

UNIVERSIDAD DEL VALLE DE GUATEMALA
Facultad de Ingeniería



**Implementación y validación de algoritmos de robótica de
enjambre en plataformas móviles en la nueva mesa de pruebas
del laboratorio de robótica de la UVG**

Protocolo de trabajo de graduación presentado por Rubén Alejandro
Lima García, estudiante de Ingeniería Electrónica

Guatemala,

2022

Resumen

En el siguiente protocolo se presenta una propuesta para trabajo de graduación enfocado en los algoritmos de inteligencia de enjambre y su implementación en plataformas móviles. En diferentes tesis anteriores se validaron los algoritmos a nivel de simulación y a nivel físico con plataformas físicas estáticas. En este trabajo se implementarán los algoritmos en diferentes plataformas móviles.

Para lograr el objetivo se van a migrar los algoritmos hacia estas plataformas y se reproducirán las pruebas hechas anteriormente en la nueva mesa de pruebas del departamento de robótica de la UVG. Finalmente se compararán los resultados obtenidos tanto entre plataformas móviles con los algoritmos evaluados como con los resultados obtenidos anteriormente.

Antecedentes

Robótica de enjambre

La robótica de enjambre (*swarm robotics*) es una rama de la robótica que enfoca sus estudios en desarrollar robots simples, escalables y con gran tolerancia a las fallas para realizar tareas complicadas, en lugar de robots robustos y complejos. Las claves de la robótica de enjambre son la reducción de costos y miniaturización ya que estas son unas de las mas grandes complicaciones que tienen los robots grandes y complicados [1].

Programmable Robot Swarms

Los científicos han estudiado los comportamientos de los animales durante mucho tiempo. Entre los mas interesantes están los de las hormigas y las abejas quienes tienen comportamientos colectivos que distribuyen las tareas de sus colonias en cada integrante de estas. Este comportamiento inspiró a los ingenieros del instituto Wyss a crear simples robots móviles que puedan trabajar colaborativamente tratando de lograr una meta general. Los “Kilobots” son un conjunto de 1024 robots con los que el usuario puede interactuar para realizar tareas complejas. Un ejemplo de uso sería utilizarlos de base para poder desarrollar un enjambre de robots-voladores con instrucciones para polinizar un sector en específico [2].



Figura 1: Kilobots [3].

Bluebots

En la vida marina se puede ver comportamientos muy sincronizados y complejos, uno de estos casos son los cardúmenes. El comportamiento sincronizado de estos los ayuda a conseguir comida, migrar e incluso escapar y protegerse de depredadores. Sin embargo, dentro de estos no hay un pez líder o una comunicación entre peces. Este tipo de comportamiento es llamado implícito, este denota que los peces individuales toman decisiones dependiendo de lo que ven que sus vecinos hacen. Este comportamiento ha fascinado a los ingenieros y científicos. En 2021 los investigadores del instituto Wyss y la escuela de ingeniería y ciencias aplicadas John A. Paulson (SEAS) desarrollaron un robot con forma de pez que puede sincronizar sus movimientos junto a otros robots en un Cardumen sin ayuda ni control externo. Los llamados “Bluebots” funcionan a base de un sistema guiado por visión de leds azules. Cada uno está equipado con 2 cámaras y 3 luces LED, las cámaras detectan las luces LED de los Bluebots vecinos y usan un algoritmo para determinar la distancia, dirección y hacia donde se dirigen sus contrapartes. Bajo este sistema los investigadores lograron representar diferentes comportamientos como concentración, dispersión y formaciones circulares [4].



Figura 2: Prototipo Bluebot, obtenido de [4].

Termite-inspired robots

Los mas grandes constructores de la naturaleza sin duda, son las termitas. Estos animales hacen estructuras de tamaño colosal siendo ellas diminutas, lo hacen en conjunto pero no hay ningún capataz, maestro de obras o comandante. Estos animales reaccionan a su alrededor para tomar la mejor decisión que ayude a la construcción de su termitero (hogar). Bajo esta idea el instituto Wyss desarrollo unos robots bastante simples que levantan pequeñas estructuras (bloques) y las utilizan para poder armar estructuras pedidas por algún humano. Estos robots pueden escalar las mismas estructuras que están creando para así poder hacer una estructura de un mayor tamaño que el del mismo robot. Estos robots actúan independientemente y no tienen conocimiento de qué está pasando fuera de su área inmediata. Esto hace que reaccionen a su ambiente tomando las mejores decisiones dependiendo de lo que puedan observar y entender [5].

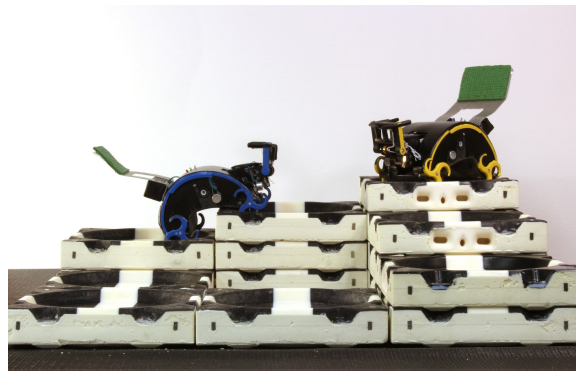


Figura 3: Robots termita junto a los bloques de construcción [5].

OFFensive Swarm-Enabled Tactics (OFFSET)

Los robots swarm también se ven incluidos en los planes de los ejércitos alrededor del mundo, tal es el caso del americano bajo la dirección del departamento avanzado de investigación para la defensa (DARPA, por sus siglas en inglés). El programa, llamado OFFSET está compuesto por una serie de robots en enjambre de más de 250 unidades. Estos se dividen en sistemas de aeronaves no tripuladas (UAS) y sistemas terrestres no tripulados (UGS) ya que se utilizan para distintas misiones en entornos urbanos muy complejos. El sistema utiliza esta tecnología para el desarrollo de tácticas para operaciones de campo con enfoque en la interacción humano máquina [6].



Figura 4: Robot terrestre UGS [7].

Megaproyecto Robotat

En la Universidad Del Valle de Guatemala se ha estado trabajando desde la perspectiva de robótica de enjambre desde hace ya un tiempo, uno de estos proyectos es el Robotat. Este constó de cuatro fases, en la primera fase se diseño y construyó una plataforma para el Robotat en donde implementaron un algoritmo de visión por computadora que regresa la posición y orientación del robot desde una perspectiva de dos dimensiones.

En la segunda fase, desarrollada por Aldo Aguilar [8], se utilizo el Particle Swarm Optimization, un algoritmo clásico de control. Este se modificó con la intención de implementarlo en robots diferenciales reales.

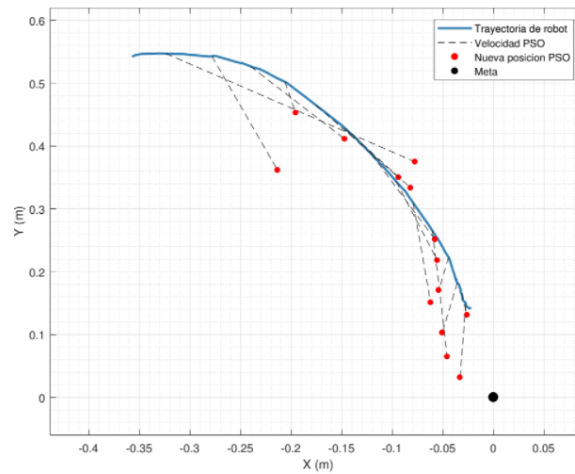


Figura 5: Trayectoria generada por marcadores PSO y controlador TUC [8].

La tercera fase, desarrollada por Eduardo Santizo [9] se enfoca en realizar mejoras al algoritmo PSO utilizando técnicas de aprendizaje reforzado y profundo. Para las mejoras se empleó el PSO tuner el cual consiste de una red neuronal recurrente la cual utiliza diferentes métricas propias de las partículas del PSO y las convierte, utilizando el procesamiento de una red LSTM, FRU o BiLstm en predicciones de los parámetros que el algoritmo debe utilizar. El resultado fue que la red BiLstm fue la mejor opción de las propuestas ya que redujo el tiempo de convergencia a sus mínimos locales. La fase IV fue desarrollada por distintos alumnos durante el 2021.

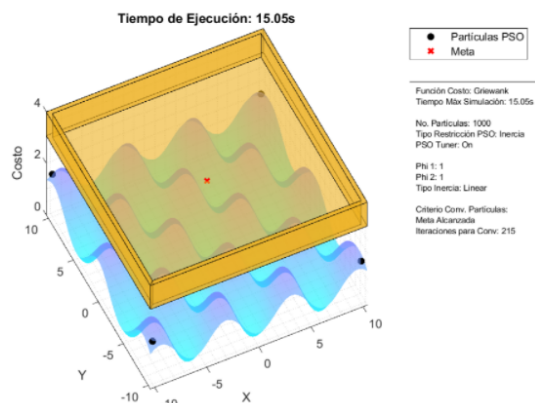


Figura 6: Partículas en las esquinas de la región de búsqueda luego de la ejecución de la primera prueba del PSO tuner con neuronas [9].

En la tesis desarrollada por Alex Maas [10] se implementó el algoritmo Modified Particle Swarm Optimization en sistemas físicos. Se validaron las pruebas hechas anteriormente a nivel de simulación utilizando métodos físicos un poco mas rústicos ya que no se contaban con los componentes necesarios. Se replicaron los resultados obtenidos bajo las mismas condiciones iniciales y se validaron estos en la ejecución física. Para lograr esto se desarrollaron dos sistemas de comunicación, el primero usaba un algoritmo de visión por computadora para detectar en tiempo real la pose de cada agente, el segundo permitía a los agentes intercambiar información que era requerida para que el algoritmo siguiera su correcto funcionamiento. Estos algoritmos fueron implementados usando programación multihilos ya que se necesitaba tanto recibir como mandar información paralelamente.

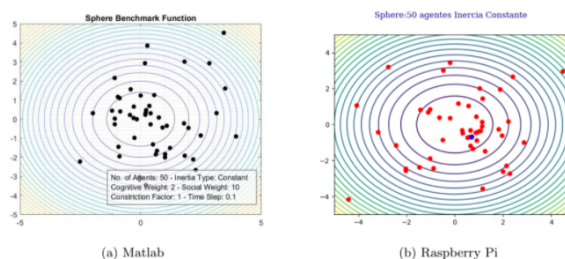


Figura 7: Partículas en posiciones iniciales [10].

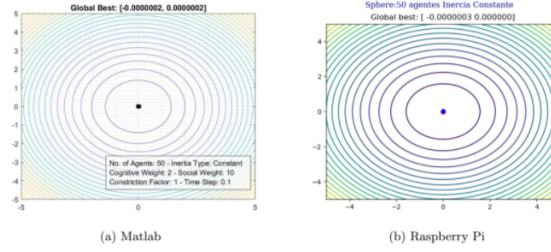


Figura 8: Partículas en posiciones finales [10].

Walter Sierra en su tesis [11], utilizo otro algoritmo para tener una referencia distinta, el *Ant Colony Optimization* (ACO). El migró el algoritmo a la plataforma Raspberry Pi utilizando el lenguaje de programación C. Las pruebas se hicieron usando métodos simples ya que no se contaba con una plataforma móvil y para validarlas se utilizo una plataforma de visión por computadora la cual permitía obtener la pose del robot. Las pruebas presentadas mostraron resultados en escenarios distintos tales como lo son, con y sin obstáculos en el mapa. Uno de los resultados mas importantes fue el lograr implementar un sistema dinámico en donde el algoritmo es capaz de recalculr la ruta al encontrar nueva información dentro de su área cercana, obstáculos o robots.

RUTA AL NODO 56	
Matlab	C++
1	1
12	12
23	23
34	34
45	45
56	56

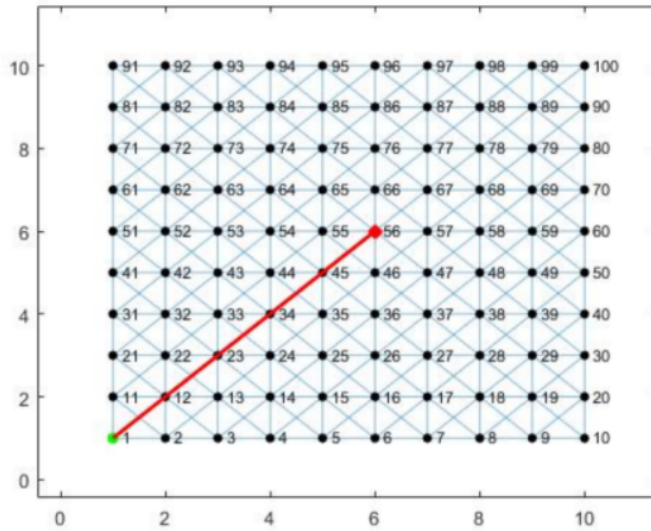


Figura 9: Ejemplo de Comparación de rutas entre matlab y C++ a un nodo específico [9].

Finalmente Julio Rodríguez [12], en su tesis esta realizando la tercera fase de una plataforma que se ha ideado desde 0 llamada Bytebot. La plataforma tiene capacidad de movilizarse, comunicarse por Wifi, recibir información sobre el desplazamiento y modificar su dirección, actualmente se esta implementando la detección y movilización de objetos. En un futuro cercano, esta plataforma se puede utilizar como opción para las pruebas físicas a realizar.

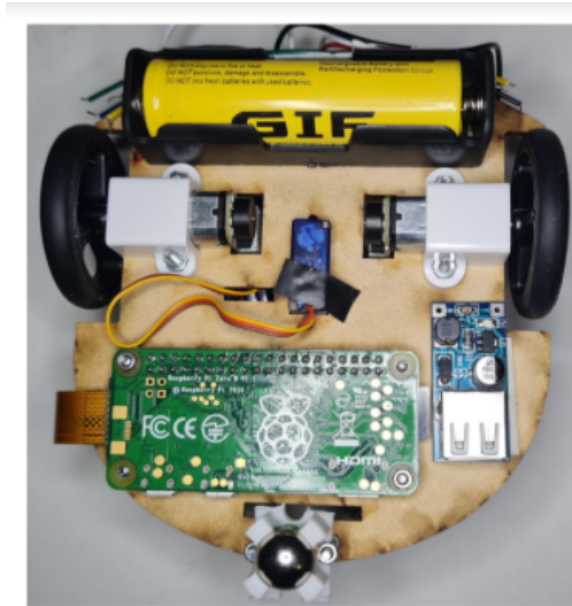


Figura 10: Vista Inferior del modelo impreso [12].

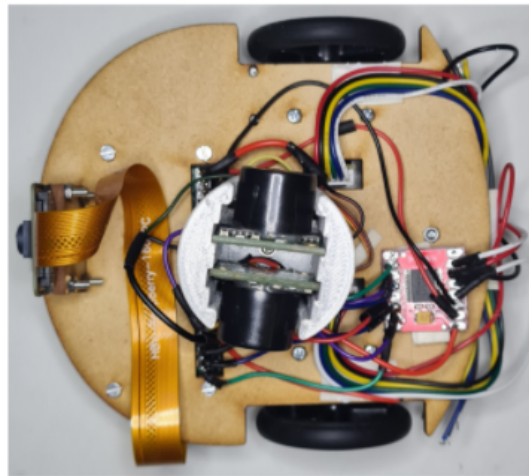


Figura 11: Vista Superior del modelo impreso [12].

Justificación

Los algoritmos de inteligencia de enjambre han inspirado distintas investigaciones y tesis dentro de la Universidad Del Valle de Guatemala y su Departamento de Ingeniería Electrónica, Mecatrónica y Biomédica. Para lograr poner a prueba estos algoritmos y su eficacia en el mundo real se necesitan plataformas móviles las cuales ya están disponibles.

En la fase anterior, a pesar de la migración de los algoritmos a plataformas físicas, estas no eran plataformas móviles autónomas por lo cual se debían mover manualmente hacia las posiciones deseadas. Debido a esto la meta es avanzar hacia la realización de un laboratorio de enseñanza e investigación que cuente con distintas plataformas móviles que ya hayan sido probadas y evaluadas sobre los algoritmos previamente diseñados para que puedan aplicarse en ambientes reales.

En esta tesis se busca utilizar las diferentes plataformas móviles a disponibilidad para poner a prueba los algoritmos de inteligencia de enjambre desarrollados por alumnos en tesis previas. Se migrarán los algoritmos a estas distintas plataformas para poder recrear pruebas y comparar resultados.

Objetivos

Objetivo General

Implementar y validar los algoritmos de robótica de enjambre desarrollados en fases previas usando plataformas móviles, en la mesa de pruebas de laboratorio de robótica de la UVG.

Objetivos Específicos

- Migrar los algoritmos desarrollados previamente para su ejecución en plataformas móviles con las que se cuenta actualmente.
- Replicar experimentos realizados en fases previas a nivel de simulación y con plataformas no móviles, esta vez usando plataformas móviles en la nueva mesa de pruebas de laboratorio de robótica de la UVG.
- Evaluar y comparar el desempeño de las distintas plataformas, así como los resultados de los algoritmos implementados.

Marco teórico

Robótica de enjambre

La robótica de enjambre es un nuevo acercamiento en robótica para lograr realizar la coordinación de un largo grupo de robots sin tener un control centralizado o ser robots de un alto nivel de entendimiento. Estos están inspirados en los grupos naturales de animales como lo son las hormigas o las abejas.

Un gran numero de pequeños y simples robots pueden realizar tareas complejas en un nivel de eficiencia muchísimo mayor a robots complejos pero en menor cantidad. Esto añade robustez y flexibilidad al grupo y sus usos.

Este estilo de robótica es diferente a la robótica-múltiple ya que, la robótica múltiple recava información de sus contrapartes y tiene un comando central, mientras que la robótica swarm, no tiene una central de información [13].

Robots Móviles

Un robot móvil es una maquina capaz de moverse a través de su entorno y no esta asegurado a una locación fija.

Esta es una rama de la industria bastante utilizada y en constante cambio e investigación ya que estos robots ayudan a empresas de almacenaje, logística, agricultura y muchas otras [14]. Entre los robots móviles están los autónomos (AMR) guiados autónomos (AVG) [15].

AlphaBot2-Pi Acc Pack

El Alfabot 2 es un kit robótico para la Raspberry Pi. Este kit incluye muchas funciones bases como lo son, seguimiento de linea, detección de obstáculos, conexiones inalámbricas (Wifi/bluetooth/control remoto/infrarrojo). Esta diseñado para no necesitar soldadura o un armado complicado y una programación de fácil comprensión.

En sus especificaciones incluye:

- Un regulador de voltaje LM2596 que provee a la Pi un voltaje estable de 5V
- Un chip TLC1543 que permite a la Pi usar sensores analógicos
- Un controlador servo PCA9685 para realizar rotaciones de manera mas rápida y sencilla
- Un convertidor UART CP2102 para controlar la Pi vía UART [16].



Figura 12: Alphabot 2 [16].

Raspberry Pi

La Raspberry Pi es una placa simple de bajo costo y muy alto rendimiento, básicamente una pequeña computadora. Fue desarrollada por la Raspberry Pi Fundación[17].

A medida que la Raspberry Pi fue ganando adeptos se fueron creando varios modelos diferentes, cada uno para tareas o necesidades diferentes. Algunos de los modelos son:

- Raspberry Pi 1 Modelo A (descontinuada)
- Raspberry Pi 1 modelo B (descontinuada)
- Raspberry Pi 2 modelo B
- Raspberry Pi 3 modelo A+
- Raspberry Pi 3 modelo B+
- Raspberry Pi 4 modelo B [18].



Figura 13: Raspberry Pi 4 Model B [18].

Software

Raspberry Pi OS es el software utilizado por la Raspberry Pi. Anteriormente llamado raspbian este sistema operativo tiene mas de 35,000 paquetes diferentes, previamente compilados para darle un mayor alcance al uso de la Raspberry Pi.

Este sistema operativo se puede descargar desde la pagina oficial de Raspberry Pi y tiene una guía completa de instalación [19].

Hardware

La Raspberry Pi cuenta con un procesador ARM con arquitectura RISC(Reduced Instruction Set Computer). Esta arquitectura utiliza instrucciones bastante simples por lo que en la ejecución el consumo de energía es mínimo [20].

Particle Swarm Optimization

Este algoritmo pertenece a una categoría de técnicas llamada optimización inteligente y es clasificado como un algoritmo estocástico de optimización basado en población. Este algoritmo simula un grupo de partículas con posiciones y velocidades iniciales asignadas, estas toman el nombre de *soluciones*. El algoritmo obtiene los 2 mejores resultados y la partícula al recibir esta información es actualizada. Uno de estos es llamado *local best*, que como su nombre indica es la mejor solución alcanzada por una partícula hasta el momento. El segundo es rastreado por el mismo algoritmo para ser el second best. Este proceso de obtención de resultados se repite hasta un numero especifico de iteraciones o se logre la convergencia. Esta convergencia es lograda al momento que todas las partículas son atraídas hacia la partícula con mejor solución global, llamada *global best*. Con estos datos es posible calcular los 2 principales factores a utilizar en la ecuación del PSO, llamados factor Cognitivo y Factor Social [21].

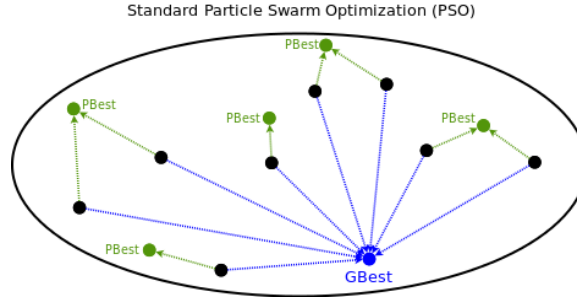


Figura 14: Representación de convergencia de partículas a su global best [21].

Ant Colony Optimization

El algoritmo de Ant Colony Optimization podemos resumirlo como un algoritmo con una red de agentes asíncronos que se mueve sobre diferentes estados o soluciones que llevan hacia un mismo punto y regresan la información a la central (El algoritmo) para que este determine la mejor ruta y así complete la solución al problema.

Ant Colony Optimization no es un algoritmo en sí, sino que es una clase de algoritmos. El primero fue el Ant System que fue propuesto por Coloni, Doeigo y Maniezzo. La idea principal, inspirada en las hormigas, es una búsqueda paralela sobre varias bifurcaciones dentro del sistema que llevan a un mismo punto. El sistema cuenta con una memoria dinámica que, a medida que los agentes manden información está pueda estructurarla y compararla con información de otros agentes obteniendo así la mejor ruta a la resolución del problema. Entre las variaciones está el Ant System (AS), Ant Colony System (ACS), Max-Min Ant System (MMAS), Ant-Q, Fast Ant System (FAS), Antabu, AS-rank y ANTS [22].

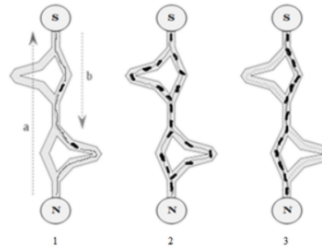


Figura 3: Proceso del algoritmo *Ant Colony Optimization* [14].

Figura 15: Funcionamiento del Algoritmo Ant Colony [22].

Metodología

Para evaluar las distintas plataformas móviles con los algoritmos de robótica de enjambre desarrollados en fases previas se propone:

- Comprender el funcionamiento de los algoritmos PSO y Ant Colony Optimization
- Migrar los algoritmos a las plataformas móviles a disposición.
- Hacer una evaluación de las pruebas hechas anteriormente y seleccionar las mas relevantes.
- Comprender el funcionamiento del programa de visión por computadora usado en la nueva mesa de pruebas del laboratorio de robótica de la UVG.
- Realizar la conexión entre la plataforma movil y el sistema de obtención de coordenadas instalado en la nueva mesa de pruebas del laboratorio de robótica de la UVG.
- Recibir, comprender y verificar las coordenadas de la plataforma movil recibidas desde sistema de obtención de coordenadas instalado en la nueva mesa de pruebas del laboratorio de robótica de la UVG.
- Aparte de las pruebas anteriores, rehacer las pruebas hechas con movimientos manuales en las fases anteriores.
- Luego de validar las pruebas anteriores, realizar pruebas propias buscando otros resultados para validar la conexión completamente.
- Obtener los resultados y compararlos con los encontrados en fases previas.
- Si es posible, comparar, usando diferentes plataformas los resultados a las mismas pruebas.

Cronograma de actividades



Figura 16: Cronograma primeras 4 semanas.

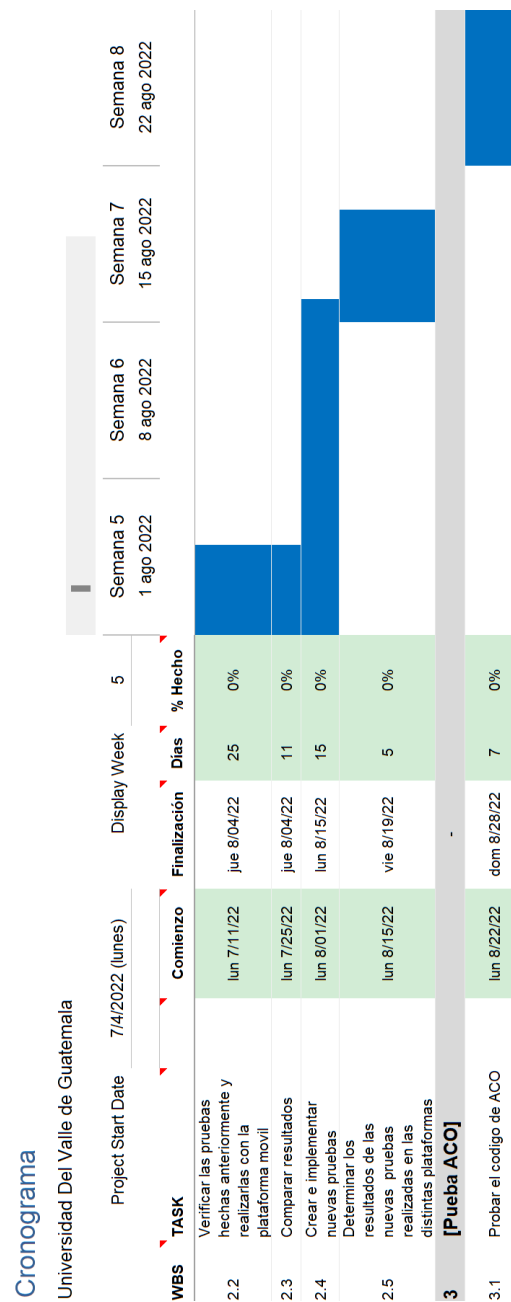


Figura 17: Cronograma segundas 4 semanas.



Figura 18: Cronograma terceras 4 semanas.

Índice preliminar

1. Prefacio	10
2. Lista de figuras	10
3. Lista de cuadros	10
4. Resumen	10
5. Abstract	10
6. Introducción	10
7. Antecedentes	10
8. Justificación	10
9. Objetivos	10
a) Objetivo general	10
b) Objetivos específicos	10
10. Alcance	10
11. Marco Teórico	10
a) Robótica de enjambre	10
b) Robots Moviles	10
c) AlphaBot2 Pi Acc Pack	10
d) Raspberry Pi	10
e) Software	10
f) Hardware	10
g) Particle Swarm Optimization	10
h) Ant Colony Optimization	10
12. Metodología	10
13. Resultados	10
a) Resultados pruebas conexión plataformas móviles	10
b) Resultados PSO	10
c) Resultados ACO	10
14. Conclusiones	10
15. Bibliografía	10
16. Anexos	10
17. Glosario	10

Referencias

- [1] M. Brambilla, “Swarm robotics: a review from the swarm engineering perspective,” *Swarm Intell*, págs. 1-41, 2013.
- [2] W. Institute. “Programmable Robot Swarms.” (2014), dirección: <https://wyss.harvard.edu/technology/programmable-robot-swarms/>.
- [3] —, “A self-organizing thousand-robot swarm.” (2014), dirección: <https://wyss.harvard.edu/news/a-self-organizing-thousand-robot-swarm>.
- [4] L. B. /. S. Communications. “Robotic swarm swims like a school of fish.” (2021), dirección: <https://wyss.harvard.edu/news/robotic-swarm-swims-like-a-school-of-fish>.
- [5] C. P. /. S. Communications. “Robots to the rescue.” (2014), dirección: <https://news.harvard.edu/gazette/story/2014/02/robots-to-the-rescue-2/>.
- [6] M. J. Parsons. “OFFensive Swarm-Enabled Tactics (OFFSET).” (2016), dirección: <https://www.darpa.mil/program/offensive-swarm-enabled-tactics>.
- [7] O. Networks. “Defense Intelligence Case Study – DARPA Offset Swarms for Raytheon BBN Technologies.” (2021), dirección: <https://www.oceusnetworks.com/defense-intelligence-case-study-darpa-offset-swarms-for-raytheon-bbn-technologies/>.
- [8] A. S. Aguilar, “Algoritmo Modificado de Optimización de Enjambre de Partículas (MPSO),” Tesis de licenciatura, Universidad del Valle de Guatemala, 2019.
- [9] E. Santizo, “Aprendizaje Reforzado y Aprendizaje Profundo en Aplicaciones de Robótica de Enjambre,” Tesis de licenciatura, Universidad del Valle de Guatemala, 2021.
- [10] A. D. Maas, “Implementación y Validación del Algoritmo de Robótica de Enjambre Particle Swarm Optimization en Sistemas Físicos,” Tesis de licenciatura, Universidad del Valle de Guatemala, Enero de 2022.
- [11] W. A. Sierra, “Implementación y Validación del Algoritmo de Robótica de Enjambre Ant Colony Optimization en Sistemas Físicos,” Tesis de licenciatura, Universidad del Valle de Guatemala, Enero de 2022.
- [12] J. A. Rodríguez, “Implementación y Validación del Algoritmo de Robótica de Enjambre Ant Colony Optimization en Sistemas Físicos,” Tesis de licenciatura, Universidad del Valle de Guatemala, 2022.
- [13] I. Navarro. “An Introduction to Swarm Robotics.” (2012), dirección: <https://www.hindawi.com/journals/isrn/2013/608164/>.
- [14] P. Prakash. “Types of Mobile Robots – What to use where?” (2020), dirección: <https://addverb.com/types-of-mobile-robots-what-to-use-where/>.
- [15] Intel. “Tecnología y casos de uso de robots móviles autónomos.” (2020), dirección: <https://www.intel.es/content/www/es/es/robotics/autonomous-mobile-robots/overview.html>.
- [16] *Alphabot 2 User Manual*.
- [17] Adafruit. “Raspberry Pi.” (2013), dirección: <https://www.adafruit.com/category/105>.
- [18] ThePiHut. “Official Raspberry Pi Products.” (2016), dirección: <https://thepihut.com/collections/raspberry-pi>.

- [19] RaspberryPi. “Raspberry Pi OS.” (2018), dirección: <https://www.raspberrypi.com/software/>.
- [20] R. Pi. “Raspberry Pi Hardware.” (2019), dirección: <https://www.raspberrypi.com/documentation/computers/raspberry-pi.html>.
- [21] C. Duarte y C. J. Quiroga, *PSO Algorithm*, 1.^a ed. Ciudad Universitaria, Santander, Colombia, 2010.
- [22] A. P. Engelbrecht, *Computational Intelligence: an introduction*, 1.^a ed. Pretoria, Sudáfrica, 2008.