

TESIS DE MAESTRÍA

Maestría en Ingeniería de Sistemas de Información UTN FRBA

Por Ing. Ignacio Javier Bonelli (tesista)
y Dr. Ing. Alejandro Hossian (director de la tesis).

Expositor: Ignacio Javier Bonelli

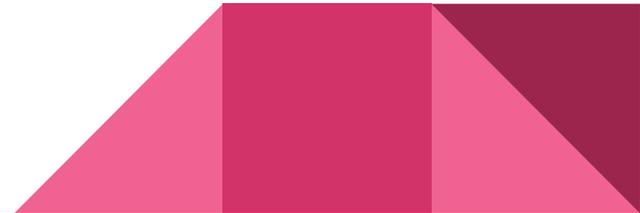
- Ingeniero en Electrónica (2001)
 - Docente de informática hace más de 20 años
 - Delivery Manager en weKnow (nearshore software development services)
 - Especializaciones en:
 - Redes de datos
 - Ecommerce B2B
 - Desarrollo Web & devOps
 - Miembro de la comunidad Drupal
 - Soluciones descentralizadas y blockchain
- 



Modelo Híbrido De Planeamiento De Caminos Para La Navegación Autónoma En Ambientes Desconocidos E Invariantes En El Tiempo

Publicaciones Científicas Vinculadas A Esta Tesis

Una versión reducida de este trabajo fue publicada en el XII Congreso Internacional sobre Aplicación de Tecnologías de la Información y Comunicaciones Avanzadas (ATICA2021) bajo el título “Propuesta de arquitectura híbrida para navegación autónoma de agentes robóticos móviles en ambientes desconocidos e invariantes en el tiempo”.



Planeamiento de caminos: ¿Por qué?

- Es una necesidad fundamental de cualquier robot que persiga un objetivo.
- Si no se conoce el mundo es difícil que un solo algoritmo permita alcanzarlo.

Planeamiento de caminos: Objetivo

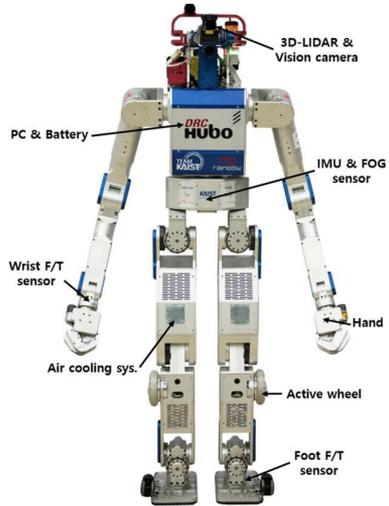
Crear un algoritmo novedoso que logre navegar de forma autónoma cualquier tipo de ambientes desconocidos e invariantes en el tiempo. Si no existe un camino, se debe declarar el ambiente sin solución.

Planeamiento de caminos: ¿Como?

Usando una Arquitectura híbrida:

- Capa reactiva APF + FollowPath
- Capa que genera información del ambiente con una simulación “LIDAR” configurable y el algoritmo Brushfire.
- Capa deliberativa que utiliza en ciertas circunstancias en A*

Tipos de Robots



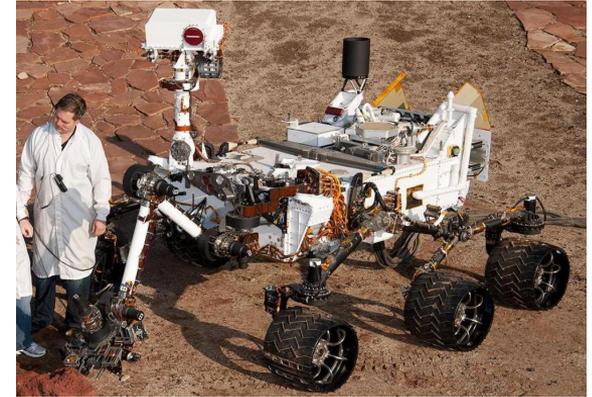
Hubo



Roomba



Spot



Mars Opportunity Rover



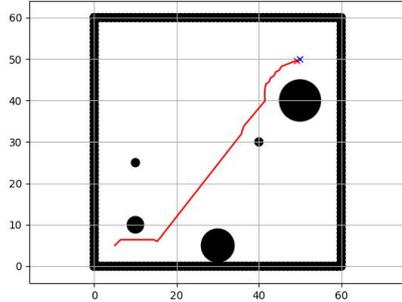
Depósito de Amazon

Algoritmos

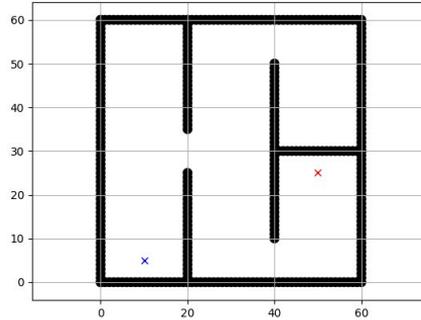
- Revisión sobre algoritmos de navegación, estrategias y arquitecturas.
- Necesidad de una arquitectura híbrida para lograr una navegación completa.
- Elección de los algoritmos base:
 - Reactivo: APF y FollowPath
 - Búsqueda: A*
 - Memoria: Brushfire
 - Deliberativo: Desarrollada de manera orgánica
- Problemas y falencias de los algoritmos elegidos.
- Búsqueda de ambientes que caracterizan los problemas a resolver.



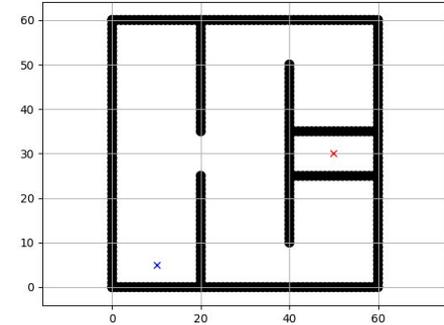
Caracterización de Ambientes



Ambiente abierto



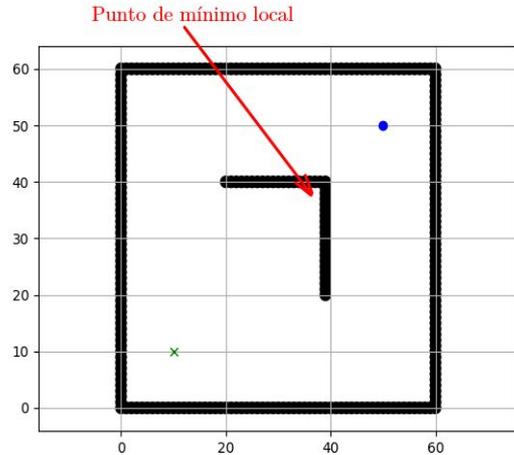
Ambiente cerrado
con solución



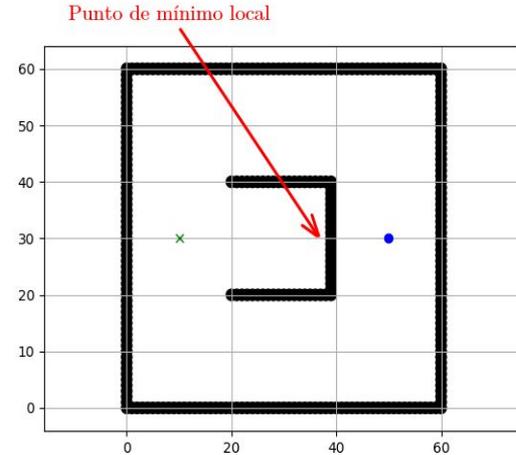
Ambiente cerrado
sin solución

Los mundos abiertos funcionan en general bien con soluciones de campos potenciales artificiales, los cerrados pueden necesitar soluciones de búsqueda exhaustiva.

Ambientes y problemas típicos



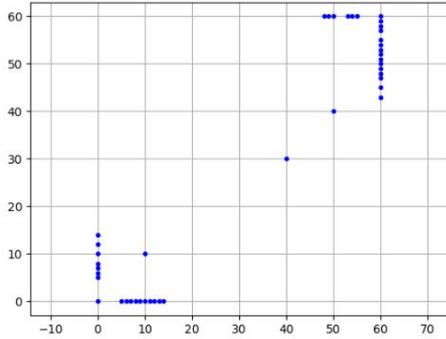
Trampa tipo 1



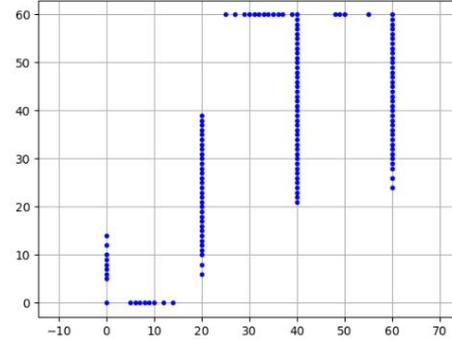
Trampa tipo 2

Ambiente y percepción

Simulación del sentido del ambiente



Sensado del Mundo 01



Sensado del Mundo 11

- Pasos LIDAR
- Límite de visión
- Objetos encontrados

El camino a una solución

- La necesidad de una arquitectura híbrida
- La elección de APF para la capa reactiva
- La necesidad de guardar información del ambiente
- La articulación con el algoritmo FollowPath
- La necesidad de plantear un objetivo local
- La capa deliberativa y la necesidad de Astar
- La arquitectura general de la solución



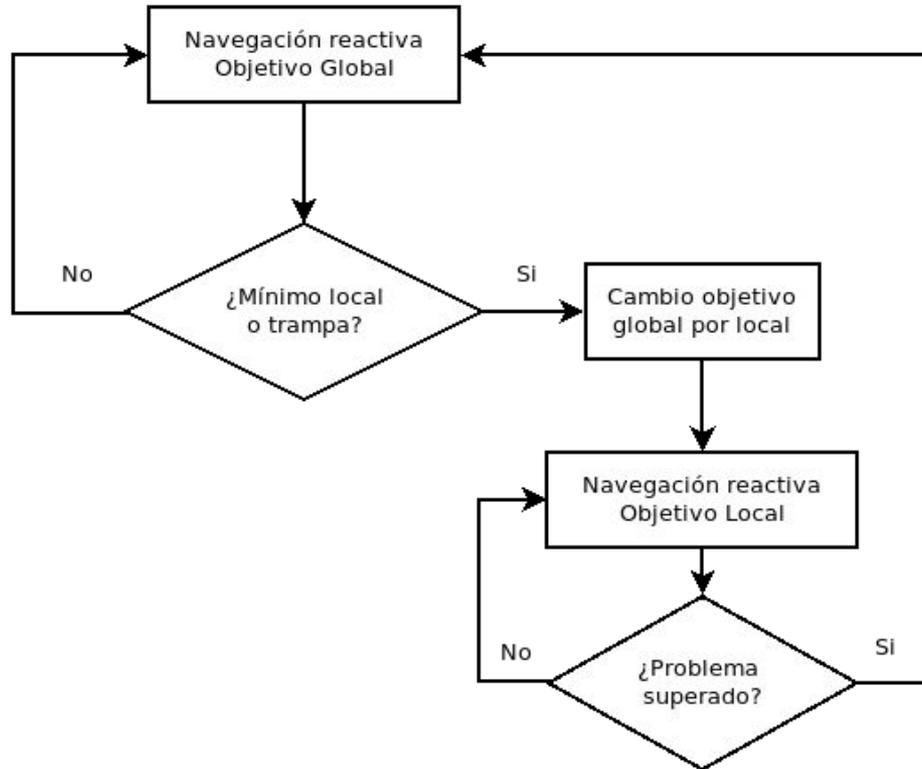
Modificación a A*

Objetivo: Evaluación del estado del mundo actual

1. Encontrar una camino a recorrer donde falte información del mundo, un nuevo objetivo local desde el cuál retomar la navegación reactiva.
2. Declarar el mundo sin solución (cuando ya se tiene información suficiente para hacerlo).



Sistema de navegación general

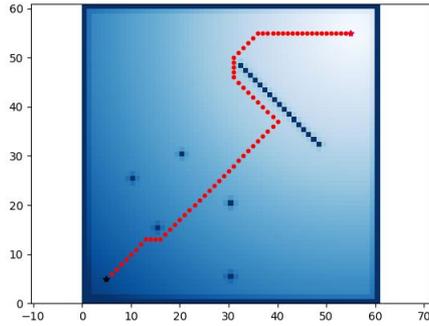


El sistema de pruebas y sus herramientas

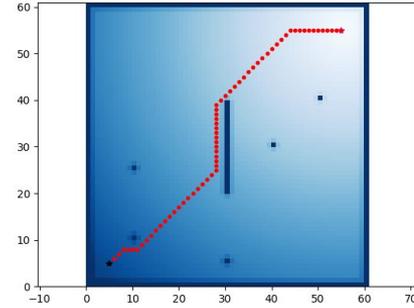
- Logs de cada subsistema:
 - Campos potenciales (APF)
 - Algoritmo Brusfire, Limites, y vista del mundo
 - Decisiones
 - Objetivos locales y globales
- Gráficos de visión y animaciones sobre la navegación.
- Diseño de mundos para prueba



Simulaciones ambientes abiertos



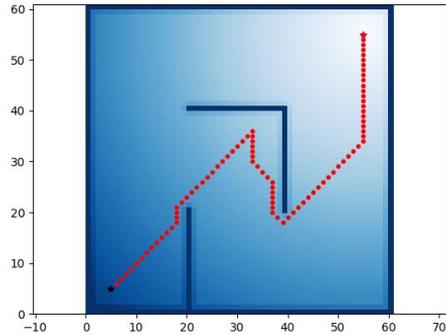
Recorriendo el mundo 3



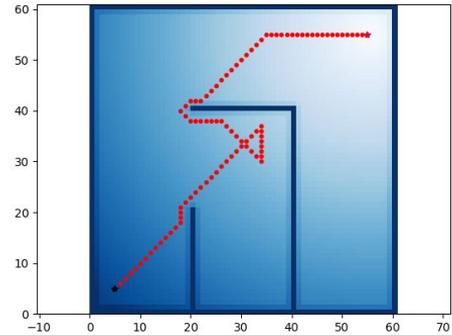
Recorriendo el mundo 4



Mínimos locales en simulaciones

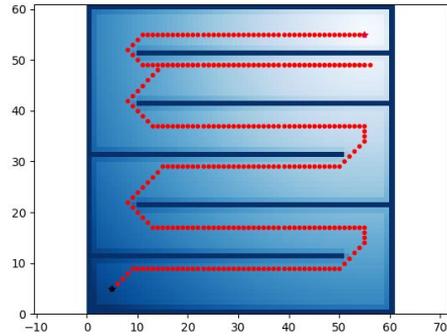


Recorriendo el mundo 15

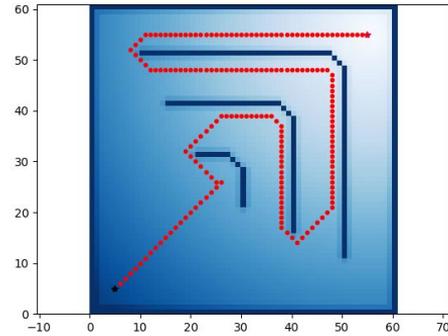


Recorriendo el mundo 21

Simulaciones y la búsqueda de información activa



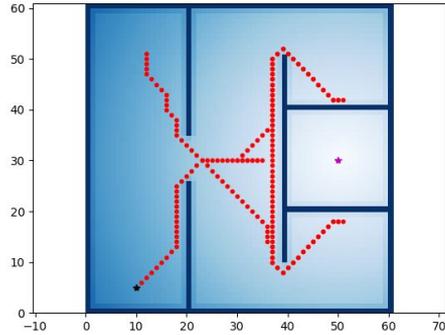
Recorriendo el mundo 22



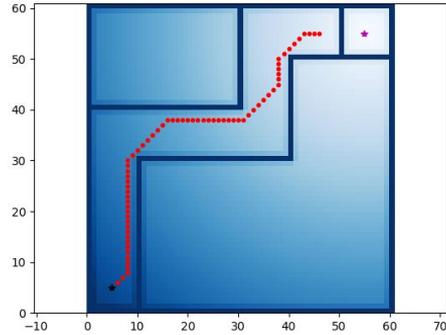
Recorriendo el mundo 24



Simulaciones en ambientes sin solución



Recorriendo el mundo 99



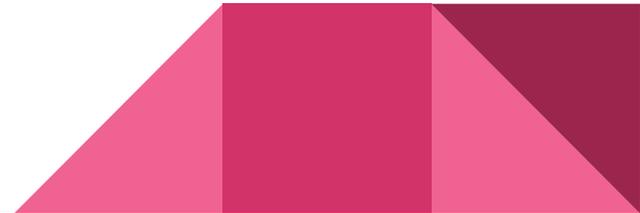
Recorriendo el mundo 97

Resultados

	Pasos		Dif.	%
	Híbrido	A*		
Mundo1	54	53	1	2%
Mundo2	56	53	3	6%
Mundo3	74	69	5	7%
Mundo4	65	62	3	5%
Mundo11	123	95	28	29%
Mundo12	86	71	15	21%
Mundo15	87	73	14	19%
Mundo21	93	73	20	27%
Mundo23	151	110	41	37%
Mundo13	232	222	10	5%
Mundo14	245	227	18	8%
Mundo22	341	237	104	44%
Mundo24	191	92	99	108%
Mundo31	57	46	11	24%
Mundo32	91	66	25	38%
Mundo33	64	60	4	7%

Revisión de objetivos Específicos

1. Presentar el área de conocimiento, su contexto y particularidades.
2. Encontrar algoritmos candidatos para formar una arquitectura híbrida.
3. Identificar distintos tipos de ambientes que permitan validar la arquitectura.
4. Ensayar distintas combinaciones de algoritmos para lograr el objetivo general.
5. Comparar los resultados de la arquitectura planteada con otros algoritmos.



Conclusiones y futuras líneas de investigación

- A medida que la complejidad del ambiente aumenta debemos combinar algoritmos para poder resolver cualquier tipo de ambiente.
 - A la hora de evitar las falencias de APF fue muy útil tener un algoritmo que siguiera un camino predeterminado (FollowPath).
 - La calidad de la información acumulada es muy importante para poder tomar buenas decisiones.
 - Hay todavía mucho trabajo por recorrer sobre la clasificación de mundos.
 - Existe una relación a explorar entre el radio de visión del robot y la eficiencia del algoritmo actual.
 - Investigar el funcionamiento de la combinación propuesta en este trabajo para que articule con un algoritmo de visión artificial.
- 

Repositorio de la Solución

- Repo: <https://github.com/ibonelli/pathplanning>
 - Solution: APF, FollowPath, BrushFire, AStar, Deliberative, Lidar & Data.
 - Tools: WorldBuilder, show_world, show_limits, show_map & show_path.



¡Muchas
gracias!

