionar al robot a una distancia final cercana a los 135 mm respecto al cubo, valor que corresponde al desfase entre el centro geométrico del Pololu 3Pi+ y el punto de agarre efectivo de la pinza. Los errores finales se mantuvieron agrupados alrededor de este valor, con desviaciones estándar de 0.98 mm, 1.43 mm y 2.87 mm para los robots 10, 2 y 14 respectivamente (Cuadro 1), y con trayectorias suaves y sin oscilaciones apreciables. Esta combinación de precisión en el error final y estabilidad en el movimiento garantiza que la pinza se ubique sistemáticamente en la zona de agarre efectivo, por lo que la probabilidad de fallar el agarre por una mala posición del robot en esta etapa es muy baja.
- El desempeño angular del controlador no lineal fue cuantificado mediante el error de orientación final  $e_{ONL}$ . En el Cuadro 2 se observa que los errores se mantuvieron mayoritariamente por debajo de 0.2 rad (aprox. 11°), con la peor condición registrada en 10.67° para el robot 2. Estos valores muestran que incluso en el peor caso el robot llega razonablemente alineado con la meta, lo que facilita el enlace con la siguiente etapa de control y reduce las correcciones posteriores de orientación necesarias para continuar la tarea.

- En conjunto, los resultados de las dos primeras etapas muestran que la arquitectura de control en cascada (PID para aproximación directa al cubo, seguido de control no lineal con planificación para atravesar el mapa con obstáculos) proporciona una navegación robusta: los errores de posición se mantienen en el orden de pocos milímetros respecto al valor objetivo de 135 mm, y los errores de orientación al final de las trayectorias son menores a  $11^\circ$ . Esto permite que cada robot llegue a sus puntos intermedios y metas de transporte con suficiente precisión para que las etapas posteriores de formación y orientación final puedan operar sin requerir grandes correcciones.
- El controlador PID permitió realizar de forma satisfactoria la formación de la letra "I": en todas las iteraciones el error de posición se mantuvo por debajo del umbral de 30 mm, por lo que el 100 % de las ejecuciones puede considerarse exitosa. Para la formación final, los errores promedio de posición fueron de 9.33 mm para el cubo 20, 4.35 mm para el cubo 30 y 15.48 mm para el cubo 21 (Cuadro 3), lo que indica que el controlador utilizado es capaz de posicionar los cubos con errores típicos inferiores a 2 cm en promedio.

- Se recomienda continuar el estudio y la optimización del uso del efecto *Fin-Ray*<sup>®</sup> en el diseño de la pinza. Profundizar en esta línea permitiría explotar mejor las ventajas de este principio de deformación controlada, de modo que la pinza pueda adaptarse a figuras irregulares y a un mayor rango de geometrías sin requerir cambios mecánicos significativos. Esto ampliaría de forma importante la funcionalidad del sistema de agarre, haciendo posible la manipulación de objetos más diversos y acercando la aplicación del robot removedor de escombros que manipula formas y tamaños de los objetos no son uniformes ni perfectamente conocidos.
- Para mejorar la precisión de la formación de la letra "1", es necesario considerar explícitamente la orientación del cubo en el esquema de control. Se recomienda, por tanto, emplear el control no lineal y la planificación de trayectoria tomando como referencia la pose del cubo en lugar de la del robot, de manera que pueda regularse también la orientación con la que se coloca cada pieza. Al sustituir el controlador PID por el controlador no lineal en la etapa de formación, se abre la posibilidad de implementar configuraciones de formación más complejas.
- Se sugiere migrar la arquitectura de control actual, implementada en MATLAB, a un entorno basado en ROS (Robot Operating System) utilizando Python. Esta migración permitiría mejorar la comunicación entre el computador y los robots mediante tópicos y servicios estandarizados, incrementar la eficiencia del sistema y facilitar la integración de nuevos módulos de planificación, control o percepción desarrollados por otros grupos. Además, ejecutando los controladores directamente en plataformas embebidas compatibles con ROS se reduciría la dependencia de licencias de software y se favorecería la escalabilidad hacia escenarios con un mayor número de robots cooperando de manera simultánea.
- Otra línea de desarrollo consiste en incorporar un sistema de visión por computadora que complemente o, en ciertos escenarios, reemplace parcialmente la dependencia del sistema de captura de movimiento. Mediante técnicas de visión sería posible detectar y clasificar diferentes tipos de objetos, estimar su posición y orientación en tiempo real

y ajustar automáticamente el punto de agarre de la pinza. Asimismo, la retroalimentación visual permitiría verificar que los cubos fueron colocados correctamente en la formación deseada y corregir pequeños errores de alineación, lo que abriría la puerta a trabajar en entornos menos estructurados donde la ubicación de los objetos no esté completamente definida de antemano.

- Finalmente, se propone implementar un módulo de odometría que permita al robot estimar su propia pose a partir de los encoders de las ruedas y, en etapas posteriores, fusionar esta información con otros sensores, como una IMU o un LIDAR. Sobre esta base se podrían integrar algoritmos de mapeo y localización (SLAM), detección y esquiva de obstáculos y, en general, estrategias de navegación autónoma dentro de un mapa. De esta forma, el sistema podría evolucionar desde un entorno de laboratorio altamente controlado hacia aplicaciones más cercanas a escenarios reales de inspección, búsqueda y rescate o logística interior.

- 
- [1] C. P. Montoya, «Robotat: un ecosistema robótico de captura de movimiento y comunicación inalámbrica,» Universidad del Valle de Guatemala, inf. téc., 2022.
  - [2] J. Luis y A. Pineda, «Desarrollo de una placa de expansión para los agentes Pololu 3Pi+ que expanda sus capacidades dentro del ecosistema Robotat,» Universidad del Valle de Guatemala, inf. téc., 2024.
  - [3] L.-M. F. M. S. W. Werth, «Challenges and opportunities of using differential-drive robots with project-based learning pedagogies,» *Elektrotechnik and Informationstechnik*, 2020.
  - [4] P. Vlantis, C. P. Bechlioulis y K. J. Kyriakopoulos, «Mutli-Robot Cooperative Object Transportation with Guaranteed Safety and Convergence in Planar Obstacle Cluttered Workspaces via Configuration Space Decomposition,» *Robotics*, vol. 11, 6 dic. de 2022, ISSN: 22186581. DOI: 10.3390/robotics11060148.
  - [5] R. L. Norton, «Diseño de maquinaria: Síntesis y análisis de máquinas y mecanismos,» McGraw-Hill, inf. téc., 2013, págs. 24-48.
  - [6] M. Chitroda y D. B. K. Patle, «A Review on Technologies in Robotic Gripper,» *International Journal of Advanced Engineering and Nano Technology*, vol. 10, págs. 1-5, 5 mayo de 2023, ISSN: 23476389. DOI: 10.35940/ijaent.C7232.0511523. dirección: <https://www.ijaent.org/portfolio-item/C72320911322/>.
  - [7] K. Tai, A. R. El-Sayed, M. Shahriari, M. Biglarbegian y S. Mahmud, *State of the art robotic grippers and applications*, jun. de 2016. DOI: 10.3390/robotics5020011.
  - [8] W. Crooks, G. Vukasin, M. O'Sullivan, W. Messner y C. Rogers, «Fin Ray® effect inspired soft robotic gripper: From the roboSoft grand challenge toward optimization,» *Frontiers Robotics AI*, vol. 3, NOV nov. de 2016, ISSN: 22969144. DOI: 10.3389/frobt.2016.00070.
  - [9] M. I. Hasib, M. H. A. Nahiyani, F. I. Anik, M. J. Hossain, M. N. Salehin y F. year, «Development of an Adaptive Gripper with Fin-Ray effect,» inf. téc.

- [10] M. S. Sofla, H. Golshanian, E. I. Sklar y M. Calisti, «Modeling and modification of fin-ray effect grippers to improve their load capacity and grasp stability,» *Sensors and Actuators A: Physical*, vol. 392, oct. de 2025, ISSN: 09244247. DOI: 10.1016/j.sna.2025.116711.
- [11] C. S. Plata y J. G. Arancibia, «Diseño e implementación de prototipo de una pinza robótica adaptativa de tres dedos basada en estructura FinRay®,» *Revista Ingenierías Universidad de Medellín*, vol. 18, págs. 103-121, 34 jun. de 2019, ISSN: 16923324. DOI: 10.22395/rium.v18n34a7.
- [12] G. Dudek y M. Jenkin, *Computational principles of mobile robotics*. Cambridge University Press, 2010, pág. 391, ISBN: 9780521692120.
- [13] H. Garnier y M. Jha, «Lab n • 5 Line tracking control & line maze solving strategies for the 3pi+ mobile robots,» inf. téc., 2023. dirección: [www.youtube.com/watch?v=WI1S3vNSuQ4](http://www.youtube.com/watch?v=WI1S3vNSuQ4).
- [14] B. Siciliano, L. Sciavicco, L. Villani y G. Oriolo, *Robotics: Modelling, Planning and Control* (Advanced Textbooks in Control and Signal Processing). London, UK: Springer-Verlag London Limited, 2010, MATLAB® is a registered trademark of The MathWorks, Inc., ISBN: 978-1-84628-641-4. DOI: 10.1007/978-1-84628-642-1.
- [15] *3pi+ 2040 Robot - Standard Edition (30:1 MP Motors), Assembled*, Pololu item #5001, <https://www.pololu.com/product/5001>, 2025.
- [16] P. Corke, *Robotics, Vision and Control*. Springer International Publishing, 2017, vol. 118, ISBN: 978-3-319-54412-0. DOI: 10.1007/978-3-319-54413-7. dirección: <http://link.springer.com/10.1007/978-3-319-54413-7>.
- [17] *Control para modelo unicycle*, Presentación Control de robots móviles con ruidas, 2024.
- [18] E. Sigma. «Garra robótica de aluminio.» Accedido: 29 de septiembre de 2025. dirección: <https://electronicasigma.com.gt/producto/garra-robotica-de-aluminio/>.

El primer repositorio contiene el firmware desarrollado para el ESP32 y programado en PlatformIO. Este código permite que el robot controle de manera simultánea las velocidades de los motores de tracción y el servomotor de la pinza, constituyendo la base de bajo nivel necesaria para la ejecución de las tareas de locomoción y manipulación.

<https://github.com/DianaAbrego/Firmware.git>

El segundo repositorio reúne el código de control implementado en *Matlab* para los robots Pololu 3Pi+ equipados con gripper. En él se incluyen los scripts responsables de leer las posiciones de los marcadores asociados a robots y cubos, y de enviar los comandos de velocidad y de control del gripper mediante comunicación TCP con mensajes codificados en CBOR, abarcando en particular los controladores utilizados para manipular los cubos 20, 30 y 21.

<https://github.com/DianaAbrego/Controlador-para-Pololu-3pi-.git>

El tercer repositorio pone a disposición los modelos 3D empleados en el proyecto, tanto de la pinza robótica tipo Fin-Ray como de las figuras geométricas utilizadas en las pruebas experimentales. Se incluyen los archivos del ensamble final de la pinza y las piezas correspondientes al cubo, cilindro, pentágono y pirámide, listos para su visualización, modificación o impresión en 3D.

<https://github.com/DianaAbrego/PIEZAS-3D.git>