



UNIVERSIDAD CARLOS III DE MADRID

DEPARTAMENTO DE INGENIERÍA DE SISTEMAS Y AUTOMÁTICA

TESIS DE MÁSTER

ALGORITMO GENÉTICO PARA LA LOCALIZACIÓN DE ROBOTS MÓVILES

Autor: Marcial Galván Sosa

Director: Luis Enrique Moreno Lorente

MÁSTER OFICIAL EN
INGENIERÍA ELÉCTRICA, ELECTRÓNICA Y AUTOMÁTICA

LEGANÉS, MADRID

JULIO 2008

UNIVERSIDAD CARLOS III DE MADRID
MASTER OFICIAL EN INGENIERÍA ELÉCTRICA,
ELECTRÓNICA Y AUTOMÁTICA

El tribunal aprueba la tesis de Master titulada
**“ALGORITMO GENÉTICO PARA LA LOCALIZACIÓN DE
ROBOTS MÓVILES ”** realizado por **Marcial Galván Sosa**.

Fecha: Julio 2008

Tribunal:

Dra. María Dolores Blanco Rojas

Dr. Luis Santiago Garrido Bullón

Dra. Beatriz López Boada



*A unos genes cabreados, ó:p:**



Índice general

Índice de Tablas	VII
Índice de Figuras	X
Agradecimientos	XIX
Resumen	XXI
Abstract	XXIII
1. Introducción	1
2. Los Algoritmos Genéticos	9
2.1. Algoritmos genéticos	14
2.2. Poblaciones e individuos. Codificación de la solución	14
2.3. Función de salud	16
2.4. Reproducción	17
2.5. Operador de Selección	17
2.6. Operador de Cruce	18
2.7. Operador de Mutación	20
2.8. Parámetros de un algoritmo genético	20

2.9. Pseudocódigo del Algoritmo Genético Canónico	21
2.10. Fundamento Matemático	22
3. El problema de la Localización Global	29
4. Propuesta de Solución	33
4.1. Modelado de los sensores	34
4.2. Modelado del entorno	34
4.3. Codificación de la solución	35
4.4. Población, elitismo y padres	36
4.5. Modelo de observación	37
4.6. Función de salud	38
4.7. Operador de Selección	40
4.8. Operador de Cruce	41
4.9. Operador de mutación	42
4.10. El Algoritmo	45
5. Datos Experimentales	49
5.1. Elección de los parámetros óptimos de funcionamiento del algoritmo	53
5.1.1. Dependencia con la Probabilidad de Mutación	54
5.1.2. Dependencia con la Probabilidad de Cruce	56
5.2. Análisis del comportamiento con los valores óptimos $P_m=0.4$, $P_c=0.8$	57
5.2.1. Robustez frente al ruido	58
5.3. Evaluación de la capacidad de convergencia en diversos entornos	60
5.3.1. Entorno 1: el hall, $x=40$ $y=60$ $\theta=0$	61
5.3.2. Entorno 2: los despachos, $x=305$ $y=10$ $\theta=90$	68
5.3.3. Entorno 3: el hueco de la escalera, $x=35$ $y=20$ $\theta=90$	74
6. Conclusiones	83
7. Líneas Futuras	87



Bibliografía

89



Índice de tablas

5.1. Densidad de Individuos para las poblaciones utilizadas	54
5.2. Porcentajes de acierto bajo los parámetros óptimos de ajuste, $P_c=0.8$ $P_m=0.4$	57
5.3. Variación del porcentaje de aciertos, las iteraciones medias y el promedio de soluciones ensayadas para la localización $x=40, y=60,$ $\theta=0$	66
5.4. Variación del porcentaje de aciertos, las iteraciones medias y el promedio de soluciones ensayadas para la localización en $x=305,$ $y=10, \theta=90$. Las tasas de acierto son sensiblemente inferiores al caso anterior, y el numero de soluciones ensayadas es muy superior.	70
5.5. Distribución del porcentaje de aciertos, iteraciones medias nece- sitadas y el promedio de soluciones ensayadas para la ubicación $x=35, y=20, \theta=90$ El porcentaje de aciertos para poblaciones pe- queñas es insignificantes debido a las dificultades intrínsecas del área de localización	79

Índice de figuras

2.1. Representación de una codificación binaria de una solución, izquierda, y de un segmento de una cadena de ADN, derecha . . .	15
2.2. Cruce básico. Se escoge un punto arbitrario, el mismo en los dos cromosomas, y se intercambian los genes de los dos padres a partir de dicho punto	19
3.1. Entorno de pruebas para el estudio de localización global.	30
3.2. Lectura del sensor en una ubicación dentro del entorno. Proporciona información en un plano 2D de las distancias que separan al robot de los objetos próximos. Se observan 4 lecturas a la derecha que están situadas en su máximo rango, es decir, no localizan obstáculos	31
3.3. Sistema de coordenadas. Se muestra la lectura de sensores de la localización $x=125$, $y=80$, $\theta=45^\circ$. Se puede ver la distribución de las lecturas de los sensores a lo largo de los 180° en los que estén distribuidos	32

4.1. Entornos utilizados en la localización. Debido a la variabilidad física que existe en el entorno hay que contemplar todas las posibles configuraciones que en las que se puede encontrar, fundamentalmente puertas abiertas y cerradas.	35
4.2. Distribución inicial de una población de 2000 individuos. En ausencia de información la distribución es aleatoria con probabilidad uniforme a lo largo de todo el espacio de soluciones	36
4.3. Area del pasillo, donde se observa la multiplicidad espacial de la misma forma, con pequeñas diferencias entre ellas. Esto da lugar a la existencia de mínimos locales a la hora de evaluar la función de salud	38
4.4. Función de salud evaluando una posición en la zona del pasillo, sin consideración del la componente de ángulo. Se observan múltiples mínimos que derivan ambigüedades en la localización	39
4.5. Evaluación de la función de salud en función de la orientación, se puede observar como en la orientación coincidente, en este caso 200°, la función nos ofrece un valor situado varios ordenes de magnitud por debajo del resto de las orientaciones	40
4.6. Cruce de los padres mediante mascara de cruce. Un 1 determina que el gen correspondiente será seleccionado del Padre A, un 0 del Padre B	42
4.7. Operación de cruce de dos padres en un espacio de dos dimensiones. Los padres, a y b, pueden generar como hijos a c y d, pero también, dependiendo de las probabilidades de cruce, a si mismos. La exploración del espacio de soluciones está muy limitada, es necesaria la introducción de la mutación	43

4.8.	La amplitud de la mutación está ponderada por una gaussiana cuya signa depende de la función de salud, a medida que el algoritmo converge las mutaciones tienen mas probabilidad de situarse en el entorno local del individuo, el efecto es el de hacer vibrar a la población en su entorno para conseguir un ajuste fino de la solución	44
4.9.	Convergencia en una zona donde solo existe una localización equivalente, las poblaciones están distribuidas alrededor del mínimo local (a la derecha) y del mínimo global (izquierda).	45
4.10.	Poblaciones distribuidas a lo largo del espacio de soluciones concentradas en mínimos locales equivalentes, en este caso las zonas de los despachos.	46
5.1.	Entorno de localización con la lectura de los sensores. El robot, señalado como el signo + rojo, está situado en las coordenadas $x=40, y=60, \theta=0$	51
5.2.	Histograma de soluciones. Muestra como se han distribuido las soluciones finales en el espacio a lo largo de las 100 iteraciones a las que se somete el algoritmo en cada test. En naranja se dibujan las soluciones en el eje X, en azul las del eje Y. En este histograma las soluciones están centradas en torno a $x=40, y=60$	52
5.3.	Evolución de los aciertos, con P_c constante, frente a variación de población y de P_m . Se puede apreciar como incrementar la probabilidad de mutación mejora la eficiencia del algoritmo	55
5.4.	Evolución de la tasa de aciertos frente a P_m constante ($P_m=0.4$) .	56

5.5. Distribución de las soluciones con una población de 2000 individuos a lo largo de 100 ejecuciones, a lo largo de todo el espacio de soluciones y detalle en la coordenada de localización es $x=40, y=60, \theta=0^\circ$, con un nivel de ruido del 2 %. La totalidad de soluciones se centra en torno a las coordenadas objetivo, aunque no todas cumplen los criterios de convergencia	58
5.6. Evolución de la tasa de aciertos frente al ruido en los sensores. La tasa de aciertos se mantiene en niveles por encima del 50 % con ruido de hasta el 40 %	59
5.7. Histograma espacial de las soluciones ofrecidas por el algoritmo y detalle para la localización en las coordenadas $x=40, y=60, \theta=0^\circ$, con $P_c=0.8$, $P_m=0.4$, población de 2000 individuos y un error introducido en los sensores de un 90 %. La totalidad de las soluciones se concentran en torno a las coordenadas objetivo, o en entorno equivalente, pero solo el 18 % cumple los criterios de convergencia.	60
5.8. Localización en las coordenadas $x:40, y=60, \theta=0$. Este entorno es de fácil localización, esta en una zona amplia, alejado de discontinuidades y no posee multiplicidad espacial	61
5.9. Distribución de los resultados a lo largo de 100 iteraciones con diferentes tamaños de población con la ubicación objetivo $x: 40, y=60, \theta=0$. A partir de 1000 individuos la localización en entornos equivalentes, que se observa en los test con poblaciones de 100 y 500, alrededor de la coordenada 400 en el eje X (naranja) es abandonada y la totalidad de las soluciones se centra alrededor de la coordenada objetivo, aunque no el 100 % cumple con los criterios de convergencia. Con 10000 individuos se obtiene un 95 % de aciertos	63

5.10. Distribución del número de aciertos y los tamaños de población frente a los tamaños de población. A partir de los 2000 individuos el porcentaje de aciertos permanece estable, pero disminuye el número de iteraciones necesarias. El número de posibilidades que se exploran es mucho mayor, y por lo tanto también los recursos consumidos	64
5.11. Evolución del incremento de soluciones ensayadas y del % de aciertos frente a la variación del tamaño de la población, para la localización $x=40$, $y=60$, $\theta=0$. no resulta eficiente incrementar la población a partir de los 3000 individuos, se disparan las operaciones a realizar sin tener recompensa en el porcentaje de aciertos.	65
5.12. Distribución de una población de 2000 individuos durante 15 iteraciones para la localización en $x:40$, $y=60$, $\theta=0$. La población converge rápidamente hacia las coordenadas objetivo y la localización equivalente. Se puede apreciar la convergencia en cruz producto del operador de cruce a partir de la iteración 8.	67
5.13. Localización en las coordenadas $x:325$, $y=10$, $\theta=90$. presenta múltiples entornos similares a lo largo del mapa	68
5.14. Distribución de los resultados a lo largo de las 100 iteraciones con diferentes tamaños de población con la ubicación objetivo $x: 305$, $y=10$, $\theta=90$. La existencia de múltiples mínimos locales con valores de salud muy próximos al de la coordenada objetivo hace que las localizaciones en el eje x permanezcan dispersas. si bien a partir de los 5000 individuos se abandonan casi todos estos mínimos locales las soluciones siguen estando distribuidas en entornos equivalentes. Con una población de 10000 individuos se obtiene un 40 % de aciertos	69
5.15. Áreas que presentan mínimos locales en el espacio de soluciones dentro del entorno.	70

5.16. Distribución del número de aciertos y de iteraciones frente a los tamaños de población en $x=305$, $y=10$, $\theta=90$. En este caso no alcanzamos una región de saturación en la curva de aciertos debido a la existencia de múltiples localizaciones equivalentes	71
5.17. Evolución del número de soluciones ensayadas y la tasa de aciertos, frente al incremento de la población en $x=305$, $y=10$, $\theta=90$. Existe una relación cuasilineal entre el número de soluciones y la tasa de aciertos	72
5.18. Evolución de las Tasas de acierto y el número de iteraciones con poblaciones ampliadas en la ubicación $x=305$, $y=10$, $\theta=90$, la tasas de aciertos tiende a estabilizarse en el entorno del 50 % debido a que el sistema oscila entre dos ubicaciones equivalentes.	73
5.19. Distribución de una población de 5000 individuos durante 15 iteraciones para la localización en $x: 305$, $y=10$, $\theta=90$. La población se agrupa en torno a los mínimos locales existente, los cuales no llegan ser abandonados en ningún momento, como se ve en la iteración 15	75
5.20. Localización en las coordenadas $x: 35$, $y=20$, $\theta=90$, tiene un mínimo local muy localizado y de difícil convergencia, con múltiples mínimos locales de valor similar	76
5.21. Distribución de los resultados a lo largo de 100 iteraciones con diferentes tamaños de población con la ubicación objetivo $x:20$, $y=35$, $\theta=90$. Al ser un máximo global muy localizado, un pozo muy abrupto en el espacio de soluciones, no se empiezan a obtener resultados significativos hasta alcanzar una población de 5000 individuos. Además, existe una ubicación equivalente en la coordenada $x=435$, y no existe localización sin ambigüedad hasta la población de 1000 miembros. La tasa de aciertos en este caso es del 33 %	77

5.22. Lectura de los sensores en las posiciones equivalentes. Difieren en la estimación de 5 sensores, 4 de ellos se encuentran en el límite de rango, y por lo tanto no contribuyen a la evaluación	78
5.23. Distribución del número de aciertos y de iteraciones frente a los tamaños de población en $x=35$, $y=20$, $\theta=90$. En número de iteraciones necesarias permanece casi constante mientras que el porcentaje de aciertos aumenta considerablemente con el tamaño de la población.	79
5.24. Evolución del número de soluciones ensayadas y la tasa de aciertos, frente al incremento de la población en $x=35$, $y=20$, $\theta=90$	80
5.25. Distribución de una población de 5000 individuos durante 15 iteraciones para la localización en $x=20$, $y=35$, $\theta=90$. A lo largo del pasillo existen múltiples mínimos locales próximos al mínimo global del espacio, de ahí la distribución de la población, y la longitud que alcanza la distribución en cruz producto del operador de cruce. Debido a la estrecha localización del mínimo global, mínimo valor de la función de salud, que implica una mutación en entorno, la población se agrupa en el entorno cercano de los brazos.	81

Agradecimientos

Gracias a Carmen, sin su apoyo, ayuda, comprensión y flexibilidad esta aventura no habría sido posible. ¡Gracias!

A Luis, por darme la oportunidad de trabajar con él.

A mi familia por su ánimo y estar siempre pendientes de mi.

A Michi, por enseñarme lo que es importante

A Héctor, Andrés y Alex por sus horas perdidas escuchando mis chorradas, y ayudándome en mis problemas.

A Sara, por saber escuchar.

A Reque, por sus sabios consejos.

A Jan, por tener siempre las cosas tan claras.

A Javier, por estar ahí en los momentos difíciles.

Y gracias, en general, a todos los que me han soportado durante este periodo, sin ellos, no habría llegado hasta aquí.

¡¡¡GRACIAS!!!

Resumen

La Localización Global de robots mòviles pretende responder a la pregunta ¿dónde estoy? dentro de un entorno conocido, siendo la posición inicial desconocida.

Los métodos de estimación bayesianos son una herramienta efectiva que suele utilizarse para resolver este problema, a partir de aproximaciones como los filtros de Kalman multihipotesis, métodos de Montecarlo o filtros probabilísticos basados en retícula entre otros.

La presente tesis de master ofrece una aproximación al uso de algoritmos genéticos a la resolución de dicho problema de una forma fiable y robusta. La selección natural, base de los algoritmos genéticos, ha demostrado ser una estrategia efectiva en la adaptación de los seres vivos a un entorno ruidoso, multimodal y cambiante, como es el planeta.

Mostraremos como esta estrategia también es efectiva en el campo de la robótica.

Abstract

The Global Localization of Mobile Robots aims to answer the question where am I? in an known environment, when the starting position is unknown.

The Bayesian estimation methods are an efficient tool that is usually applied to tackling this problem, based on approaches such as Kalman multihypotheses, Montecarlo methods or grid-based probabilistic filters.

This master thesis offers an approach to the application of genetic algorithms to solving the above problem in a fashion that is both reliable and robust.

Natural selection, the underpinning of genetic algorithms, is an efficient process in the adaptation of living beings to a changing, multimodal and noisy environment such as the planet.

We will show that this strategy is also efficient in the field of robotics.

Capítulo 1

Introducción

Esta Tesis de Master presenta una aproximación teórica a la aplicación del Algoritmo Genético Canónico, también conocido como Algoritmo Genético Simple, a la resolución del problema de la localización global de un robot móvil.

Los robots móviles autónomos se enfrentan a multitud de problemas a la hora de desenvolverse en su entorno, evitar obstáculos, orientarse, no dañar a los elementos que le rodean, manipular objetos, interactuar con personas y elementos, todos estos problemas son de una extrema complejidad y se va avanzando en su desarrollo. Pero el primer paso que tiene que dar un robot, es saber dónde está.

El objetivo de la localización global de robots es responder a esta pregunta, estimar su posición¹ y orientación, en un entorno conocido, en ausencia de cualquier información acerca de su posición inicial, respecto a un punto de origen, y utilizando los datos que el propio robot puede adquirir de su entorno. En líneas generales la localización en un entorno de tres dimensiones estará determinada por seis variables, tres para las coordenadas cartesianas, $[x,y,z]$ y tres para la posición angular respecto a cada uno de los ejes de coordenadas. Nuestro objetivo

¹en el presente trabajo usaremos posición para referirnos conjuntamente a la posición y orientación

es realizar esta localización en el plano, quedando estas seis variables reducidas a tres, $[x,y]$ que definen nuestra localización y θ , que define la orientación, a partir de las lecturas ofrecidas por un escaner láser de distancias.

Debemos hacer una distinción entre el problema de la localización global y el de la relocalización

El problema de la relocalización intenta mantener el seguimiento de la posición del robot, partiendo de la base de que ya hay un conocimiento previo, por lo menos aproximado, de su posición inicial, manteniendo la localización durante su itinerario.

La localización global intenta estimar la posición inicial del robot, con el conocimiento previo del mapa del área de localización, a partir de la información que éste obtiene de su entorno cercano mediante sus sistemas de adquisición de datos. Permite al robot localizarse a si mismo. Esto es un punto de partida importante a la hora de abordar la navegación en entornos conocidos, y proporciona información útil para la planificación de los primeros movimientos.

En lo concerniente a las técnicas de representación y estimación de la posición de robots, la gran mayoría de los algoritmos existentes están orientados hacia el problema del seguimiento. En estos casos, el pequeño error incremental que se produce a lo largo del movimiento, intrínseco a cualquier tipo de odometría, hace que aproximaciones clásicas, como el uso de filtros de Kalman (P. Jensfelt, 1999) sea aplicable. Si consideramos la estimación de la posición del robot como un problema recursivo de tipo bayesiano, los filtros de Kalman estimarían la distribución posterior condicionada a los datos recibidos por los sensores. Basándonos en la asunción de la existencia de ruido gaussiano, y de una incertidumbre inicial con una distribución igualmente gaussiana, los filtros de Kalman constituyen una solución eficiente para los problemas de relocalización. Sin embargo, la naturaleza de la incertidumbre inicial hace que los filtros de Kalman no sean robustos en los problemas de localización global y la no linealidad del espacio de soluciones implica la necesidad de una linealización del sistema

utilizando filtros de Kalman extendidos.

Diversas familias de algoritmos proporcionan una solución al problema de la localización global: métodos de localización de Montecarlo, filtros probabilísticos basados en retícula, filtros de Kalman multihipotesis y filtros de localización evolutivos entre otros. Estos métodos pueden ser incluidos dentro de el grupo mucho más amplio de los métodos de estimación bayesianos y necesitan incorporar el movimiento para la estimación de su posición.

Los filtros bayesianos (Thrun, Fox, Burgard, & Dellaert, 2001) parten de la prerrogativa de que el sistema tiene un comportamiento de Markov, el pasado y el futuro del sistema son condicionalmente independientes, y el futuro sólo depende del estado actual. La idea principal es estimar la distribución posterior de la posición del robot condicionada a la lectura de los sensores. Esta distribución se suele denominar estimación o *belief*

$$Bel(x_t) = p(x_t | d_{0..t}) \quad (1.1)$$

x representa el estado en el tiempo t , $d_{0..t}$ representa los datos desde el tiempo 0 hasta t . Generalmente para robots móviles se suele distinguir entre dos tipos de datos, los datos de los sensores, s_t , y los datos procedentes de la odometría, o_t

$$Bel(x_t) = p(x_t | s_t, o_{t-1}, s_{t-1}, o_{t-2}, \dots, s_0) \quad (1.2)$$

Un filtro bayesiano realiza esta estimación recursivamente, la estimación inicial caracteriza el conocimiento inicial acerca del estado del sistema, típicamente es iniciada como una distribución aleatoria a lo largo de todo el espacio de soluciones ya que la posición inicial es desconocida. Para obtener la estimación recursiva aplicamos la regla de Bayes a la ecuación 1.2, obtenemos:

$$Bel(x_t) = \frac{p(s_t | x_t, o_{t-1}, \dots, s_0) p(x_t | o_{t-1}, \dots, s_0)}{p(s_t | o_{t-1}, \dots, s_0)} \quad (1.3)$$

Como el denominador es una constante de normalización relativa a x , podemos expresar 1.3 como

$$Bel(x_t) = \eta p(s_t | x_t, o_{t-1}, \dots, s_0) p(x_t | o_{t-1}, \dots, s_0) \quad (1.4)$$

siendo

$$\eta = p(s_t | o_{t-1}, \dots, s_0)^{-1} \quad (1.5)$$

Como el filtro bayesiano asume la existencia de una independencia de Markov, la expresión 1.4 puede ser simplificada como:

$$Bel(x_t) = \eta p(s_t | x_t) p(x_t | o_{t-1}, \dots, s_0) \quad (1.6)$$

podemos expandir el miembro derecho de la expresión 1.6 integrando el estado en $t - 1$

$$Bel(x_t) = \eta p(s_t | x_t) \int p(x_t | x_{t-1}, o_{t-1}, \dots, s_0) p(x_{t-1} | o_{t-1}, \dots, s_0) dx_{t-1} \quad (1.7)$$

de nuevo aplicamos la asunción de Markov

$$p(x_t | x_{t-1}, o_{t-1}, \dots, s_0) = p(x_t | x_{t-1}, o_{t-1}) \quad (1.8)$$

Obteniendo

$$Bel(x_t) = \eta p(s_t | x_t) \int p(x_t | x_{t-1}, o_{t-1}) p(x_{t-1} | o_{t-1}, \dots, s_0) dx_{t-1} \quad (1.9)$$

Sustituyendo la ecuación básica de la estimación obtenemos la ecuación generalizada, que representa la actualización recursiva en los filtros de Bayes.

$$Bel(x_t) = \eta p(s_t | x_t) \int p(x_t | x_{t-1}, o_{t-1}) Bel(x_{t-1}) dx_{t-1} \quad (1.10)$$

Junto con la estimación inicial define un estimador de estado a partir de una

observación parcial del sistema. Para poder calcular la integral 1.10 es necesario conocer dos tipos de probabilidades, el modelo del sensor que depende del tipo de sensor usado

$$p(s_t | x_t) \quad (1.11)$$

Y el modelo de movimiento que es una generalización probabilística de la dinámica del robot, o densidad de probabilidad del estado siguiente

$$p(x_t | x_{t-1}, o_{t-1}) \quad (1.12)$$

en ambos hay que considerar la presencia de ruido.

La resolución de la ecuación es no trivial y de ahí las diversas soluciones propuestas.

Una de las más extendidas es la de la localización basada en métodos de Montecarlo(F Dellaert, 1999)(P Jensfelt, 2000)(Sebastian Thrun, 2001)(M.H. Kalos, 1986), en ella se representa la posición inicial del robot por un conjunto discreto de partículas distribuidas aleatoriamente a las que se les asigna un peso uniforme. En función de las medidas de entorno efectuado este peso, y no la ubicación de las partículas, cambia. En las siguientes iteraciones se redistribuyen las partículas, pero esta vez su distribución no es uniforme a lo largo de todo el espacio, sino aleatoriamente en el entorno de las partículas que en la evaluación anterior adquirieron mayor peso, creando una mayor densidad en las áreas donde se estima más probable la ubicación del robot, nuevamente se asigna a todas ellas un peso uniforme y se reevalúan en función de las medidas de los sensores.

Los algoritmos de localización basados en retícula (Thrun et al., 1999)(Wolfram Burgard, 1996) (D Fox, 1999)(J.Reuter, 2000) descomponen el espacio en áreas, que pueden ser o no uniformes, tanto para la localización como para la orientación. La resolución del reticulado es un punto crucial ya que de ella depende tanto la precisión como la velocidad de cálculo, así mismo la elección del

tipo de retícula también depende el tipo de sensores utilizado. A cada una de estas divisiones que les asigna una probabilidad, que es recalculada en función de las medidas del entorno que se obtienen. Todas las retículas han de ser evaluadas.

Los filtros de Kalman multi-hipótesis (K. Arras, 2002) (D.J. Austin, 2000) (I.J. Cox, 1994) (P. Jensfelt, 1999) (S.I. Roumeliotis, 2000) son aplicables en espacios continuos, representan las distribuciones usando una mezcla de gaussianas, habilitándolas para mantener el seguimiento de múltiples hipótesis, cada una de estas representada por una gaussiana diferente.

Los filtros de localización evolutivos (L. Moreno, 2006) están basados en el modelo de evolución diferencial propuestos por Storn y Price (R. Storn, 1997), trabajan con poblaciones vectores de soluciones, a las que se les asignan pesos, que iteran en el tiempo, generando nuevos individuos mediante el proceso de añadir la diferencia de dos vectores a un tercero. Si el nuevo vector presenta una mejor respuesta se conserva, con sustitución del que le dio origen, si no es así se descarta. La idea básica es extender mediante perturbaciones un vector ya existente.

Esta tesis de master propone la aplicación de un algoritmo alternativo basado en procesos evolutivos llamado Algoritmo Genético Canónico para solucionar el problema de la localización global de una forma eficiente y robusta. Como veremos más adelante el algoritmo puede acomodar cierto nivel de ruido y adaptarse a un espacio de soluciones no lineal y no continuo, operando directamente con datos que proporcionan los sensores, El principio de funcionamiento del Algoritmo Genético, AG, es representar la posición del robot mediante un conjunto de posibles soluciones, ponderadas por una función de salud, que interactúan recursivamente, evolucionando mediante selección, reproducción y mutación, hasta alcanzar una solución. Esta idea es similar a la utilizada en los métodos de Montecarlo, la diferencia estriba en que en la importancia del peso asociando a cada tipo de solución y en la forma en la que el conjunto de soluciones evoluciona a lo largo del tiempo mediante el uso de la información obtenida directamente de los sensores,

y en que nosotros no utilizamos información de movimiento, realizamos una estimación inicial de posición, para conseguir una estimación con un margen de error aceptable.

En el *Capítulo 1, Introducción*, se presenta el problema de la localización global y se muestran los métodos de estimación bayesianos más usados para su resolución y sus principales características.

En el *Capítulo 2, Los Algoritmos Genéticos*, se introduce al lector en el mundo de los algoritmos genéticos, y se le familiariza con los conceptos asociados y principales características.

El *Capítulo 3, El Problema de la Localización Global*, describe el entorno de pruebas y las características del robot elegido para el estudio.

En el *Capítulo 4, Propuesta de Solución*, se muestra la aplicación del paradigma de los algoritmos genéticos a la resolución de la localización global.

En el *Capítulo 5, Datos Experimentales*, se describe el ajuste de principales parámetros del algoritmo para posteriormente evaluar su comportamiento en tres áreas bien diferenciadas del entorno de pruebas con el fin de estudiar su fiabilidad y robustez.

Finalmente en *Conclusiones* y en *Líneas Futuras* se analiza el trabajo se proponen caminos para su extensión.

Pero antes de adentrarnos en nuestro problema, será mejor que veamos algo más de los algoritmos genéticos

Capítulo 2

Los Algoritmos Genéticos

Los algoritmos genéticos fueron desarrollados en la década de los 70 por John Holland (Holland, 1975) , sus colegas y sus estudiantes, en la Universidad de Michigan. El objetivo de sus investigaciones tenía dos vertientes: explicar el proceso adaptativo de los sistemas naturales y diseñar un sistema artificial que contuviera el mecanismo de la selección natural. Desarrollaron un algoritmo que forma parte de la denominada computación evolutiva, en él se toma como modelo de funcionamiento a la selección natural darwiniana, la recombinación de genes a través de la reproducción sexual, y puede ser usado en la resolución de un amplio espectro de problemas, siendo especialmente adecuado en problemas no lineales donde el espacio de soluciones es lo suficientemente extenso como para que otros tipos de búsqueda exhaustiva de soluciones resulten, en la práctica, inviables.

Existen varios tipos de enfoques en computación evolutiva, entre los que destacan el *hill-climbing* y el *simulated annealing*.

El *hill-climbing* tiene una comprensión bastante intuitiva, si representamos el espacio de soluciones como una superficie tridimensional, escogemos tres dimensiones por motivos de visualización, pero es aplicable a un espacio n-dimensional,

donde las mejores soluciones tienen un valor más elevado en el eje z , veremos cómo las mejores soluciones formarán colinas o altos picos en este espacio, rodeados de valles. El ascenso de la colina comenzará en algún punto del espacio y se moverá hacia arriba en esta superficie. Comienza con una o varias soluciones elegidas al azar, pero pueden ser previamente escogidas, depende de nuestro conocimiento previo del problema. Esta solución se muta, se le provoca un cambio aleatorio, si esta mutación proporciona una solución con mayor salud que la solución anterior, se conserva, en caso contrario se conserva la solución actual. Este tipo de algoritmos son conocidos como algoritmos voraces, lo que se refiere a que hará la mejor elección posible en cada paso, con la esperanza de obtener el mejor resultado global.

El *simulated annealing* está inspirado en procesos industriales donde un material se calienta por encima de su punto de fusión, y luego se enfría gradualmente para minimizar los defectos en su estructura. Existe también aquí una función de salud que determinará la adaptación de cada una de las soluciones. En este caso sólo existe una solución candidata que tendrá asociada una temperatura representada por una cantidad numérica global que disminuye gradualmente con el tiempo. En cada paso del algoritmo la solución muta y su salud es evaluada. El algoritmo decide si se queda con la solución madre o con la mutada, dependiendo de la temperatura que posea el sistema, si es alta puede llegar a elegir adaptaciones peores, al igual que ocurre con los materiales en estado de fusión en este estadio tampoco existirían unas posiciones de mínima energía preferenciales, pero a medida que se enfría es más exigente a la hora de adoptar un cambio. Cuando la temperatura llega a cero se detiene el proceso y se obtiene el resultado.

En los algoritmos genéticos la fuente de inspiración es la evolución darwiniana. ¿Qué ocurre en la naturaleza? Los seres vivos están dispersos en su entorno, en el que se desarrollan a lo largo de generaciones, los individuos más aptos logran reproducirse gracias a los mecanismos de presión de selección existentes,

resistencia a condiciones ambientales o enfermedades, habilidad para cazar,... Los mejor adaptados a su entorno de supervivencia ganan el derecho a la reproducción sexual, en la cual existe un intercambio genético entre dos individuos de la misma especie (recombinación genética). Como dichos individuos son, por adaptación al medio, los que mejor sobreviven en su entorno, se espera que sus hijos hereden características propias de sus padres que les permitan, a su vez, sobrevivir de forma óptima. Ocasionalmente se producen variaciones genéticas al azar en sus cromosomas, mutaciones, que pueden hacer que los individuos desarrollen mejores cualidades para su supervivencia, que surjan individuos inviables, o simplemente no provocar cambios. A lo largo de las generaciones globalmente la población estará mejor adaptada al medio, existiendo individuos cuya adaptación sea cuasióptima, y otros con una no tan buena. Este simple y bello mecanismo tiene una clara validación que podemos observar cada día a nuestro alrededor: la vida ha existido durante millones de años, con una asombrosa capacidad de adaptación y de supervivencia. Pero la aceptación de este método en la resolución de problemas no es solo por la belleza implícita que posee, los Algoritmos Genéticos han demostrado, teórica y empíricamente, un método robusto de búsqueda en espacios complejos. Desde la primera publicación al respecto, "Adaptation in Natural and Artificial Systems", (Holland, 1975), multitud de publicaciones han avalado esta técnica en la resolución de problemas de optimización y control.

Como ejemplo de problemas extremadamente complejos donde se ha aplicado algoritmos genéticos con éxito destacamos el diseño de una sala de conciertos (Sato, 2002) donde se conseguía maximizar calidad del sonido para la audiencia, el director y los músicos o el adiestramiento de redes neuronales para distinguir reflexiones sonoras desde distintos tipos de objetos (Porto, 1995): esferas metálicas, montañas submarinas, peces y plantas, y ruido aleatorio de fondo, es decir, detección de minas submarinas.

La filosofía de la selección, cruce y mutación es aplicada en los algoritmos

genéticos, de forma que se genera un conjunto de soluciones, cada una de ellas codificadas por ciertos elementos, por una serie de genes, y estas soluciones se distribuyen al azar en el espacio de soluciones. Mediante una función de salud se determina cuales de los individuos tienen mayor proximidad a la solución, estos son los que se podrán reproducir, mediante la combinación y mutación de sus genes, con la esperanza de que sus hijos evolucionen hacia espacios de solución donde encontremos el mínimo global que perseguimos, de forma que a lo largo de las generaciones se produce una convergencia global de la población hacia la solución.

Cuatro son las características que diferencian a este tipo de algoritmos de la mayoría de los procedimientos de búsqueda y optimización:

1. Los AG trabajan con una codificación de los parámetros de búsqueda, no con los parámetros en sí mismos
2. Los AG hacen una búsqueda mediante una población de soluciones, no con una sola solución
3. Los AG utilizan una función de optimización que se aplica directamente a los parámetros, sin necesidad de un conocimiento auxiliar acerca del espacio de soluciones
4. Las reglas de funcionamiento son probabilísticas, no deterministas.

Una de las principales ventajas que ofrecen es su funcionamiento intrínsecamente paralelo ya que se explora el espacio de soluciones en múltiples puntos a la vez, las áreas poco relevantes son rápidamente abandonadas y el algoritmo converge hacia las posibles soluciones. Esto constituye una ventaja en la exploración donde el espacio de soluciones es lo suficientemente extenso como para que un algoritmo de búsqueda exhaustiva no pueda resolverlo en un tiempo razonable, y especialmente cuando este espacio es no lineal. La no linealidad produciría una explosión combinatoria en la búsqueda de soluciones, inabordable en la práctica mediante métodos tradicionales.

Como veremos hay un alto componente de azar en cada uno de los pasos que se ejecutan, siendo su evolución no determinista, por lo cual no siempre llegaremos a la misma solución, aunque partamos de la misma población inicial, pero si, probablemente, a una buena solución frente a nuestro problema. Es posible que sea necesario aplicar un tratamiento estadístico a un conjunto de soluciones obtenidas para poder llegar a conclusiones válidas acerca del problema que estemos abordando.

Aunque el azar tenga un importante papel en el funcionamiento del algoritmo no hay que confundir este con una técnica aleatoria de búsqueda de soluciones. Los algoritmos genéticos son un ejemplo de uso de elecciones aleatorias utilizadas par una exploración exhaustiva del espacio de soluciones, pero de forma guiada. Usar la aleatoriedad en un proceso de búsqueda directa de soluciones puede parecer contradictorio en principio, pero es una herramienta efectiva que se usa en otros métodos, como el simulated annealing, para la búsqueda guiada de estados de mínima energía.

Una ventaja adicional de los algoritmos genéticos es que se desenvuelven bien en un sistema en el que los datos de entrada, y por lo tanto la función de salud, es ruidosa, discontinua, que cambia con el tiempo o que posee muchos óptimos locales, es decir, soluciones que se presentan como mejores a todas las similares a ella, pero que no son mejores a otras soluciones distintas situadas en otro lugar del espacio de soluciones.

Estas dos características, paralelismo y robustez frente al ruido, hacen de los algoritmos genéticos un buen candidato para resolver el problema de la localización global: es posible la existencia de muchos óptimos locales para una ubicación determinada (todas las habitaciones de similares dimensiones son virtualmente iguales), las señales de nuestros sensores poseen ruido y el entorno puede cambiar ligeramente con el tiempo (puertas que pueden estar abiertas o cerradas, por ejemplo).

En nuestro trabajo utilizamos la versión mas simple del algoritmo genético,

que utiliza tan solo tres operadores:

- selección
- cruce
- mutación

Para familiarizarnos con los conceptos básicos con los que vamos a trabajar veamos en detalle cada uno de los elementos y el funcionamiento de estos operadores, así como sus principales características.

2.1. Algoritmos genéticos

2.2. Poblaciones e individuos. Codificación de la solución

Vamos a trabajar con conjuntos de soluciones, denominadas poblaciones, y cada una de estas soluciones constituye un individuo. Trabajar con poblaciones de soluciones hace este tipo de algoritmos muy eficientes a la hora de trabajar con espacios de soluciones multimodales, con múltiples mínimos locales, evitando en buena medida la convergencia temprana hacia mínimos locales.

En el planteamiento original es necesario codificar cada uno de los parámetros de la solución mediante una secuencia finita de un conjunto finito de caracteres, de esta forma cada uno de los individuos estará representado por un conjunto de parámetros que a su vez están codificados mediante una serie de genes, nuestro conjunto finito de caracteres, que forma el cromosoma del individuo

En figura 2.1 se muestra una codificación binaria, que es el tipo de codificación tradicional para los algoritmos genéticos. Estos genes, 0's y 1's en este caso se agrupan formando el cromosoma, la expresión de este cromosoma, la solución que ofrece cada individuo, se denomina fenotipo.

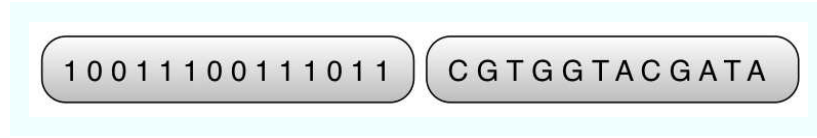


Figura 2.1: Representación de una codificación binaria de una solución, izquierda, y de un segmento de una cadena de ADN, derecha

El tratamiento de la solución en forma de cromosoma formado por genes es la base que permite utilizar la estrategia evolutiva en el algoritmo, ya que de esta forma puede ser procesada y combinada con otras soluciones con facilidad.

Existen diferentes enfoques a la hora de realizar la codificación, casi tantos como problemas a resolver. Podemos codificarlas como cadenas binarias (secuencias de 0's y 1's), cadenas de enteros, números decimales, etc., en estos últimos cada posición suele codificar una parte de la solución.

Los diferentes cromosomas, con sus diferentes fenotipos, se agrupan para formar una población que será un vector n -dimensional compuesto por i -individuos

$$I_i = [I_{i,1} \dots I_{i,g}] \quad g = \text{número de genes de cada individuo} \quad (2.1)$$

La codificación de la solución ha de ser robusta, de forma que al combinarse de forma aleatoria los genes de diferentes individuos se combinan no se produzcan soluciones carentes de sentido en nuestro espacio de búsquedas. Esto, en la práctica, puede ser imposible, así que se han de introducir mecanismos de control que aseguren la propagación de soluciones válidas.

La elección del tamaño idóneo de la población para que la convergencia tenga lugar en el menor tiempo posible y con la precisión deseada no es trivial, y será sujeto de estudio en nuestro problema.

2.3. Función de salud

Como en cualquier problema de optimización hay una función objetivo que hemos de minimizar, en el caso de los algoritmos genéticos esta función se denomina función de salud.

Otras técnicas de búsqueda requieren información auxiliar acerca del entorno del individuo para poder evaluar su validez, por ejemplo, las técnicas basadas en gradiente necesitan la evaluación de la derivada del hiperplano en el que se encuentra. Los algoritmos genéticos no, se consideran ciegos, tan solo necesitan la evaluación de los parámetros codificados en los genes del individuo mediante la función de salud en el proceso de búsqueda de la solución.

Este enfoque hace que los algoritmos genéticos sea adaptable a la resolución de un amplio rango de problemas, con otras técnicas de búsqueda un conocimiento exacto de la problemática asociada a la búsqueda de la solución es necesario, para la extracción de esa información auxiliar. En los Algoritmos Genéticos implemente se ha de diseñar una función de salud adecuada a nuestro objetivo.

Se ha de evaluar individualmente la capacidad de adaptación mediante la función de salud, de forma que a cada individuo se le asigna un número real que determina su capacidad de solucionar el problema. El objetivo es minimizar esta cifra.

$$salud = F, f(Sm, Se) \quad (2.2)$$

La elección de una buena función de salud no es trivial, al ser una función que debe ser diseñada para cada problema de forma específica, debe darnos realmente información de cuan cerca estamos de la solución, o de cuanto mejor es una solución frente a otra sin ambigüedades ni incertidumbre. Este valor determinará la probabilidad que tiene de que sus genes sean utilizados para generar la nueva población, o para ser elegido para la solución final, siguiendo con la filosofía de la selección natural esperamos que los mejores individuos dan lugar

a hijos con mayor probabilidad de supervivencia.

2.4. Reproducción

Mediante la reproducción los genes de ciertos individuos, denominados padres, se combinan para la generación de los hijos, con la esperanza de que hereden las mejores características de ambos progenitores y que formen una población que, globalmente, este mejor adaptada.

Computacionalmente el coste de este proceso es mínimo, lleva asociado la generación de números aleatorios y copia de cadenas numéricas, y se realiza utilizando, en su forma mas básica, que es la hemos implementado, tres operadores: selección, cruce y mutación.

2.5. Operador de Selección

La selección se encarga de escoger, de entre toda la población, a los individuos mejor adaptados para que sean los padres de la siguiente generación. Sus genes poseen la información necesaria para poder resolver el problema, y por lo tanto son merecedores de perpetuarse. A grandes rasgos podemos hablar de métodos preservativos, donde todos los individuos tienen una probabilidad de selección distinta de cero, y de métodos extintivos, donde hay individuos donde su probabilidad de selección es 0, dentro de esta división existen multitud de métodos de selección, algunos son mutuamente excluyentes, otros se pueden combinar entre si:

- *Selección elitista*: Los miembros mas aptos de cada generación tiene garantizado el paso a la siguiente generación. Este esquema es el único que garantiza alcanzar la convergencia en un problema de optimización (Fogel, 1992) (A. E. Eiben & Hee, 1991)

- *Selección proporcional a la salud*: los individuos mas aptos tiene mas probabilidad de ser seleccionados para la reproducción, pero no la certeza
- *Selección por ruleta*: La probabilidad de selección de cada individuo es proporcional a la diferencia entre su salud y la de sus competidores. Cuanto mejor es un individuo mas probabilidades tiene de ser seleccionado para la reproducción
- *Selección por torneo*: Se eligen dos subgrupos al azar de entre toda la población y los miembros de cada subgrupo compiten entre ellos. Sólo se elige un individuo de cada subgrupo para la reproducción
- *Selección por rango*: a cada individuo se le asigna un rango numérico basado en su capacidad de adaptación, y no en la diferencia con sus competidores. Esto puede evitar que individuos muy aptos inicialmente descarten a individuos menos aptos al principio, lo que podría provocar una convergencia errónea.

Una vez hemos seleccionado los padres que se van a reproducir para dar lugar a la siguiente generación hemos de generar a sus hijos, esto se realiza mediante recombinación genética, que puede ser efectuada solo entre los padres, o entre los padres y toda la población.

La recombinación de los genes se efectúa utilizando dos operadores, el de cruce y el de mutación.

2.6. Operador de Cruce

El operador de cruce es el encargado de combinar los genes de los dos individuos seleccionados para reproducirse. Se intercambia segmentos de genes, o genes individuales, entre los individuos, generando nuevos individuos denominados hijos.

El proceso es análogo al que realiza la naturaleza cuando combina los cromosomas en la reproducción sexual, como formas típicas de cruce tenemos:

- *Cruce básico*: Se selecciona un punto aleatorio dentro de la cadena de genes. Uno de los hijos hereda, del padre A la parte anterior a dicho punto y del padre B la posterior. Figura 2.2
- *Cruce multipunto*: igual que la anterior, pero escogiendo mas de un punto de cruce
- *Cruce uniforme*: Cada gen de los hijos tiene una probabilidad del 50 % de que pertenezca al padre A o al padre B
- *Cruce mediante mascara*: se genera una mascara del tamaño del número de genes, de 1's y 0's, según el valor que tenga ésta en cada uno de los genes se escogerá en gen de un padre u otro.

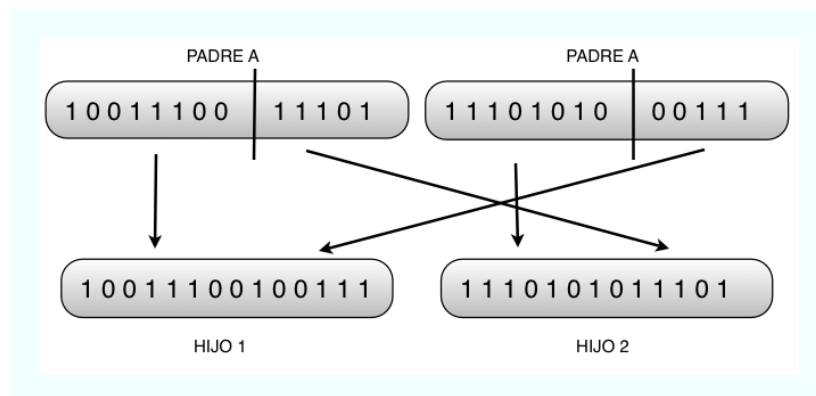


Figura 2.2: Cruce básico. Se escoge un punto arbitrario, el mismo en los dos cromosomas, y se intercambian los genes de los dos padres a partir de dicho punto

Al operador de cruce le está asociada una Probabilidad de Cruce, P_c , que determinara si, una vez seleccionados los padres A y B el intercambio genético se va a producir.

La operación de cruce es fundamental en la búsqueda de soluciones, ya que existe una transferencia de información entre los mejores individuos, de forma que, potencialmente, los hijos pueden heredar las mejores características de ambos padres, y ninguno de sus defectos.

2.7. Operador de Mutación

Con el fin de garantizar la variabilidad genética en nuestra población, que en la practica se traduce en permitirnos explorar mas extensamente el espacio de soluciones, se utiliza el operador de mutación.

Se encarga de introducir variaciones aleatorias en los genes, condicionada por una Probabilidad de Mutación, P_m , que actúa individualmente con cada gen, o grupo de genes. La elección adecuada de esta probabilidad de mutación, y de la de combinación es un factor importante que influye decisivamente el la eficiencia de nuestro algoritmo, y será objeto de estudio en nuestro desarrollo.

Esta operación se aplica a cada uno de los hijos que se generan, dando lugar cada uno de ellos a un hijo con mutación, H_m .

$$Hm_i \begin{cases} H_i + random & P_m < m_i \\ H_i & \text{otro caso} \end{cases} \quad i = [1..ngenesis]; \quad (2.3)$$

Donde m_i es un número escogido aleatóriamente con probabilidad uniforme en el intervalo $[0,1]$

2.8. Parámetros de un algoritmo genético

Existen una serie de parámetros que determinan el comportamiento del algoritmo, y por lo tanto debemos ser cuidadosos al escogerlos.

Tamaño de la población: una población pequeña puede hacer que el algoritmo no explore lo suficiente el espacio de soluciones como para ofrecer resultados

consistentes, una población demasiado alta puede sobrecargar innecesariamente el número de operaciones, y por lo tanto el tiempo, que necesita el algoritmo para converger.

Probabilidades de cruce y mutación: Determinan nuestra variabilidad genética y la evolución de nuestra población, por ejemplo, una P_m elevada puede hacer que nuestro algoritmo se comporte de forma aleatoria, perdiendo la ventaja de la evolución, incluso evitando la convergencia. Una P_c baja nos llevaría a estar atrapados en zonas del espacio sin posibilidad de salir de ellas.

El tipo de selección y elitismo: El elitismo (Fogel, 1992) (A. E. Eiben & Hee, 1991) se considera necesario para garantizar la convergencia de los algoritmos genéticos canónicos. Hay tipos de selección que provocan una hiperconvergencia prematura, ya que si surgen superindividuos son premiados por esta.

2.9. Pseudocódigo del Algoritmo Genético Canónico

El algoritmo genético canónico, también conocido como algoritmo genético simple, tiene el siguiente pseudocódigo, que es en el que nos basamos para nuestro algoritmo.

```
BEGIN Generar una población inicial
Evaluar la salud de cada individuo
WHILE NOT terminado Do
    BEGIN Generar una nueva población
    FOR Tamaño población / 2 DO
        BEGIN Ciclo reproductivo
            - Seleccionar dos individuos de la anterior generación,
              para el cruce (probabilidad de selección proporcional a
              la función de evaluación del individuo)

            - Cruzar con cierta probabilidad los dos individuos ob-
              teniendo dos descendientes.
```

- Mutar los dos descendientes con cierta probabilidad.
- Computar la función de evaluación de los dos descendientes mutados.
- Insertar los dos descendientes mutados en la nueva generación.

END

IF convergencia de la población **then** Terminado :=TRUE

END

END

2.10. Fundamento Matemático

Hemos visto de forma general e intuitiva el funcionamiento de los algoritmos genéticos, pero ... ¿por qué funcionan realmente?

Una idea acertada es pensar que existan características comunes en los individuos mejor adaptados que guían en la búsqueda de la solución. Los individuos están formados por genes, cadenas de caracteres de su alfabeto, ¿que similitudes hay en estas cadenas de unos individuos respecto a otros?

El elemento que nos auxilia en este sentido son los esquemas (Holland, 1968) (Holland, 1975), son estructuras similares que describen un subconjunto de la cadena con similitudes en ciertas posiciones.

Sin pérdida de generalidad vamos a considerar que trabajamos con un alfabeto binario $\{0, 1\}$, al que se añade un símbolo para poder especificar los esquemas, $*$, o el símbolo de "no importa el contenido". Obtenemos así un alfabeto ternario $[0, 1, *]$ que nos permite definir esquemas, que asocian individuos.

Por ejemplo, el esquema $*0000$ define un subconjunto de 2 cadenas, $[10000, 00000]$,

111 de cuatro [01110, 01111, 11110, 11111], * no es un símbolo del alfabeto, sino un metasímbolo que nos permite apreciar similitudes entre individuos.

Resumiendo, los esquemas nos permiten la estimación de similitudes entre cadenas de una longitud finita y sobre un alfabeto finito.

Para caracterizar los esquemas definimos su orden y su longitud.

El orden de un esquema H , $o(H)$, es en número de genes no variables que pose, por ejemplo, en el esquema $H[0***01]$, $o(H) = o(0***01) = 3$

La longitud, $\delta(H)$ será la distancia entre el primer y último gen fijo del esquema, de forma que $\delta_{max} = l - 1$.

$$\delta(0***01*) = 5$$

$$\delta(0*****) = 0$$

Siendo k el número de elementos de nuestro alfabeto que conforman la cadena de longitud l en general tendremos $(k+1)^l$ plantillas de esquemas de similitud, y k^l diferentes cadenas de longitud l , pudiéndose obtener k^l esquemas diferentes.

¿Cuántos esquemas pueden coexistir en una población? Si tenemos una población de n individuos el número de esquemas posibles estará contenido entre k^l y nk^l esquemas diferentes.

La introducción de figura del esquema nos permite un análisis de cómo afectan las operaciones de reproducción, cruce y mutación a la evolución de la población, y por lo tanto a la convergencia de la población

Los individuos con mejores niveles de salud son conjuntos con esquemas similares, estos esquemas definen los bloques constituyentes que se propagan de una generación a otra, incrementando o decrementando, como veremos, su propagación de forma exponencial, de una generación a otra. A este proceso se le considera paralelismo implícito.

Si consideramos que tenemos una población de n individuos I

$$I = g_1, \dots, g_l \quad (2.4)$$

siendo l la longitud de la cadena de genes que define el individuo de forma que los el conjunto I_j , $j = 1, 2, \dots, n$ formaran la población P_t en cada una de las iteraciones.

En primer lugar analizamos el efector que produce la reproducción. Imaginemos que en una población P_t existen momento particular existen m copias de un esquema H , $m = m(H, t)$. Durante la reproducción, un individuo I_j es seleccionado con una probabilidad

$$p_j = \frac{f_j}{\sum_n f_j} \quad (2.5)$$

En P_{t+1} esperamos que haya una evolución que proporciones $m(H, t + 1)$ copias del esquema.

$$m(H, t + 1) = m(H, t) \frac{f(H)}{\sum_n f_j} \quad (2.6)$$

siendo $f(H)$ el promedio de salud de los individuos pertenecientes a un esquema. El promedio de salud de la población global es

$$\bar{f} = \frac{\sum_n f_j}{n} \quad (2.7)$$

por lo que

$$m(H, t + 1) = m(H, t) \frac{f(H)}{\bar{f}} \quad (2.8)$$

Es decir, un esquema en particular crecerá, decrecerá, en función de ratio entre su promedio y el promedio global, los esquemas mejor adaptados se multiplican en la siguiente generación.

Para estimar que forma puede tener la evolución del número de promedios a lo largo de las iteraciones realizamos la suposición de que un determinado esquema posee un valor \bar{f} , siendo c una constante, de esta forma podemos

expresar 2.8 como

$$m(H, t + 1) = m(H, t) \frac{(\bar{f} + c\bar{f})}{\bar{f}} = m(H, t) (1 + c) \quad (2.9)$$

si comenzamos en $t = 0$ y asumimos un valor estable de c obtenemos

$$m(H, t) = m(H, 0) (1 + c)^t \quad (2.10)$$

El efecto de la reproducción 2.10 supone un incremento, o decremento, exponencial de los esquemas con un mayor, o menor, promedio.

Añadamos el efecto del cruce. Podemos interpretar el cruce como un intercambio de información entre individuos realizado de forma aleatoria. Durante este intercambio de información los individuos se segmentan, con lo que existe la posibilidad de que los esquemas se vean fragmentados en el proceso. Cuanto mayor longitud tenga un esquema mayor será su probabilidad de que la selección del punto de cruce lo escinda, la probabilidad de que se fragmente será

$$p_{fragmentacion} = \frac{\delta(H)}{l - 1} \quad (2.11)$$

y por lo tanto la de probabilidad de supervivencia

$$p_{sH} = 1 - \frac{\delta(H)}{l - 1} \quad (2.12)$$

y teniendo en cuenta que la realización del cruce también está ponderado por una probabilidad de cruce P_c .

$$p_{sH} \geq 1 - p_c \frac{\delta(H)}{l - 1} \quad (2.13)$$

El efecto combinado de la reproducción y el cruce es siguiente generación, P_{t+1} es el número esperado de esquemas por la probabilidad de supervivencia debida al cruce.

$$m(H, t + 1) \geq m(H, t) \frac{f(H)}{\text{prom}(f)} \left[1 - p_c \frac{\delta(H)}{l - 1} \right] \quad (2.14)$$

Esta expresión nos muestra que el número de esquemas seguirá creciendo, o decreciendo, depende ahora de dos factores, su valor promedio respecto al promedio de toda la población P_t y de la longitud de los esquemas. Los esquemas cortos y con un buen promedio se extenderán rápidamente.

Finalmente añadimos el efecto de la mutación, el cambio aleatorio de alguno de los genes que componen los individuos ponderado por una probabilidad p_m . Esta probabilidad se aplica de forma estadísticamente independiente a cada uno de los genes, individualmente la probabilidad de supervivencia de un alelo será $(1 - p_m)$, el esquema sobrevive si cada una de las $o(H)$ posiciones fijas permanece inalterada.

La probabilidad de supervivencia a la mutación, P_{sm} , es obtenida multiplicando la probabilidad de supervivencia por si misma $o(H)$ veces

$$p_{sm} = (1 - p_m)^{o(H)} \quad (2.15)$$

si $p_m \ll 1$ la expresión ?? se puede aproximar a

$$p_{sm} = 1 - o(H) p_m \quad (2.16)$$

con lo que concluimos que la expectativa de que un esquema obtenga un determinado número de copias en las siguiente generación, teniendo en cuenta los efectos combinados de la reproducción, el cruce y la mutación, podemos aproximar, despreciando términos menores del producto cruzado a

$$m(H, t + 1) \geq m(H, t) \frac{f(H)}{\text{prom}(f)} \left[1 - p_c \frac{\delta(H)}{l - 1} - o(H) p_m \right] \quad (2.17)$$

Esta expresión ?? denomina Teorema del esquema es el *Teorema Fundamental de los Algoritmos Genéticos*, nos dice los esquemas con promedio mayor que la

media, cortos y de bajo orden tienen un crecimiento exponencial a lo largo de las generaciones, es decir, en conjunto, nuestra P_t tendrá mejor salud, estará más cerca de la solución, que P_{t-1}

Capítulo 3

El problema de la Localización Global

Abordamos, de forma teórica, la localización global en un entorno conocido, la pregunta que se ha de responder es ¿Donde estoy?, una cuestión que puede ser trivial para un humano, pero que entraña serias dificultades para un algoritmo.

Se ha elegido como entorno de prueba una planta tipo del edificio Agustín de Betancourt, figura 3.1, perteneciente a la Universidad Carlos III de Madrid, situado en el Campus de Leganés. Este entorno posee diversas características que lo convierten en una buena zona de pruebas, presenta zonas con alta repetibilidad, como la zona de los despachos, en las que nos encontramos con una multiplicidad espacial de un mismo contorno con pequeñas diferencias entre ellos. El pasillo, también representa una zona de difícil localización ya que presenta casi el mismo aspecto en toda su extensión. También nos encontramos con zonas en las que la localización es, en principio, más sencilla. Además, existe variabilidad en el entorno de localización, como las puertas o muebles que pueden cambiar de posición. Todo ello nos permite comprobar el funcionamiento del método en diferentes y muy variadas condiciones.



Figura 3.1: Entorno de pruebas para el estudio de localización global.

Otra ventaja añadida en la elección de este entorno es la posibilidad futura de poder efectuar la comprobación experimental de la eficiencia de nuestro método: el robot disponible nos permitiría la implementación online del algoritmo propuesto y la verificación real de los resultados.

El robot en el que nos hemos basado es Manfred (MAN FRiEnDly mobile manipulator), un robot móvil autónomo perteneciente al Laboratorio de Robótica de la Universidad CarlosIII de Madrid.

Dicho robot posee como sistema de adquisición de datos del entorno un escaner de distancia basado en láser. El sensor realiza un barrido a lo largo de 180° . Por cuestiones de coste computacional no utilizamos la totalidad de los datos, escogemos una medida cada 3° , 61 medidas en total que proporcionan información unidimensional acerca de la distancia que lo separa de los obstáculos que tiene alrededor, al disponer de un array perimetral obtenemos información bidimensional, 2D, generando un plano de distancia de su entorno, Figura 3.2. El alcance operativo estimado de los sensores es de 50 metros, pero solo se consideran válidas las medidas situadas en los primeros 15 metros. Uno de los objetivos es poder trabajar directamente con los datos del sensor, sin procesado de ningún tipo, estas lecturas serán la única fuente de información exterior con las que contará el algoritmo a la hora de realizar la localización.

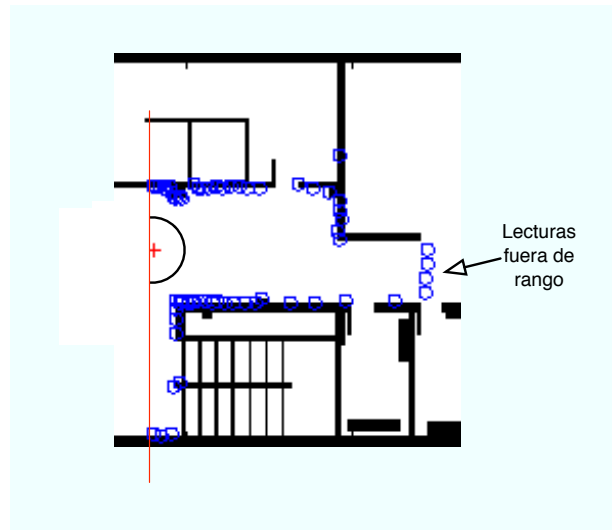


Figura 3.2: Lectura del sensor en una ubicación dentro del entorno. Proporciona información en un plano 2D de las distancias que separan al robot de los objetos próximos. Se observan 4 lecturas a la derecha que están situadas en su máximo rango, es decir, no localizan obstáculos

Resumiendo, nos encontramos en un entorno amplio en el que debemos localizar nuestra situación a partir de un mapa bidimensional de obstáculos obtenido a partir de unos sensores de distancia, dentro de un entorno del que ya poseemos el mapa.

La pregunta que inicialmente abordábamos ¿Dónde estoy? ahora la hemos transformado, gracias a esta adquisición de información del entorno cercano al robot, en ¿Dónde encaja nuestro mapa bidimensional dentro del entorno?

El resultado de la localización, quedará definido por tres parámetros, dos nos proporcionan información acerca de la ubicación en el plano, en coordenadas cartesianas, (x, y) , referidas a un punto de origen, $(0,0)$, y una orientación θ , ver Figura 3.3

El entorno tiene unas dimensiones de 60.5×14.5 metros, aproximadamente 878 metros cuadrados, y nos planteamos el objetivo de conseguir una localización con un error menor a $\pm 12,15$ cm, como veremos mas adelante dividiremos

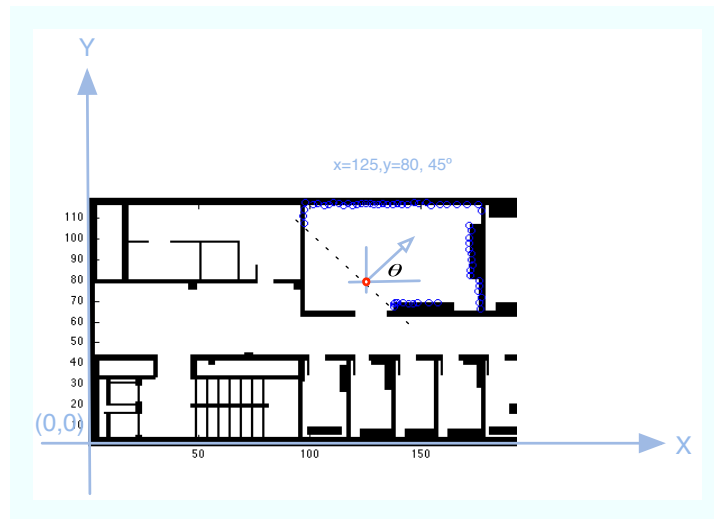


Figura 3.3: Sistema de coordenadas. Se muestra la lectura de sensores de la localización $x=125$, $y=80$, $\theta=45^\circ$. Se puede ver la distribución de las lecturas de los sensores a lo largo de los 180° en los que estén distribuidos

el área en celdas con esa dimensión de arista para efectuar la localización. Esto se traduce en un error del 0.2 % en el eje x y del 0,83 % en el eje y en la orientación deseamos un error menor a $\pm 3^\circ$, un 0.8 %.

Capítulo 4

Propuesta de Solución

Nuestro objetivo es la búsqueda de un método alternativo al uso de filtros bayesianos en la resolución del problema de la localización global de robots.

Optamos por la utilización de algoritmos genéticos ya que se adaptan a las características de nuestro problema:

- Nos enfrentamos a un espacio de soluciones extenso y no lineal
- Existen múltiples mínimos locales
- El sistema de recogida de datos es de buena calidad, introducimos ruido en el modelo de forma artificial.
- Queremos tener la capacidad de trabajar con los datos en bruto de los sensores

Aplicamos en nuestro problema el esquema general de organización de un algoritmo genético que fue expuesto en el capítulo 2, ahora veremos como lo hemos adaptado con el fin de ejecutar con éxito la localización global.

4.1. Modelado de los sensores

El sensor láser es el único dispositivo de toma de datos del exterior que se considera en este caso. Gracias a la capacidad de trabajar directamente con los datos recogidos por los sensores no es necesario introducir ningún tipo de pre-procesado en la señal. La medida tendrá la forma

$$Sm = (Sm_1..Sm_{ns}); \quad ns = 1..N \text{ medidas} \quad (4.1)$$

disponiendo de un vector de distancias perimétricas en 180° distribuidas a lo largo del semiperímetro frontal del robot.

4.2. Modelado del entorno

Dos aspectos fundamentales a la hora de diseñar un sistema de localización son como representar la información acerca del entorno del robot, y como representar su localización. Se ha optado por una representación en mapa de retícula, en la que el área está dividida en celdas. Segmentamos el mapa en una retícula de 12.1 cm de arista, siendo la unidad básica de medida que representamos en los mapas de localización mostrados.

Representamos los obstáculos existentes en el entorno mediante segmentos en el plano, que representan cualquier tipo de objetos que puedan encontrarse en el terreno, como muros o puertas.

Debido a la existencia de elementos móviles naturales en el entorno, como las puertas, utilizamos dos mapas para la localización, uno con las puertas cerradas y otro con las puertas abiertas para la evaluación de la función de salud. Figura 4.1

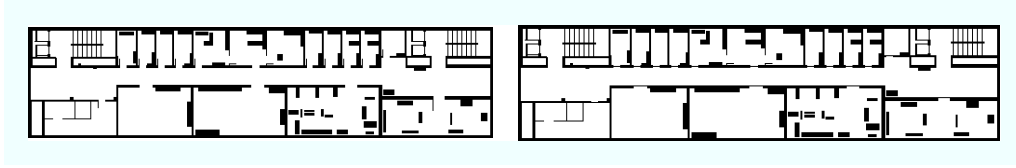


Figura 4.1: Entornos utilizados en la localización. Debido a la variabilidad física que existe en el entorno hay que contemplar todas las posibles configuraciones que en las que se puede encontrar, fundamentalmente puertas abiertas y cerradas.

4.3. Codificación de la solución

El primer es paso establecer una forma adecuada de codificación de la solución, su expresión en los genes de los individuos. La solución se encuentra en un espacio continuo euclídeo multidimensional, por lo que la opción tradicional de codificación mediante una secuencia de caracteres discretos, $[0, 1]$, es de compleja aplicación. Adoptamos una representación del espacio real (Z. Michalewicz, 1991) mediante una 3-tupla, cada individuo sera

$$I = (x, y, \theta) \quad (4.2)$$

Donde x representa la coodenada en el eje X, y la coordenada en el eje Y y θ , la orientación espacial.

Los Algoritmos Genéticos Canónicos en el Espacio Continuo poseen algunas características que no tienen los aplicados al Espacio Discreto, para una discusión detallada ver *Theoretical Analysis of Evolutionary Algoritms With an Infinite Population Size in Continous Space Part I and Part II*(Xiaofeng Qi, 1994)

Esta codificación carece de ambigüedades, ya que dos soluciones diferentes no pueden estar referidas al mismo punto, y disponemos de tres genes para realizar la reproducción. Ademas nos permite identificar facilmente aquellas soluciones generadas que queden fuera del plano de soluciones, descartarlas y generar otras válidas.

4.4. Población, elitismo y padres

En ausencia de cualquier tipo de información, y este es nuestro punto de partida, el robot puede estar situado en cualquier posición en el entorno, a priori no podemos descartar ningún lugar del espacio de soluciones como solución correcta. Por lo tanto la distribución inicial de la población es aleatoria con una probabilidad uniforme a lo largo de todo el espacio. Figura 4.2

Se genera un conjunto de N individuos, la población, tal que

$$P = (I_1 \dots I_N); \quad (4.3)$$

de forma que obtenemos un vector

$$I_i = [I_{i,x}, I_{i,y}, I_{i,\theta}]; \quad i = 1 \dots N \quad (4.4)$$

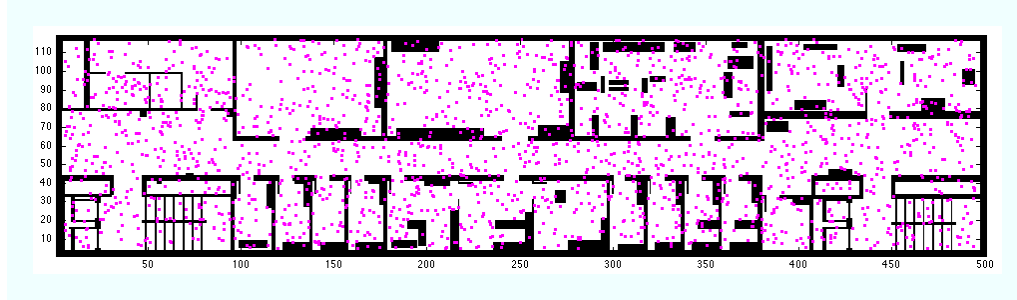


Figura 4.2: Distribución inicial de una población de 2000 individuos. En ausencia de información la distribución es aleatoria con probabilidad uniforme a lo largo de todo el espacio de soluciones

Para garantizar la convergencia del algoritmo es necesario implementar elitismo (Fogel, 1992) (A. E. Eiben & Hee, 1991) en nuestro algoritmo. El 2.5 % de los individuos mejor adaptados constituirán la élite, el 25 %, que engloba también a la élite, los padres. La élite tiene garantizado el paso a la siguiente generación sin

sufrir ningún tipo de variación. La elección de un tamaño dependiente del tamaño global de la población persigue evitar la convergencia temprana hacia un mínimo local. Debido a la existencia de zonas con una alta repetibilidad, un tamaño de élite pequeño, o de padres, puede causar una convergencia prematura hacia un mínimo local si alguna solución cae cerca de este y se torna dominante. De esta forma evitamos centrar la búsqueda de soluciones en espacios que, por puro azar, al empezar la ejecución del algoritmo tengan un buen valor de salud, pero no el mejor que podría tener.

La élite no solo tiene garantizado el paso a la siguiente generación sino que también será sometida a las operaciones de cruce y mutación, en el caso de ser seleccionadas para la reproducción

En el apartado de datos experimentales, en el entorno 3, veremos como esta medida resulta ser efectiva, ante la existencia de múltiples localizaciones equivalentes o similares, la población se distribuirá a lo largo del espacio de soluciones concentrada alrededor de estas zonas. Si bien se podría entender que el algoritmo no esta ofreciendo una respuesta correcta a nuestra pregunta de ¿donde estamos? También se puede interpretar que esta ofreciendo una información precisa de donde podemos estar, con cierto grado de incertidumbre entre varias ubicaciones posibles. La resolución de esta incertidumbre se realizaría mediante el movimiento del robot.

4.5. Modelo de observación

El modelo de observación se encarga de generar, a partir de un individuo dado $[x, y, \theta]$ de la población y el plano del entorno, el vector de observación que obtendría el robot si estuviera situado en dichas coordenadas $[x, y, \theta]$ y su escáner realizara una barrido, de forma que obtenemos una lectura estimada de los sensores, Se

$$Se = (Se_1..Se_{ns}); \quad ns = 1..n \text{ medidas} \quad (4.5)$$

El entorno posee capacidad de variación intrínseca: hay puertas que pueden estar abiertas o cerradas, por lo cual a la hora de realizar la estimación de lectura de los sensores desde una determinada posición hay que considerar ambas opciones, la función de salud será la encargada de decidir cual de las dos lecturas es mas adecuada para estimar la solución.

4.6. Función de salud

En el problema de la localización global existen ciertos aspectos que hacen que la salud de los individuos sea difícil de evaluar:

- El rango, la precisión y el número de sensores limitan la posibilidad de discriminación entre diferentes posiciones, conduciendo a la función de salud a ofrecer un gran número de falsos mínimos locales
- Las multiplicidades geométricas en el entorno también generan múltiples mínimos locales que implican incertidumbre en la selección de la posición real. Figuras 4.3,4.4



Figura 4.3: Area del pasillo, donde se observa la multiplicidad espacial de la misma forma, con pequeñas diferencias entre ellas. Esto da lugar a la existencia de mínimos locales a la hora de evaluar la función de salud

La función de error elegida es la suma de los errores cuadráticos medios, entre la predicción realizada por el modelo de observación, S_e , y las medidas

reales adquiridas a través del sensor.

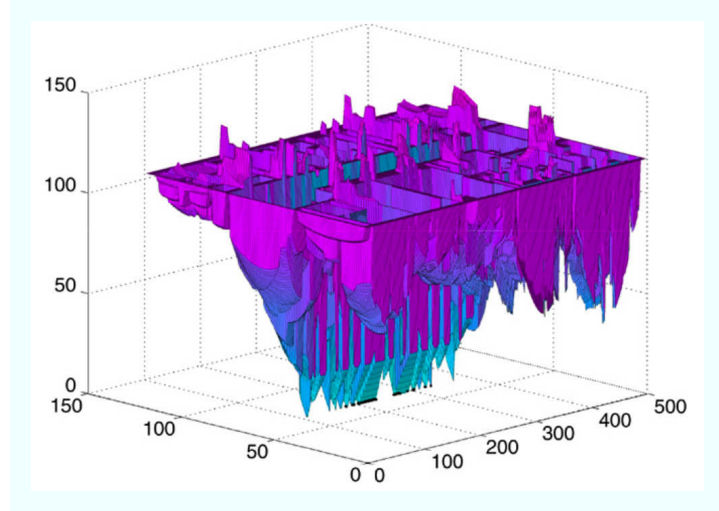


Figura 4.4: Función de salud evaluando una posición en la zona del pasillo, sin consideración del la componente de ángulo. Se observan múltiples mínimos que derivan ambigüedades en la localización

Se utilizan los mapas con las puertas abiertas y con las puertas cerradas en la evaluación de Se , para intentar adaptarnos a la variabilidad del medio obtenidas por el robot, Sm , comparadas punto a punto.

$$Salud = F, f(Sm, Se) \quad (4.6)$$

Donde Sm es la Lectura real del sensor y Se es la lectura estimada del sensor en la posición del individuo I a evaluar.

El la figura 4.5 muestra la eficiencia de la función de error respecto a las variaciones de ángulo en una ubicación determinada. La diferencia de valores obtenidos entre la orientación correcta y las erróneas es de varios ordenes de magnitud, teniendo la curva de respuesta una pendiente muy elevada Este hecho muestra la fiabilidad de nuestra función, en la que podemos confiar a la hora

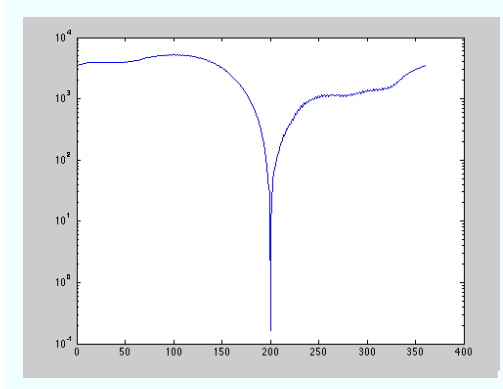


Figura 4.5: Evaluación de la función de salud en función de la orientación, se puede observar como en la orientación coincidente, en este caso 200° , la función nos ofrece un valor situado varios ordenes de magnitud por debajo del resto de las orientaciones

de determinar si nuestro modelo de observación encaja con el plano de obstáculos medido por los sensores, pero a la vez nos presenta el reto de ser extremadamente selectiva en determinadas áreas, lo cual puede dificultar la convergencia.

4.7. Operador de Selección

Utilizamos un método de selección preservativo, todos los individuos tienen posibilidades de ser elegidos para la recombinación, uno de los padres se obtiene de la selección de padres previamente realizada, el otro de la población entera. Una vez determinado el número de padres que queremos utilizar, evaluamos la función de salud del total de la población, que nos proporciona información acerca de su proximidad a la solución, los individuos con mejores resultados serán elegidos para ser los progenitores de la siguiente generación.

La selección del conjunto de padres la efectuamos por el método de la ruleta, a los padres con mejor salud se les asigna más probabilidad de ser elegidos que a los peores, dependiendo de su diferencial de su función de salud respecto al anterior. La selección del segundo de los padres se realiza con probabilidad

uniforme en toda la población, debiendo ser diferentes ambos padres.

En cada una de las reproducciones se generaran dos hijos, que deben ser viables, es decir, deben tener expresión dentro del espacio de soluciones.

4.8. Operador de Cruce

En cada operación de cruce se genera un hijo, y se realizan dos cruces por cada pareja de padres seleccionados. Disponemos de tan solo tres genes x, y, θ , para cruzar, y cada hijo va a heredar el gen de uno de los padres con una probabilidad P_c

Hay dos posibilidades, que cada hijo sea complementario del otro, o la generación independiente de cada uno de ellos. Esta segunda opción es la que hemos escogido, ya que en la naturaleza, que es nuestra inspiración, la primera opción no existe.

Cada gen de la descendencia es escogido mediante una mascara de cruce, que se genera para cada hijo de forma independiente. Si existe un 1 en la mascara de cruce se escoge el gen del padre A, si hay un 0 del padre B. Figura 4.6

En general, si tenemos un Padre A, P_a , escogido de entre los padres en función de su probabilidad asignada, y un Padre B, P_b , elegido uniformemente entre toda la población, la formación de los hijos será:

$$H_i = \begin{cases} P_{a_i} & P_c < m_i \\ P_{b_i} & \text{otro caso} \end{cases} \quad i = [1., 3] \quad (4.7)$$

Siendo m_i un número aleatorio escogido uniformemente en el intervalo $[0, 1]$

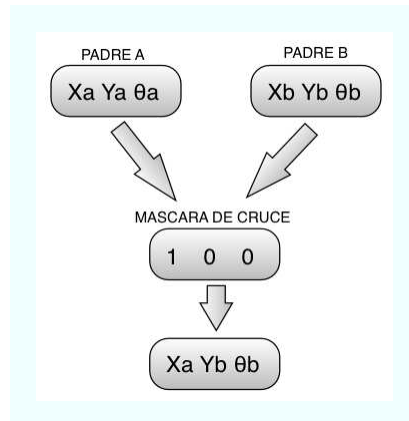


Figura 4.6: Cruce de los padres mediante mascara de cruce. Un 1 determina que el gen correspondiente será seleccionado del Padre A, un 0 del Padre B

4.9. Operador de mutación

El operador de mutación tiene papel fundamental a la hora de la exploración del espacio de soluciones, introduce variaciones aleatorias, con una probabilidad P_m , en cada uno de los hijos que se generan.

Si no introdujéramos la mutación la población inicial marcaría unívocamente todas las soluciones posibles a las que podemos llegar mediante el operador de cruce, las soluciones que se pueden explorar a lo largo de las iteraciones. Esto es así debido a nuestra forma de codificar la solución en tres genes. Si la codificación hubiera sido binaria, utilizando unos y ceros, y lo suficientemente extensa, el operador de cruce podría hacernos explorar todo el espacio sin necesidad de la mutación. En la figura 4.7 podemos verlo que ocurre con la operación de cruce en un espacio de dos dimensiones (Xiaofeng Qi, 1994) , nosotros tenemos 3, pero las conclusiones son extensibles para cualquier espacio n -dimensional. Dos padres, a y b , combinan sus genes, de forma que generan dos hijos, dependiendo del azar, estos hijos pueden ser c ó d , y también a ó b . Nuestra exploración del espacio esta realmente limitada, se torna evidente la necesidad de incluir un

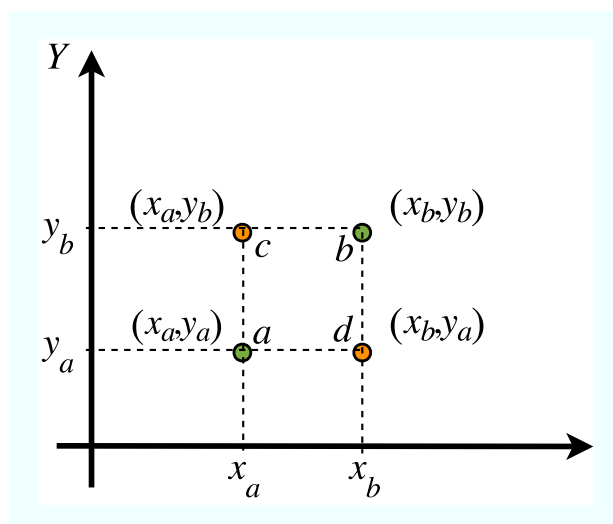


Figura 4.7: Operación de cruce de dos padres en un espacio de dos dimensiones. Los padres, a y b , pueden generar como hijos a y d , pero también, dependiendo de las probabilidades de cruce, a si mismos. La exploración del espacio de soluciones está muy limitada, es necesaria la introducción de la mutación

operador de mutación.

La mutación es aleatoria, pero en nuestro caso la magnitud de esta aleatoriedad está ponderada por una gaussiana, cuya sigma depende de lo cerca que nos encontremos de la solución, es decir, del valor que adopte la función de salud de nuestros individuos mejor adaptados.

$$Hm_i \begin{cases} H_i + Nn(fitness) & Pm < mi \\ H_i & \text{otro caso} \end{cases} \quad i = [1..ngenes]; \quad (4.8)$$

La idea subyacente es que, al principio, la mutación permite explorar todo el espacio de soluciones, y a medida que la función de salud nos indica que nos aproximamos a un mínimo local, hacemos oscilar la solución en un entorno local mas reducido en busca de un mejor ajuste.

Al comienzo de las iteraciones ajustamos la desviación típica de la gaussiana de forma que cubra el área en función de la densidad de población. Si a cada individuo le corresponden n metros cuadrados, este área corresponde a la σ de nuestra campana. A medida que la salud va acercándose a su mínimo el valor

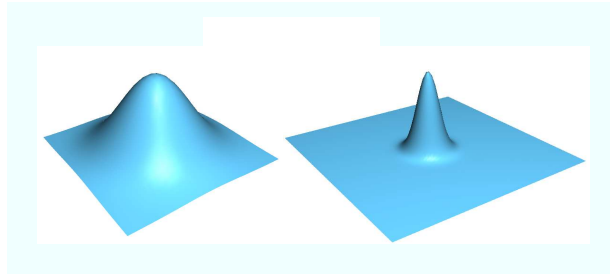


Figura 4.8: La amplitud de la mutación está ponderada por una gaussiana cuya sigma depende de la función de salud, a medida que el algoritmo converge las mutaciones tienen mas probabilidad de situarse en el entorno local del individuo, el efecto es el de hacer vibrar a la población en su entorno para conseguir un ajuste fino de la solución

de sigma disminuye, de forma que sea mas probable explorar cerca de nuestra solución. Figura 4.8

En la figura 4.9 podemos observar un estado intermedio de como la población va convergiendo hacia la solución a medida que las iteraciones se van sucediendo. En este caso, la multiplicidad de la zona es baja, por lo que las poblaciones se distribuyen rápidamente alrededor de nuestra zona objetivo, y de la zona de localización equivalente. En la figura 4.10 tenemos una etapa de la con-

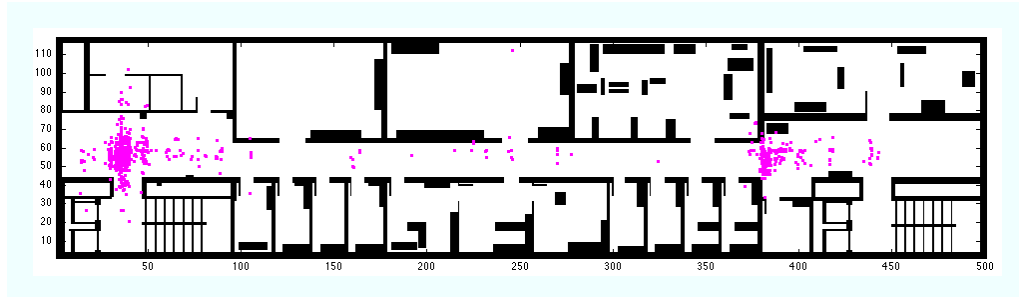


Figura 4.9: Convergencia en una zona donde solo existe una localización equivalente, las poblaciones están distribuidas alrededor del mínimo local (a la derecha) y del mínimo global (izquierda).

vergencia en la zona de los despachos, este entorno presenta múltiples entornos equivalentes, por lo que la población se distribuye de forma proporcional a lo largo de estos mínimos locales. A lo largo de las iteraciones, pequeñas diferencias en las capacidades de adaptación efectúan la atracción de las soluciones, gracias a la reproducción, hacia el mínimo-mínimos globales.

En ambos casos se aprecia el efecto de la distribución en forma de cruz de la población debido al operador de cruce.

4.10. El Algoritmo

Paso 1: inicialización

Generamos una población inicial arbitrariamente distribuida a lo largo de todo el espacio de soluciones, en ausencia de cualquier tipo de información,

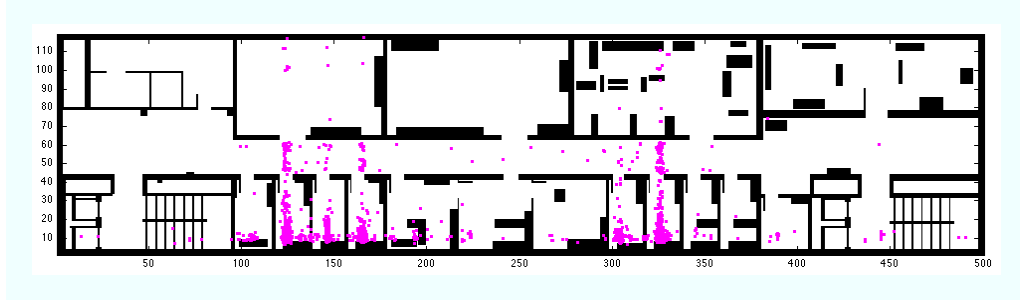


Figura 4.10: Poblaciones distribuidas a lo largo del espacio de soluciones concentradas en mínimos locales equivalentes, en este caso las zonas de los despachos.

cualquier posición es candidata como solución

$$P = (I_1..I_N); N : \text{número de individuos en la población} \quad (4.9)$$

$$I_i = [I_{i,x}, I_{i,y}, I_{i,\theta}]; i = 1..N \quad (4.10)$$

Evaluamos la salud de cada individuo utilizando la observación real y el modelo de observación calculado.

$$F_i^k, f(Sm, Se(I_i^k)) \quad (4.11)$$

Paso 2: Iteraciones

En cada iteración varia nuestra población

$$I_i^k = [I_{i,x}^k, I_{i,y}^k, I_{i,\theta}^k] \quad k = k - \text{esima iteración} \quad (4.12)$$

Seleccionamos la élite y los padres de los elementos con mejor salud

$$\text{Élite} = E^k = [e_1^k .. e_{te}^k]; te = \text{número de individuos de la élite} \quad (4.13)$$

$$\text{Padres}, G^k = [p_1^k .. p_{np}^k]; np : \text{número de padres} \quad (4.14)$$

Seleccionamos dos miembros para la reproducción, uno de los padres, Pa , con probabilidad

$$P_{selección\ Pa} = \frac{F(p_i)}{\sum_{tp} F(p_i)} \quad (4.15)$$

otro de la población entera, Pb , con probabilidad

$$P_{selección\ Pb} = \frac{1}{N} \quad (4.16)$$

y generamos dos hijos mediante el cruce

$$H_{j,i} = \begin{cases} Pa_i & Pc < m_i \\ Pb_i & \text{otro caso} \end{cases} \quad i = [1.,3], j = [1,2] \quad (4.17)$$

y aplicamos la mutación

$$Hm_{j,i} = \begin{cases} H_i + Nn(fitness) & Pm < mi \\ H_i & \text{otro caso} \end{cases} \quad i = [1.,3], j = [1,2] \quad (4.18)$$

Repetimos la reproducción, seleccionando padres diferentes en cada ocasión, hasta que

$$hijos\ generados = \text{número de individuos en la población} - \text{número de individuos en élite} \quad (4.19)$$

Generamos una nueva población para una nueva iteración

$$\text{nueva población} = \text{élite} + \text{hijos generados} \quad (4.20)$$

Evaluar nuestra figura de merito, en nuestro caso es una evaluación de la función de salud de la solución.

$$F_i^k, f(Sm, Se(I_i^k)) \quad (4.21)$$

Si la figura de merito no esta satisfecha, o el número de iteraciones máximo no se ha alcanzado, ejecutamos una nueva iteración k

Paso 3: Actualización

El individuo con mejor valor de salud es elegido como solución.

Capítulo 5

Datos Experimentales

Es necesario someter el algoritmo desarrollado a pruebas experimentales para determinar su precisión y robustez en la localización en diferentes escenarios. No se ha hecho hincapié en la evaluación de la velocidad de ejecución del algoritmo debido a que la exploración del espacio de soluciones mediante los algoritmos genéticos es intrínsecamente paralela, por lo que el incremento de la velocidad de ejecución debería conseguirse no solo por la optimización del código sino también por la paralelización del mismo, consiguiendo de esta forma un rendimiento no alcanzable con nuestro actual paradigma de programación. La implementación y ejecución paralela posibilitara conseguir tiempos de ejecución imposibles con nuestro sistema. Un dato que si ofrece una orientación de la eficiencia de ejecución del algoritmo es el número de soluciones ensayadas antes de la convergencia, dato que se muestra en cada uno de los escenarios de prueba.

Se consideraremos que una localización ha sido exitosa cuando está situada en un entorno de ± 1 celda respecto a la posición real. Esto se traduce en un error del $\pm 0,1\%$ en el eje X y del $\pm 0,4\%$ en el eje Y . Asimismo toleramos $\pm 3^\circ$ grado en ángulo, que representa en error de $\pm 0,4\%$ en la resolución angular de la solución.

Como se ha visto en el Capítulo 2, Los algoritmos Genéticos, existen una

serie de parámetros, el tamaño de la población, el tamaño de la elite, la probabilidad mutación y la probabilidad de combinación, cuya elección condiciona el funcionamiento del algoritmo.

Antes de evaluar la eficiencia se ha de determinar cuales son estos parámetros óptimos de funcionamiento, para ello posicionamos el robot en la posición $x=40, y=60, \theta=0$, situado en el vestíbulo. Esta ubicación se ha elegido ya que se caracteriza por tener una sola multiplicidad espacial y está en un entorno alejado de discontinuidades. Esto la convierte en un entorno amable para el algoritmo.

Figura 5.1

Una vez seleccionados los parámetros óptimos someteremos su funcionamiento a prueba en tres entornos con diferentes características que nos permitan evaluar su funcionamiento en situaciones típicas:

- Entorno 1: $x=40, y=60, \theta=0$ El vestíbulo, que es el mismo utilizado para el ajuste de los parámetros, presenta un mínimo amplio y alejado de discontinuidades, una localización, a priori, sencilla. Figura 5.1
- Entono 2: $x=305, y=10, \theta=90$ Los despachos, se caracteriza principalmente por la existencia de múltiples posiciones equivalentes, nos ayudara a determinar la capacidad de discriminación entre mínimos locales. Figura 5.13
- Entorno 3: $x=35, y=20, \theta=90$ Rellano de la escalera izquierda, que presenta un mínimo local muy localizado, evaluaremos la capacidad del algoritmo de resolver situaciones complejas, Figura 5.20

Con el fin de tener una muestra significativa de su comportamiento sometemos al algoritmo a 100 ejecuciones en cada uno de los entornos y se analizan los resultados obtenidos.

En la visualización de los resultados se utilizan diferentes modelos de grafica:



Figura 5.1: Entorno de localización con la lectura de los sensores. El robot, señalado como el signo + rojo, está situado en las coordenadas $x=40$, $y=60$, $\theta=0$

- *Grafica de porcentaje de acierto frente a incremento de población:* Permiten observar como evoluciona la convergencia del algoritmo respecto al número de individuos involucrados
- *Gráfica de iteraciones medias para conseguir la convergencia frente a población:* nos muestra el número medio de iteraciones necesarias para converger a la solución
- *Gráfica de soluciones medias ensayadas para conseguir la convergencia frente a incremento de población:* Nos proporciona información acerca de la velocidad de ejecución del algoritmo.
- *Mapa de distribución de población:* muestran, superpuestas al entorno de exploración, la distribución espacial de los individuos que forman las poblaciones, excluyendo la representación de orientación por claridad.
- *Histograma de soluciones:* muestran como se distribuyen espacialmente las soluciones a lo largo de las 100 ejecuciones del algoritmo que realizamos en cada ensayo. Solo se representan las soluciones de las coordenadas x e y en las que se localiza el robot Figura 5.2

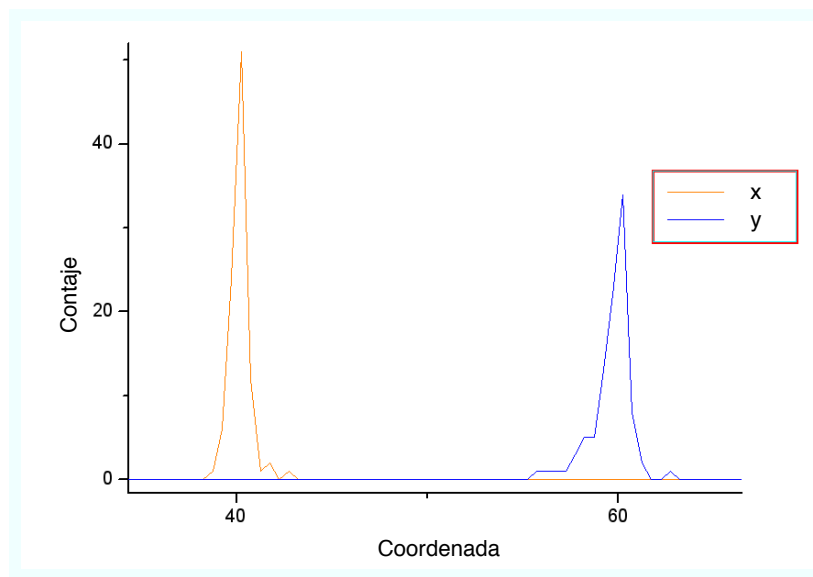


Figura 5.2: Histograma de soluciones. Muestra como se han distribuido las soluciones finales en el espacio a lo largo de las 100 iteraciones a las que se somete el algoritmo en cada test. En naranja se dibujan las soluciones en el eje X, en azul las del eje Y. En este histograma las soluciones están centradas en torno a $x=40$, $y=60$

5.1. Elección de los parámetros óptimos de funcionamiento del algoritmo

Antes de empezar a evaluar el algoritmo hemos de determinar la configuración óptima de los parámetros que lo controlan, para tener una máxima confianza en el resultado obtenido, tanto en precisión como en robustez, para ello se ha elegido el entorno del hall. Figura 5.1. Este área presenta una cierta bondad al tener en el espacio de soluciones un mínimo alejado de discontinuidades.

Los parámetros se van a evaluar para el ajuste son la Probabilidad de Cruce P_c , Probabilidad de Mutación P_m , el tamaño de población será evaluado indirectamente y el tamaño de la población y de la élite se preestablecen.

Debido a la multiplicidad espacial de entornos similares, que se traduce en multiplicidad de mínimos locales, existe el peligro de una convergencia prematura hacia uno de estos mínimos. Para evitarlo se ha elegido una élite compuesta por el 2.5 % de los individuos mejor adaptados de cada generación, los padres serán el 25 % de los mejores individuos, esto puede tener el efecto no deseado de mantener a la población dispersa innecesariamente durante las primeras iteraciones, pero evita la convergencia temprana.

La detención del algoritmo se fija en 15 iteraciones sin consideración de la adaptación de la población. En las pruebas de evaluación en los diferentes entornos la condición de detención también tendrá en cuenta la adaptación. Una vez transcurridas las 15 iteraciones elegimos al individuo mejor adaptado como la solución.

El entorno de pruebas tiene un área de 877 m^2 , un dato importante número de individuos por metro cuadrado que necesarios para alcanzar la convergencia, con una alta densidad de población el algoritmo casi se convierte en uno de búsqueda exhaustiva. En la Tabla 5.1 vemos las densidades de población para las poblaciones utilizadas en las pruebas.

Tamaño de población	Individuos/ m^2
100	0,11
200	0,23
300	0,34
400	0,46
500	0,57
1000	1,14
2000	2,28

Tabla 5.1: *Densidad de Individuos para las poblaciones utilizadas*

5.1.1. Dependencia con la Probabilidad de Mutación

Para el estudio de la convergencia en función de la variación de la Probabilidad de Mutación, P_m , se establece una Probabilidad de Cruce $P_c = 0.8$, y variamos la Probabilidad de Mutación de 0.1 a 0.5, Figura 5.3.

Valores superiores a 0.5 harían que el algoritmo prácticamente se comportara como un sistema de búsqueda puramente estocástica. En la figura 5.3 se observa como el incremento de la probabilidad de mutación hace que mejore la eficiencia del algoritmo. Al incrementar la probabilidad de mutación conseguimos incrementar el comportamiento estocástico del sistema, la mejora de eficiencia es debido a las características de nuestro entorno: el espacio de soluciones no es continuo. Las discontinuidades surgen de la existencia de disrupciones físicas en el entorno, como son las paredes y demás obstáculos naturales, que hacen que la exploración continua tenga peores resultados: las soluciones tienen que poder "saltar" de unas zonas del espacio a otras separadas por discontinuidades, y este proceso se potencia con la mutación.

Este resultado es esperado ya que como hemos visto la mutación juega un papel fundamental en la exploración del espacio de soluciones, especialmente cuando el número de genes en juego es bajo, en nuestro caso solo disponemos de 3 genes en la codificación de la solución.

También se puede apreciar que la evolución de la eficiencia con el número de

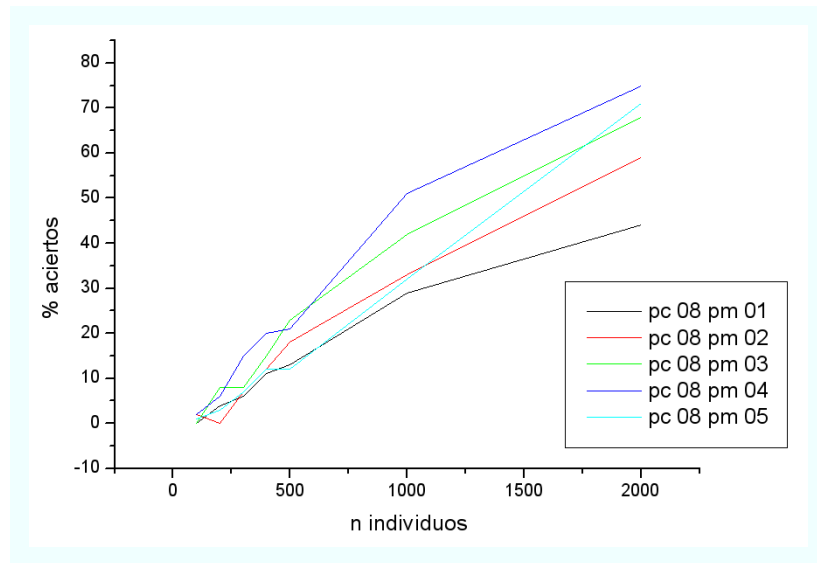


Figura 5.3: Evolución de los aciertos, con P_c constante, frente a variación de población y de P_m . Se puede apreciar como incrementar la probabilidad de mutación mejora la eficiencia del algoritmo

individuos no es lineal, existe una tendencia a una zona de saturación, a partir de la cual el incremento del número de individuos no proporciona mejores resultados, pero si mayor consumo de recursos. En el entorno 1 de experimentación se estudiara con mas detalle este aspecto.

5.1.2. Dependencia con la Probabilidad de Cruce

A partir de los resultados derivados de la evaluación del comportamiento con la variación de la Probabilidad de Mutación, estimamos que el comportamiento óptimo se consigue con $P_m=0.4$.

Para estudiar como afecta la variabilidad en la Probabilidad de Cruce se establece la Probabilidad de Mutación en dicho valor y se varia la Probabilidad de Cruce de 0.5 a 0.9, Figura 5.4

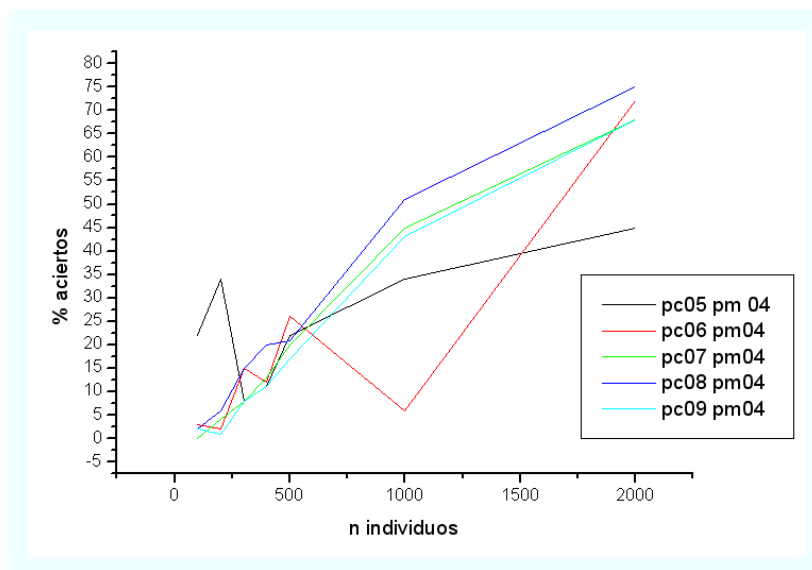


Figura 5.4: Evolución de la tasa de aciertos frente a P_m constante ($P_m=0.4$)

Para poblaciones inferiores a los 500 individuos y Probabilidad de Cruce inferior a 0.6 el incremento de la tasa de convergencia no está directamente relacionada con el número de individuos, esto es debido a que el número de soluciones ensayadas no es suficiente para la correcta exploración del espacio, no tiene tiempo de converger hacia las zonas adecuadas, y se depende mas del azar que de la evolución.

Con Probabilidades de Cruce mayores a 0.6 el mecanismo de evolución funciona satisfactoriamente aun con poblaciones pequeñas, observándose un incremento en la tasa de aciertos proporcional al incremento de individuos involucrados en la búsqueda de solución, así como un incremento de la eficacia general a medida que la Probabilidad de cruce. Con $P_c=0.9$ disminuye nuevamente la eficacia global.

El comportamiento mas efectivo se consigue con una Probabilidad de Cruce de 0.8.

5.2. Análisis del comportamiento con los valores óptimos $P_m=0.4$, $P_c=0.8$

Hemos determinado que los valores óptimos de funcionamiento son $P_c=0.8$ y $P_m=0.4$, con ellos se alcanzan las tasas de acierto mostradas en la tabla 5.2

Tamaño de población	%exito	Indiv/ m^2
100	2	0,11
200	6	0,23
300	15	0,34
400	20	0,46
500	21	0,57
1000	51	1,14
2000	75	2,28

Tabla 5.2: Porcentajes de acierto bajo los parámetros óptimos de ajuste, $P_c=0.8$ $P_m=0.4$

Estas tasas de acierto están determinadas por las restricciones de convergencia que hemos establecido, pero se observa que la totalidad de las soluciones obtenidas por el algoritmo a lo largo de las 100 ejecuciones, con una población de 2000 individuos, se distribuyen en torno de la posición objetivo. Figura 5.5

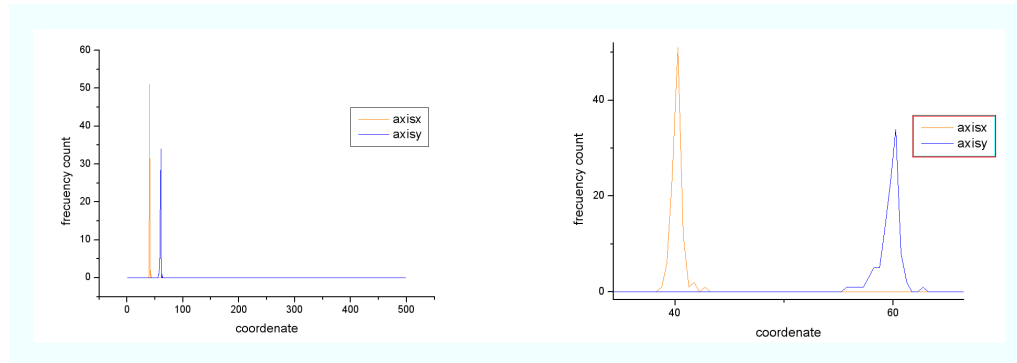


Figura 5.5: Distribución de las soluciones con una población de 2000 individuos a lo largo de 100 ejecuciones, a lo largo de todo el espacio de soluciones y detalle en la coordenada de localización es $x=40, y=60, \theta=0^\circ$, con un nivel de ruido del 2 %. La totalidad de soluciones se centra en torno a las coordenadas objetivo, aunque no todas cumplen los criterios de convergencia

5.2.1. Robustez frente al ruido

Una vez evaluada la dependencia con la probabilidad de mutación y la probabilidad de cruce, y determinados los parámetros óptimos de funcionamiento evaluamos la robustez del algoritmo ante la presencia de ruido en la lectura del sensor de distancias.

Se introduce ruido aleatorio en la estimación del mapa 2D generado por el sensor, con amplitudes de 2, 5, 8, 10, 15, 20, 30, 40, 50, 60, 70, 80 y 90 %, y se introducen en el algoritmo, con los parámetros óptimos y los mismos criterios de la convergencia, el tamaño de población en este caso es de 2000 individuos por ser la situación en la que mejores resultados se obtienen.

En la figura 5.5 se muestra la distribución de las soluciones ofrecidas por el

algoritmo a lo largo de 100 iteraciones, con un 2 % de error, se aprecia como la totalidad de las soluciones ofrecidas se encuentran en el entorno de la solución.

El algoritmo presenta un alto nivel de tolerancia al ruido, con niveles de ruido de hasta un 35 % en la lectura de los sensores, el índice de aciertos se mantiene por encima del 50 % Figura 5.6

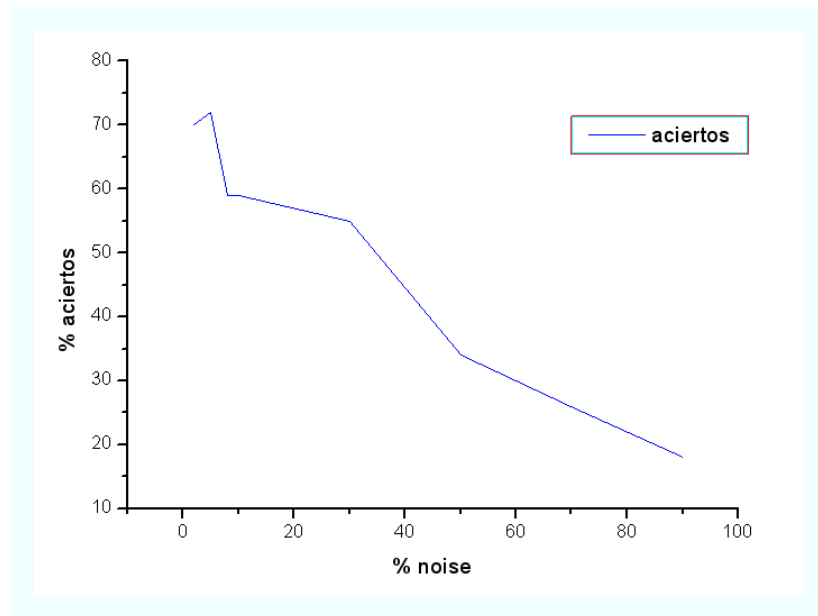


Figura 5.6: Evolución de la tasa de aciertos frente al ruido en los sensores. La tasa de aciertos se mantiene en niveles por encima del 50 % con ruido de hasta el 40 %

Si bien la tasa de aciertos disminuye significativamente a partir de niveles de ruido del 35 %, observamos en el histograma de soluciones ofrecidas a lo largo de las 100 iteraciones con un 90 % de ruido como la totalidad de soluciones están distribuidas en torno a la posición objetivo (o una posición equivalente en x , alrededor de la coordenada 400), aunque solo el 18 % satisface los criterios de convergencia, Figura 5.7

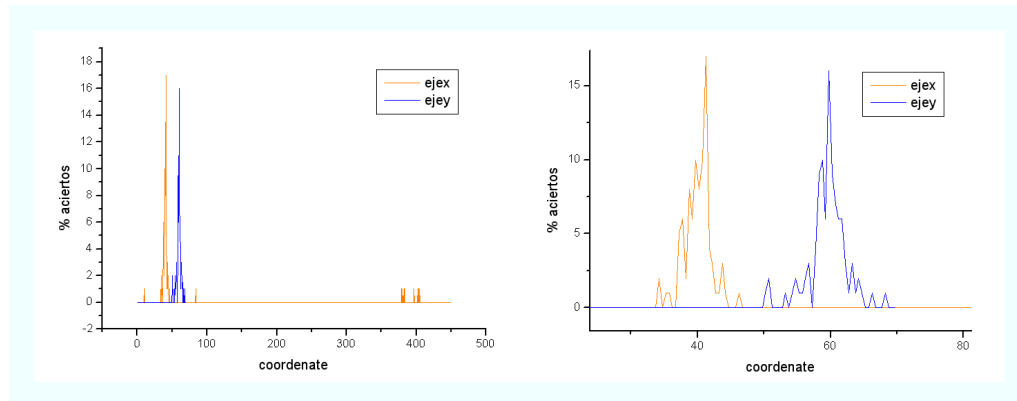


Figura 5.7: Histograma espacial de las soluciones ofrecidas por el algoritmo y detalle para la localización en las coordenadas $x=40, y=60, \theta=0^\circ$, con $P_c=0.8$, $P_m=0.4$, población de 2000 individuos y un error introducido en los sensores de un 90 %. La totalidad de las soluciones se concentran en torno a las coordenadas objetivo, o en entorno equivalente, pero solo el 18 % cumple los criterios de convergencia.

5.3. Evaluación de la capacidad de convergencia en diversos entornos

Una vez determinados los parámetros óptimos de funcionamiento del algoritmo examinamos su capacidad de convergencia en los tres entornos de prueba seleccionados: el vestíbulo ($x=40, y=60, \theta=0$, entorno amable), despachos ($x=305, y=10, \theta=90$, entorno con multiplicidad espacial) y rellano de escalera ($x=35, y=20, \theta=90$, entorno con mínimo global muy localizado).

En el test se utilizan poblaciones de 100, 200, 300, 400, 500, 1000, 2000, 3000, 4000 y 10000 individuos, para cada una de las poblaciones examinaremos las tasas de convergencia y el número de iteraciones necesario, a partir de estos datos estimaremos el número de soluciones ensayadas en cada caso

Para la evaluación de la capacidad de convergencia se ha modificado la condición de detención del algoritmo respecto a la utilizada en el ajuste de parámetros, ahora el número máximo de iteraciones permitido es 50 y se añade la detención por la salud de la población, establecida en un valor de la función de salud que garantiza que la convergencia se ha alcanzado. La condición de detención por número de iteraciones es necesaria para asegurarnos la detención de la ejecución, en el caso de que la población convergiera hacia una zona alejada del mínimo global es probable que entrara en un bucle infinito.

Al igual que en el ajuste el algoritmo es ejecutado 100 veces en cada uno de los entornos para tener una muestra fiable de su comportamiento.

5.3.1. Entorno 1: el hall, $x=40$ $y=60$ $\theta=0$

El hall de la planta, situado en la zona izquierda del mapa, es el mismo entorno utilizado en el proceso de ajuste de los parámetros. Representa un entorno sin discontinuidades cercanas y con sólo un mínimo local alternativo, la posición y lectura del sensor en este área se muestra en la figura 5.8



Figura 5.8: Localización en las coordenadas $x:40$, $y=60$, $\theta=0$. Este entorno es de fácil localización, esta en una zona amplia, alejado de discontinuidades y no posee multiplicidad espacial

En la Figura 5.9 se muestran los histogramas de localizaciones realizadas con poblaciones de 100, 500, 1000, 5000 y 10000 individuos. El eje de Abcisas

representa la coordenada espacial donde se ha realizado la localización, el eje de ordenadas, el número de localizaciones realizadas.

El comportamiento en este área es excepcionalmente bueno, debido a sus condiciones de contorno y a que los parámetros están optimizados para esta ubicación. Con 100 individuos el grueso del conjunto de las localizaciones se agrupa en torno de la localización objetivo, habiendo un pequeño porcentaje que se localiza en la posición equivalente, en el entorno de la coordenada 400 (naranja). Esto ocurre para el eje X, en y la localización es estable en torno al la coordenada objetivo (60)

A partir de poblaciones de 1000 individuos el entorno equivalente es abandonado como solución y el 100 % de las soluciones ofrecidas se concentra en la solución objetivo, observándose una estabilización de la tasas de convergencia en torno al 95 % a partir de los 2000 individuos. Figura 5.10

El número de iteraciones necesarias para conseguir la convergencia disminuye notablemente con el incremento de la población, esto es debido a que se explora el espacio de forma mas exhaustiva, pero a partir de los 2000 individuos la tasa de aciertos permanece acotada en torno al 95 %, grandes poblaciones solo contribuyen a un incremento soluciones ensayadas, que se traduce en un incremento de consumo de recursos. Esta evolución se muestra en la Tabla 5.3 y la Figura 5.11

Debemos tener en cuenta que aunque el número de iteraciones desciende a medida que incrementamos el tamaño de la población, llegando a tan solo 9 con poblaciones de 10000 individuos, el número de soluciones ensayadas para conseguir la convergencia se incrementa innecesariamente puesto que no se mejora la tasas de aciertos, figura 5.11.

La evolución de una población de 2000 individuos a lo largo de las sucesivas iteraciones se muestra en la figura 5.12, en la que están representada la localización espacial y no la orientación. La población inicial está uniformemente distribuida y vemos como en la iteración 4 prácticamente todos los individuos

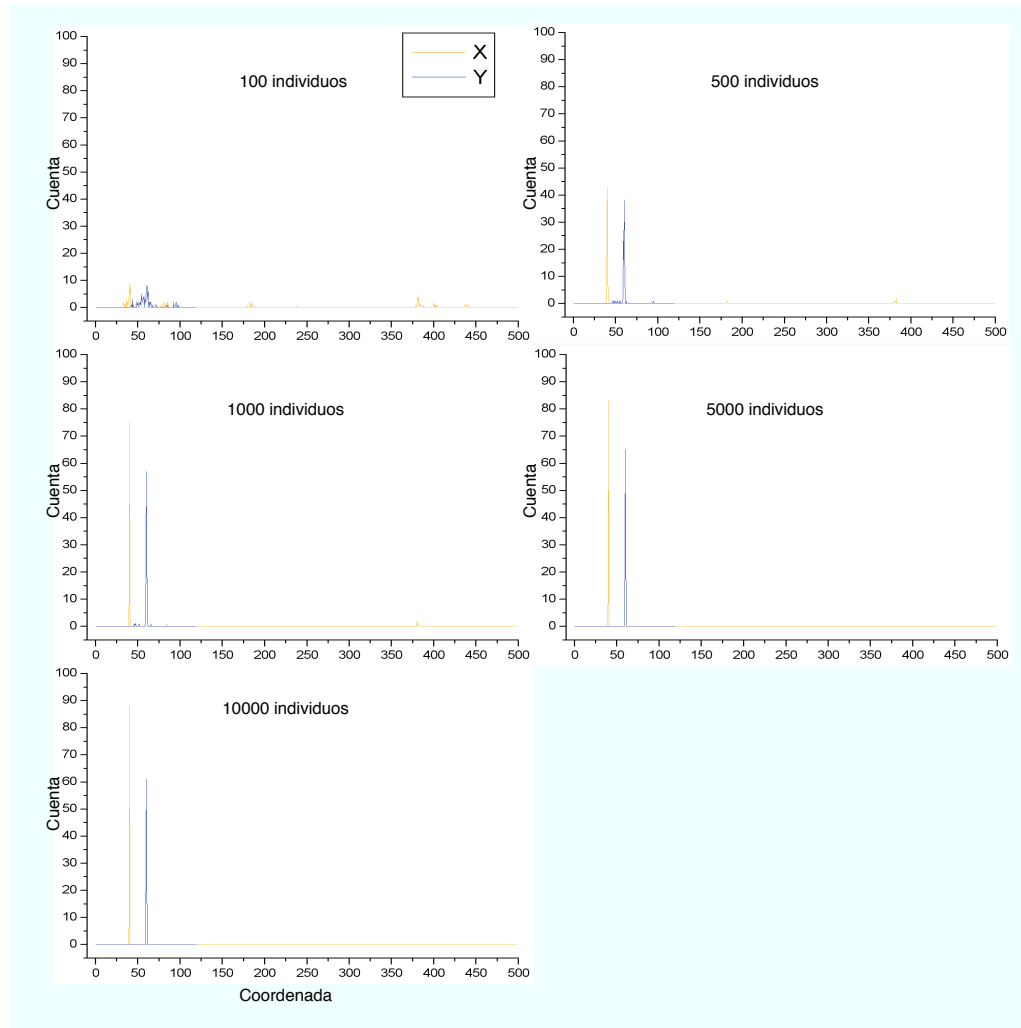


Figura 5.9: Distribución de los resultados a lo largo de 100 iteraciones con diferentes tamaños de población con la ubicación objetivo $x: 40, y=60, \theta=0$. A partir de 1000 individuos la localización en entornos equivalentes, que se observa en los test con poblaciones de 100 y 500, alrededor de la coordenada 400 en el eje X (naranja) es abandonada y la totalidad de las soluciones se centra alrededor de la coordenada objetivo, aunque no el 100 % cumple con los criterios de convergencia. Con 10000 individuos se obtiene un 95 % de aciertos

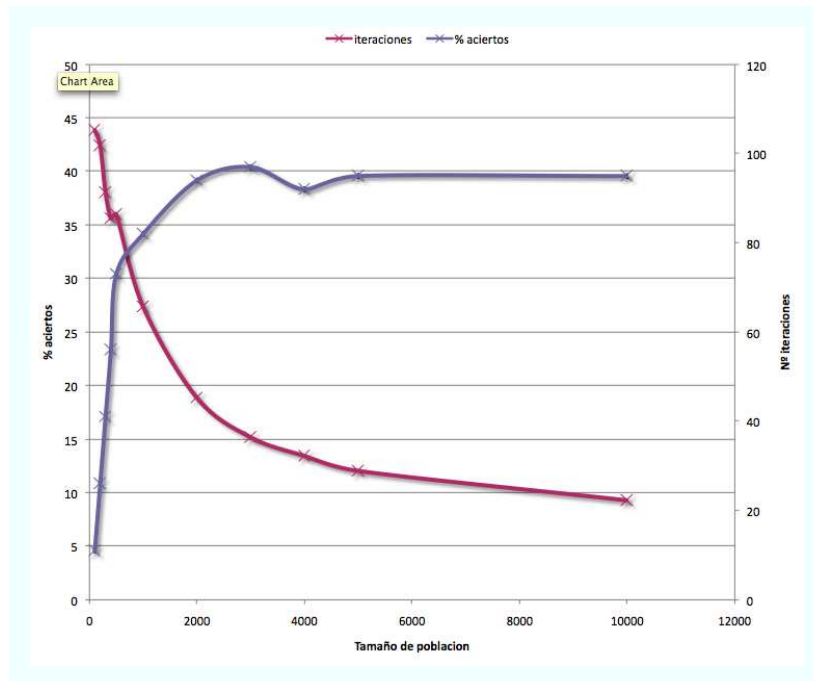


Figura 5.10: Distribución del número de aciertos y los tamaños de población frente a los tamaños de población. A partir de los 2000 individuos el porcentaje de aciertos permanece estable, pero disminuye el número de iteraciones necesarias. El número de posibilidades que se exploran es mucho mayor, y por lo tanto también los recursos consumidos

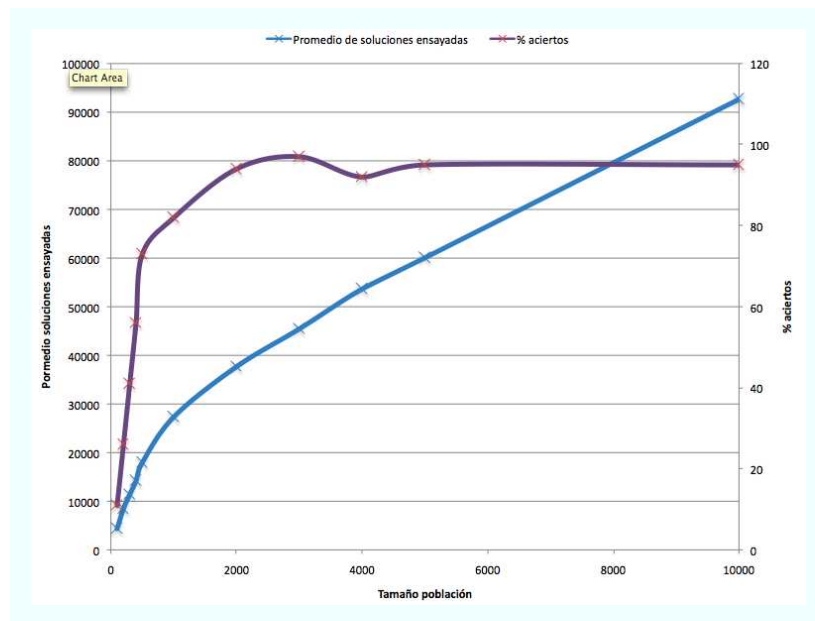


Figura 5.11: Evolución del incremento de soluciones ensayadas y del % de aciertos frente a la variación del tamaño de la población, para la localización $x=40$, $y=60$, $\theta=0$. no resulta eficiente incrementar la población a partir de los 3000 individuos, se disparan las operaciones a realizar sin tener recompensa en el porcentaje de aciertos.

Tamaño poblacion	% aciertos	iteraciones medias	Promedio Soluciones ensayadas
100	11	43,86	4385,71
200	26	42,41	8481,25
300	41	38,02	11406,25
400	56	35,60	14241,10
500	73	35,98	17987,96
1000	82	27,36	27354,84
2000	94	18,85	37700
3000	97	15,16	45480
4000	92	13,42	53680
5000	95	12,02	60100
10000	95	9,27	92700

Tabla 5.3: Variación del porcentaje de aciertos, las iteraciones medias y el promedio de soluciones ensayadas para la localización $x=40$, $y=60$, $\theta=0$.

han abandonado los espacios pequeños, su salud en dichos entornos es muy baja. El mínimo global que posee la solución no es demasiado abrupto, por lo que aunque tengamos individuos en las inmediaciones de la solución el valor de la función de salud aun es bastante elevado, y por lo tanto la magnitud de la mutación introducida tambien lo es, encontrandose los individuos relativamente dispersos. La población en las iteraciones 6 y 8 ya se encuentra concentrada alrededor de la solución y de la localización equivalente que existe. Se aprecian los efectos del cruce y de la mutación. El cruce de poblaciones en un plano de dos dimensiones provoca que se distribuyan en cruz debido al la mezcla de tan solo los dos genes, x e y , con los que podemos operar. El efecto de la mutación se aprecia en la distribución espacial de las muestras. La amplitud de las mutaciones dependen del valor alcanzado por la función de salud, y está ponderado por una gaussiana, de forma que a medida que la salud es mejor la mutación introducida es mas cercana al individuo original

En las iteraciones 10, 13 y 15 el entorno equivalente es prácticamente abandonado y el la mayoría de individuos converge hacia la solución correcta.

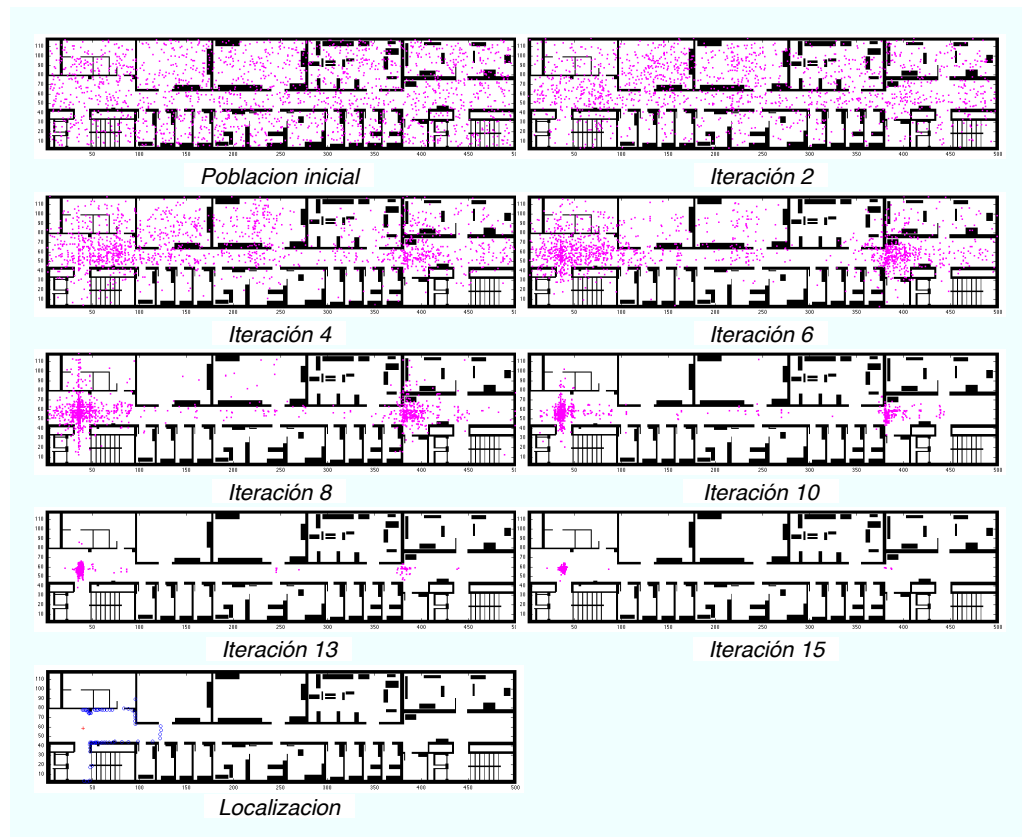


Figura 5.12: Distribución de una población de 2000 individuos durante 15 iteraciones para la localización en $x:40$, $y:60$, $\theta=0$. La población converge rápidamente hacia las coordenadas objetivo y la localización equivalente. Se puede apreciar la convergencia en cruz producto del operador de cruce a partir de la iteración 8.

5.3.2. Entorno 2: los despachos, $x=305$ $y=10$ $\theta=90$

El área de los despachos presenta repetibilidad en el areas de localización. La prueba del algoritmo en esta localización nos proporciona información acerca de su capacidad de discriminación en un espacio multimodal, las características de los despachos son muy similares, con alguna diferencia puntual. Figura 5.13

En la figura 5.15 se muestra un detalle de las areas en las que el algoritmo efectua la mayoría de las localizaciones.

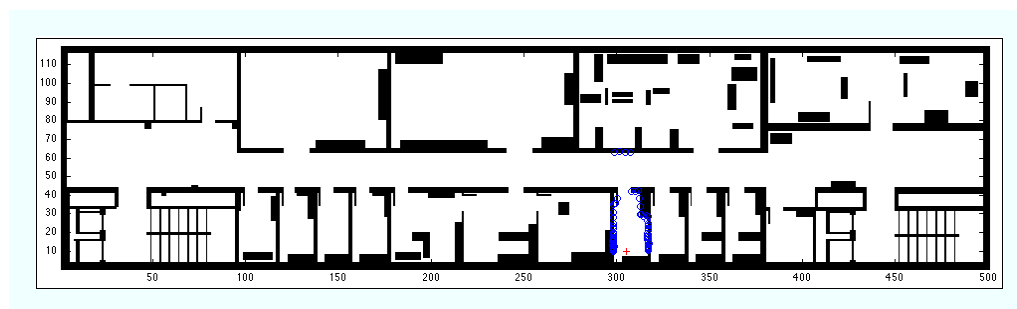


Figura 5.13: Localización en las coordenadas $x:325$, $y=10$, $\theta=90$. presenta múltiples entornos similares a lo largo del mapa

La evolución de las tasas de convergencia es peor a la del entorno 1, al existir diversas localizaciones equivalentes las poblaciones se distribuyen a lo largo de los entornos equivalentes del espacio impidiendo una rápida convergencia hacia la localización objetivo, como se aprecia en la Figura 5.14

En los histogramas de soluciones, figura 5.14, se observa que para poblaciones inferiores a los 1000 individuos las poblaciones se distribuyen en torno a los mínimos locales, siendo la tasas de aciertos del 8 %, tabla 5.4. Las poblaciones se encuentran distribuidas en el espacio principalmente en las tres áreas que presentan características similares, señalados como A, B y C en la Figura 5.15

Esta distribución es esperada, el sistema solo tiene información de un plano bidimensional de lecturas de distancias, y estas distancias son muy similares en

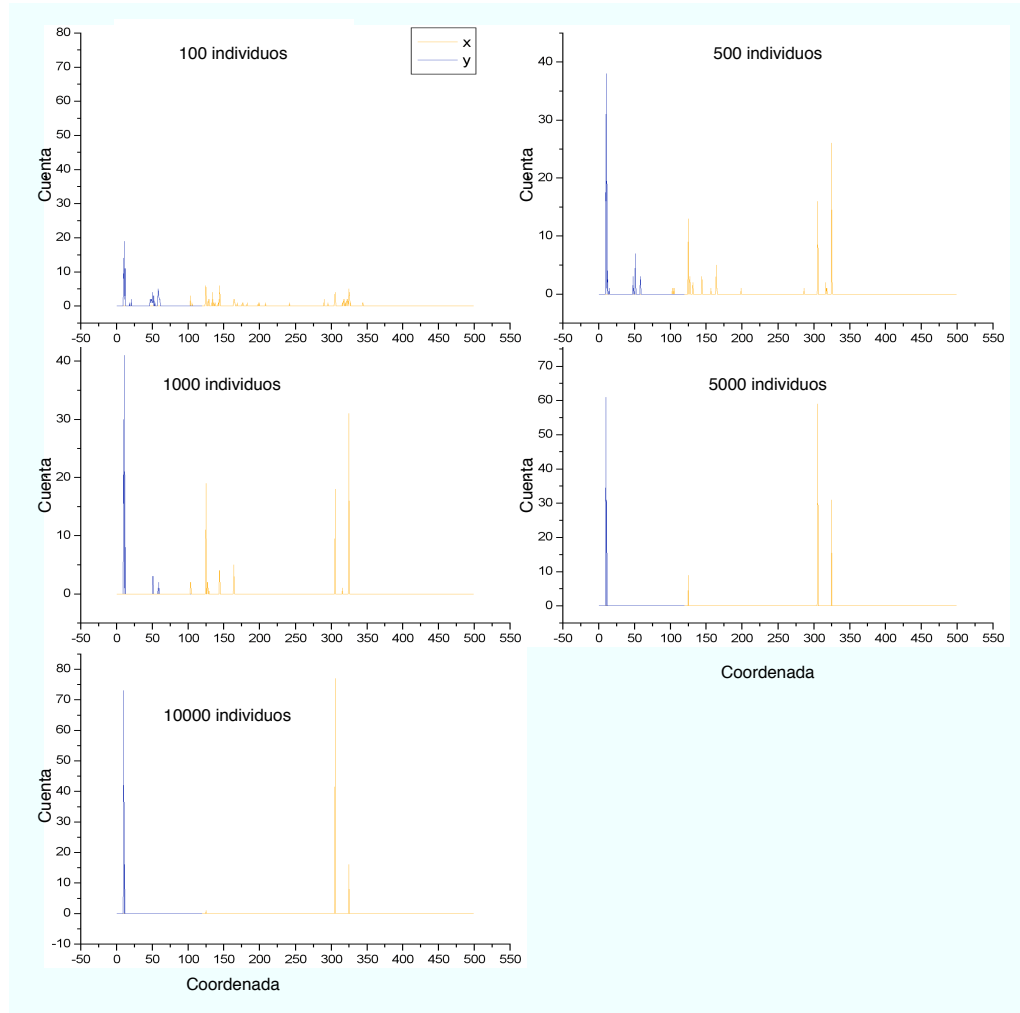


Figura 5.14: Distribución de los resultados a lo largo de las 100 iteraciones con diferentes tamaños de población con la ubicación objetivo $x: 305, y=10, \theta=90$. La existencia de múltiples mínimos locales con valores de salud muy próximos al de la coordenada objetivo hace que las localizaciones en el eje x permanezcan dispersas. si bien a partir de los 5000 individuos se abandonan casi todos estos mínimos locales las soluciones siguen estando distribuidas en entornos equivalentes. Con una población de 10000 individuos se obtiene un 40 % de aciertos

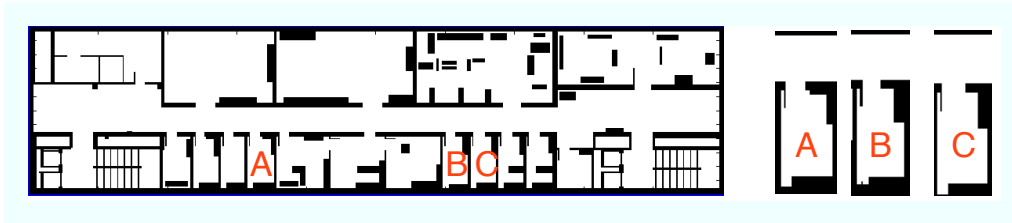


Figura 5.15: Áreas que presentan mínimos locales en el espacio de soluciones dentro del entorno.

cada una de las ubicaciones donde sitúa la soluciones, que tiene diferentes dimensiones en cada uno de los despachos. La ubicación A es prácticamente abandonada por la población a partir de los 5000 individuos, ya que sus dimensiones son lo suficientemente diferentes respecto a B y C.

Finalmente, para la población de 10000 individuos solo se presenta cierta incertidumbre entre B y C pero es razonable, de hecho la tasa de aciertos alcanza el 40 %, pero la distribución de poblaciones esta mucho mas concentrada en B, la ubicación objetivo, que en C, como se ve en la figura 5.14

Tamaño población	% aciertos	iteraciones medias	Promedio Soluciones Ensayadas
100	4	50	5000
200	0	50	10000
300	4	43,11	12933,33
400	2	42,22	16888,88
500	4	38,29	19147,06
1000	8	38,89	38894,74
2000	19	32,88	65769,23
3000	18	29,87	89600
4000	29	28,34	113358,49
5000	26	30	150000
10000	40	24,39	243855,42

Tabla 5.4: Variación del porcentaje de aciertos, las iteraciones medias y el promedio de soluciones ensayadas para la localización en $x=305$, $y=10$, $\theta=90$. Las tasas de acierto son sensiblemente inferiores al caso anterior, y el numero de soluciones ensayadas es muy superior.

El número de iteraciones necesarias para alcanzar la convergencia también es superior al necesario en el primer escenario. Se observa que no se alcanza saturación en las tasas de aciertos, debido a la distribución de las poblaciones entre las ubicaciones equivalentes. Figura 5.16

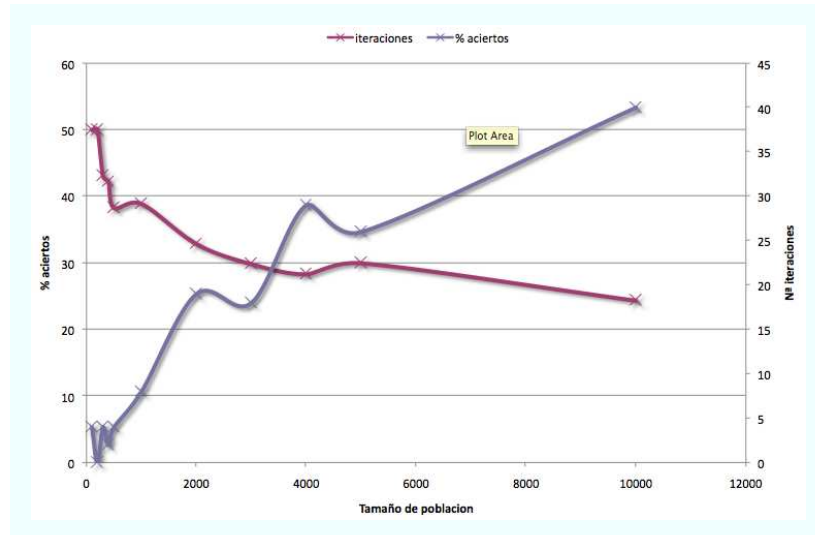


Figura 5.16: Distribución del número de aciertos y de iteraciones frente a los tamaños de población en $x=305$, $y=10$, $\theta=90$. En este caso no alcanzamos una región de saturación en la curva de aciertos debido a la existencia de múltiples localizaciones equivalentes

Aunque las tasas de acierto se mantienen bajas, este dato es un tanto engañoso: las poblaciones se sitúan en entornos equivalentes, que son soluciones correctas ante los datos de entrada disponibles, son mínimos locales del espacio de soluciones que poseen valores muy parecidos, y ante falta de información extra el algoritmo no tiene capacidad para descartar ninguno de ellos. Este resultado es correcto y deseable, falta información para realizar una correcta ubicación y el comportamiento algoritmo nos proporciona alternativas de localización. Un análisis estadístico de la distribución de soluciones podría asignar una probabilidad de acierto, en función de las localización, a cada una de los mínimos locales.

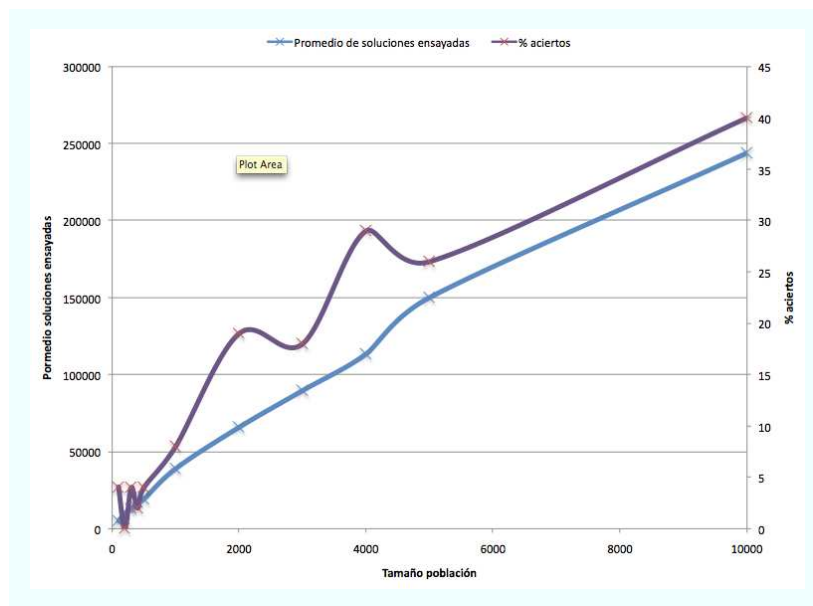


Figura 5.17: Evolución del número de soluciones ensayadas y la tasa de aciertos, frente al incremento de la población en $x=305$, $y=10$, $\theta=90$. Existe una relación cuasilineal entre el número de soluciones y la tasa de aciertos

La evolución de la tasa de aciertos con el número de soluciones ensayadas es lineal, y la tasa de aciertos se detiene en el 40 %, con una evolución ascendente, al operar con una población de 10000 soluciones. Eso nos invita a pensar que, al estar el algoritmo oscilando entre dos ubicaciones que poseen un mínimo local del espacio de soluciones muy parecido, Figura 5.17, tendrá una estabilización cercana al 50 %.

Para comprobar esta hipótesis realizamos la evaluación del algoritmo utilizando poblaciones ampliadas a 20000, 50000 y 70000, comprobando que la tasa de aciertos tienden a estabilizarse en el entorno de un 50 %. Figura 5.18, y también tiende a estabilizarse el número de iteraciones necesarias para alcanzar la convergencia.

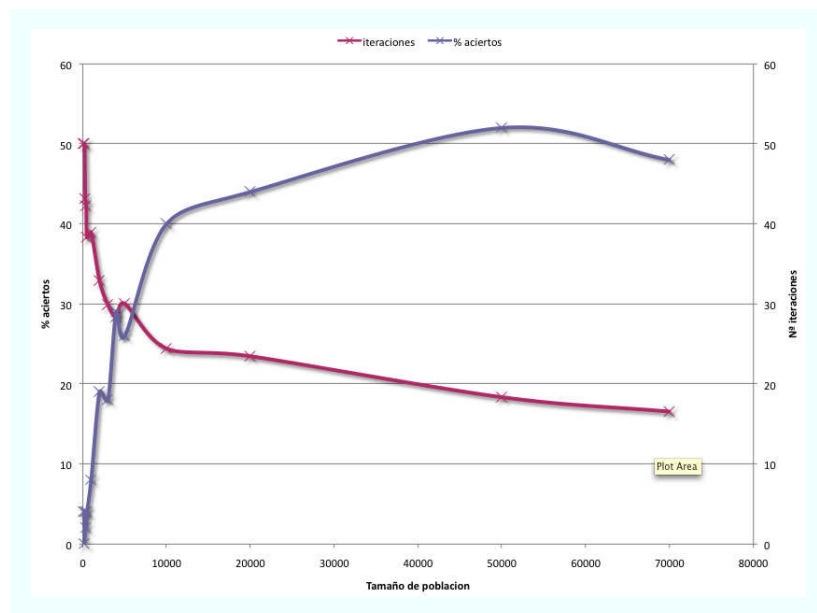


Figura 5.18: Evolución de las Tasas de acierto y el número de iteraciones con poblaciones ampliadas en la ubicación $x=305$, $y=10$, $\theta=90$, la tasas de aciertos tiende a estabilizarse en el entorno del 50 % debido a que el sistema oscila entre dos ubicaciones equivalentes.

Aunque los resultados son mejores en termino de porcentaje de aciertos y

de número de iteraciones necesarias para alcanzar la convergencia no lo es en cuanto al consumo de recursos necesarios para conseguirlo. El test con las poblaciones de 20000, 50000 y 70000 individuos ha tenido una duración de aproximadamente 60 horas en un Core 2 Duo a 2Ghz con 2 Gb de Ram

En la figura 5.19 se muestra la evolución espacial de una población de 5000 individuos. Rápidamente se abandonan los espacios alejados de la solución, iteración 4, y los individuos comienzan a agruparse en torno a la zona de localización y las equivalentes, iteraciones 6 y 8. Aquí, como en el caso anterior se aprecia el efecto del operador de cruce en las distribuciones espaciales, y del la modificación de la amplitud de mutación, que en este caso alcanza con rapidez un valor bajo, de ahí la baja dispersión de la población. Aun no se han concentrado en las zonas realmente equivalentes, sino que se encuentran distribuidos por zonas similares. Esta localización en las tres áreas equivalentes, A, B y C, figura 5.15, se hace mas notable a partir de la iteración 12, y al alcanzar la iteración 15 prácticamente solo quedan individuos en estas tres zonas.

La localización final se realiza en el área correcta, pero dada la similitud existente entre los tres mínimos locales hay individuos con altos valores de adaptabilidad en las tres áreas, por lo que no se abandona ninguna de ellas.

5.3.3. Entorno 3: el hueco de la escalera, $x=35$ $y=20$ $\theta=90$

La antesala de la escalera, figura 5.20, presenta un mínimo local muy estrecho, lo podemos visualizar como un pozo muy profundo en el espacio de soluciones, además en el entorno existe otra localización equivalente, al otro extremo del mapa, y a lo largo de todo el pasillo existe cierta similitud, esta multiplicidad en el entorno y lo angosto de su mínimo convierten a este punto en entorno de difícil localización

Como se aprecia en los histogramas de soluciones, figura 5.21, el algoritmo tiene serios problemas para alcanzar la convergencia con poblaciones menores

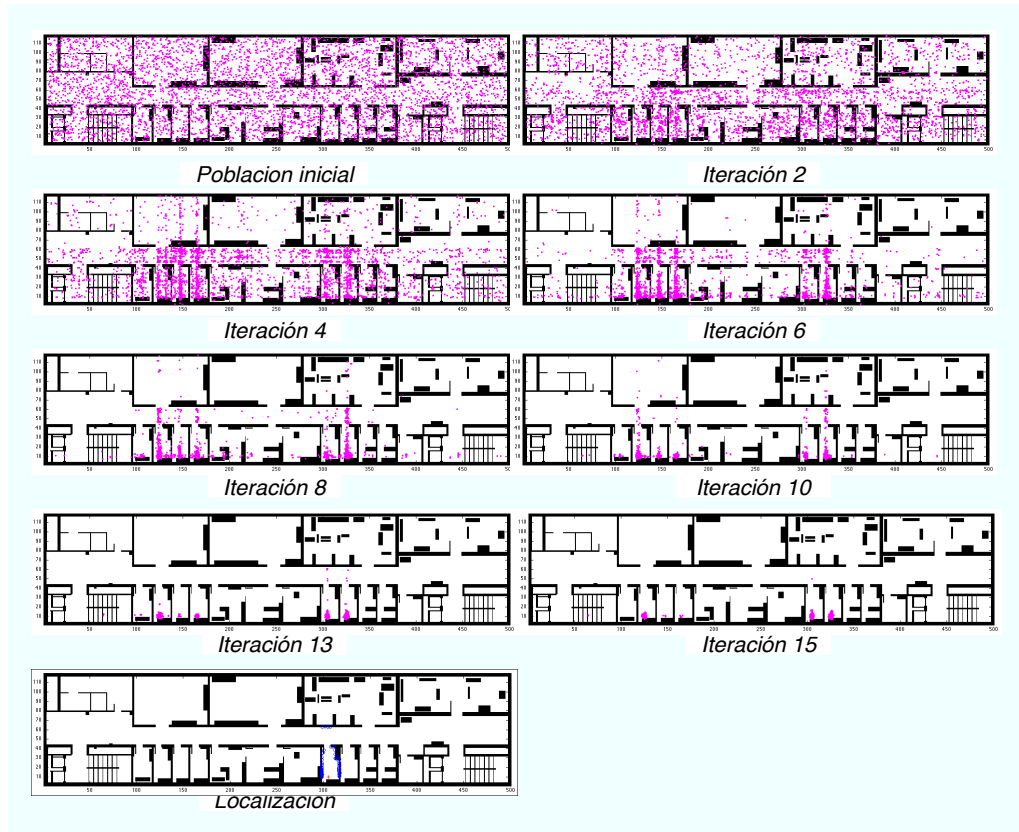


Figura 5.19: Distribución de una población de 5000 individuos durante 15 iteraciones para la localización en $x: 305, y=10, \theta=90$. La población se agrupa en torno a los mínimos locales existente, los cuales no llegan ser abandonados en ningún momento, como se ve en la iteración 15

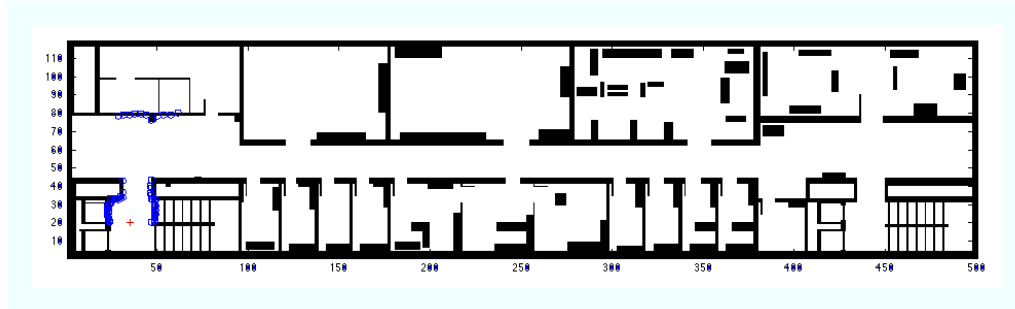


Figura 5.20: Localización en las coordenadas $x: 35, y=20, \theta=90$, tiene un mínimo local muy localizado y de difícil convergencia, con múltiples mínimos locales de valor similar

de 1000 individuos, presentando múltiples localizaciones posibles, sin que ninguna ellas aparezca como preferente. Existen múltiples localizaciones que atraen a la población, ya que, aunque el entorno equivalente es único, si existen múltiples configuraciones que ofrecen entornos similares en las que la población es dispersada

Gracias a los mecanismos se han introducido para evitar la convergencia temprana hacia un mínimo local, el uso de un elitismo de un 2.5 % de la población, y el uso de un 25 % como padres, el algoritmo no nos ofrece una respuesta exacta acerca de nuestra localización, sino que nos informa acerca de aquellos lugares donde es posible que estemos situados, no descartando ninguno de ellos, tal y como sucedía en el caso anterior.

Cuando la población aumenta a los 5000 individuos la localización es mas selectiva, con 10000 se observa claramente que la solución oscila entre dos posiciones, las posiciones equivalentes, con cierta tendencia hacia la correcta.

En realidad estas dos posiciones equivalentes difieren en la lectura del sensor en 6 puntos, figura 5.22, cuatro de estas lecturas se encuentran fuera de rango. En la evaluación de la función de salud las medidas fuera de rango no contribuyen, así que la diferencia de valor obtenido entre las dos ubicaciones es mínima, pero existe y afecta a la distribución de soluciones, como se aprecia en el histograma

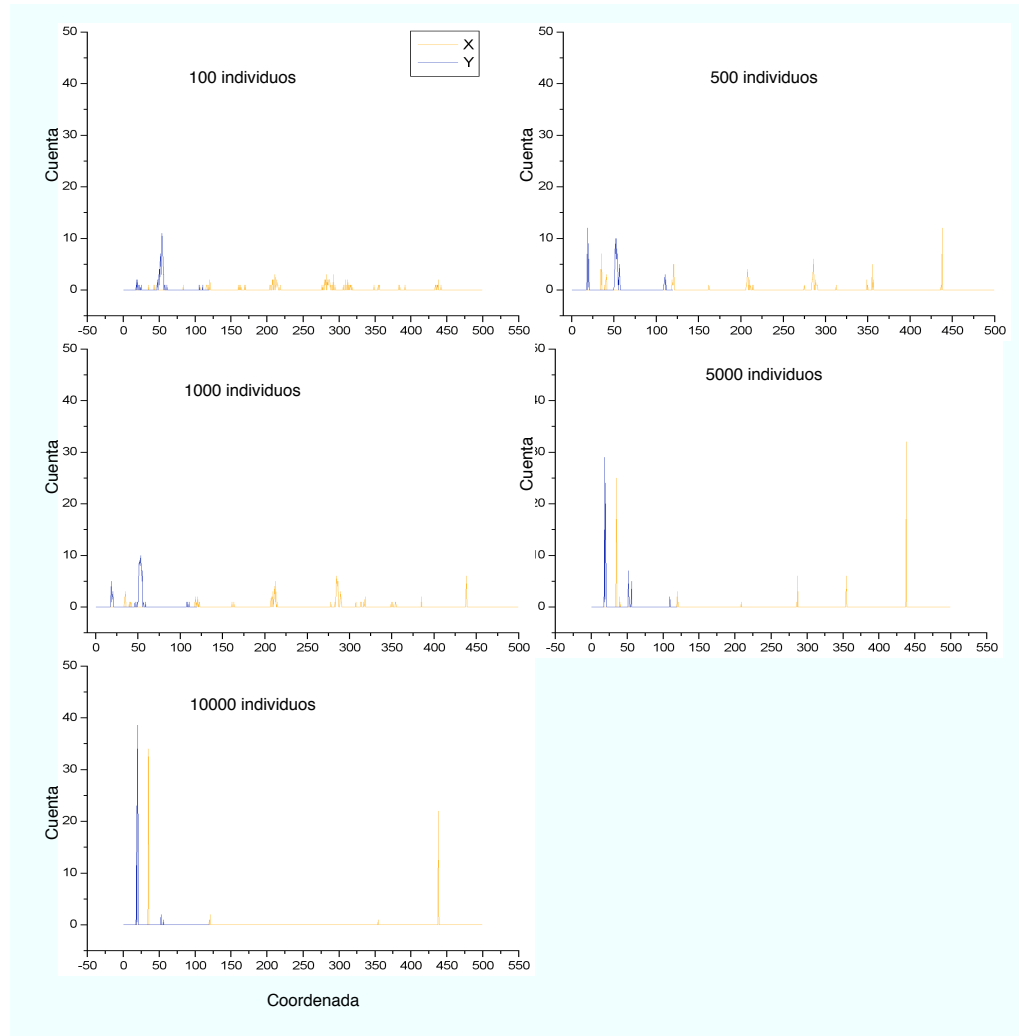


Figura 5.21: Distribución de los resultados a lo largo de 100 iteraciones con diferentes tamaños de población con la ubicación objetivo $x=20$, $y=35$, $\theta=90$. Al ser un máximo global muy localizado, un pozo muy abrupto en el espacio de soluciones, no se empiezan a obtener resultados significativos hasta alcanzar una población de 5000 individuos. Además, existe una ubicación equivalente en la coordenada $x=435$, y no existe localización sin ambigüedad hasta la población de 1000 miembros. La tasa de aciertos en este caso es del 33 %

de 10000 individuos.

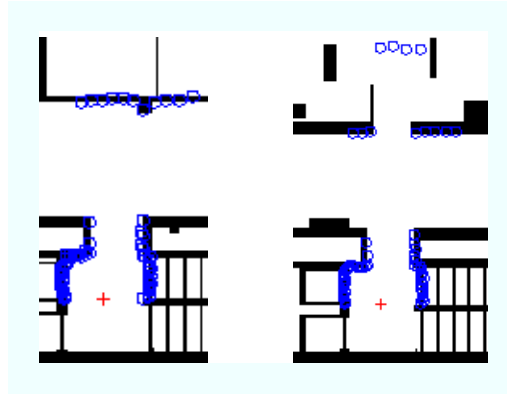


Figura 5.22: Lectura de los sensores en las posiciones equivalentes. Difieren en la estimación de 5 sensores, 4 de ellos se encuentran en el límite de rango, y por lo tanto no contribuyen a la evaluación

La convergencia no es buena hasta alcanzar un tamaño de población elevado, debido a que, como hemos visto, las poblaciones se encuentran muy distribuidas tabla 5.5, llegando a alcanzar una tasa de aciertos del 33 % con 10000 individuos.

El número de iteraciones necesarias para alcanzar la convergencia no desciende abruptamente a medida que aumentamos el tamaño de la población, esto es debido a la dispersión de la población, así como tampoco se estabiliza la tasas de aciertos, figura 5.23, el algoritmo tiene serias dificultades para decidir cual es la ubicación correcta. Si embargo, a partir del análisis del histograma si podemos estimar donde es mas probable que estemos situados

Podemos ver como el número de soluciones ensayadas, figura 5.24, sigue a la evolución de las tasas de acierto, sin que esta se llegue a estabilizar. Al igual que el caso anterior, existen multiplicidad de entornos equivalentes, o cuasi equivalentes, lo que hace que el algoritmo oscile en su decisión, pero se puede ver el histograma de respuestas que, para 10000 individuos (figura 5.21), existe una desviación estadística hacia la posición correcta.

Tamaño poblacion	% aciertos	Iteraciones medias	Promedio Soluciones Ensayadas
100	0	34	3400
200	1	11,33	2266,67
300	2	32,50	9750
400	1	24,25	9700
500	1	22,17	11083,33
1000	3	17,75	17750
2000	7	18,72	37454,54
3000	10	15,06	45187,5
4000	13	12,83	51310,34
5000	19	15,16	75813,95
10000	33	12,89	128873,24

Tabla 5.5: Distribución del porcentaje de aciertos, iteraciones medias necesitadas y el promedio de soluciones ensayadas para la ubicación $x=35$, $y=20$, $\theta=90$. El porcentaje de aciertos para poblaciones pequeñas es insignificantes debido a las dificultades intrínsecas del área de localización

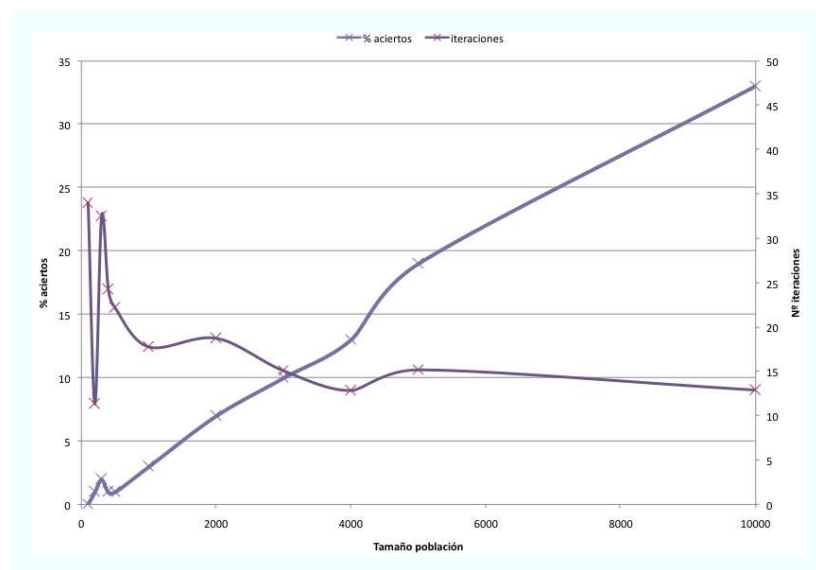


Figura 5.23: Distribución del número de aciertos y de iteraciones frente a los tamaños de población en $x=35$, $y=20$, $\theta=90$. En número de iteraciones necesarias permanece casi constante mientras que el porcentaje de aciertos aumenta considerablemente con el tamaño de la población.

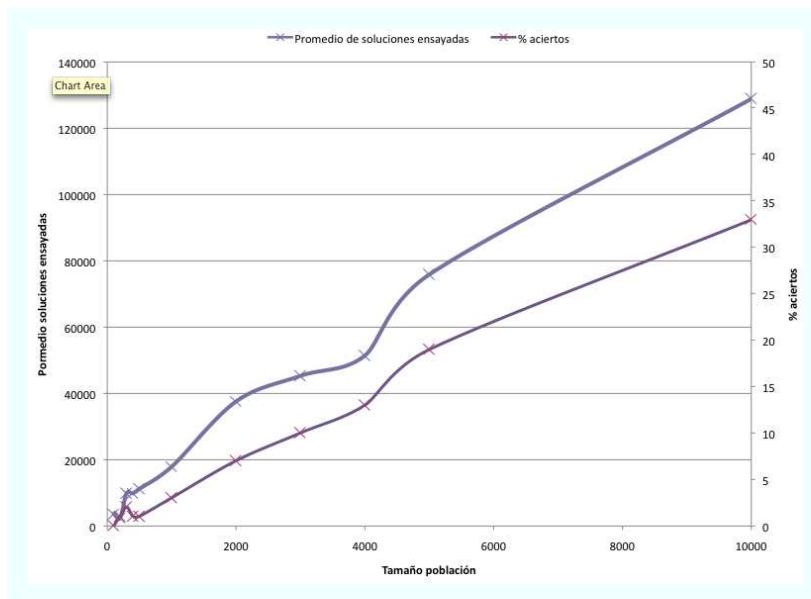


Figura 5.24: Evolución del número de soluciones ensayadas y la tasa de aciertos, frente al incremento de la población en $x=35$, $y=20$, $\theta=90$.

Es necesario incrementar el tamaño de la población para poder alcanzar cotas mas altas de localización efectiva, con el consiguiente incremento en el consumo de recursos.

En la figura 5.25 podemos ver la evolución de la distribución espacial de una población de 5000 individuos a lo largo de las diferentes iteraciones hasta alcanzar la convergencia.

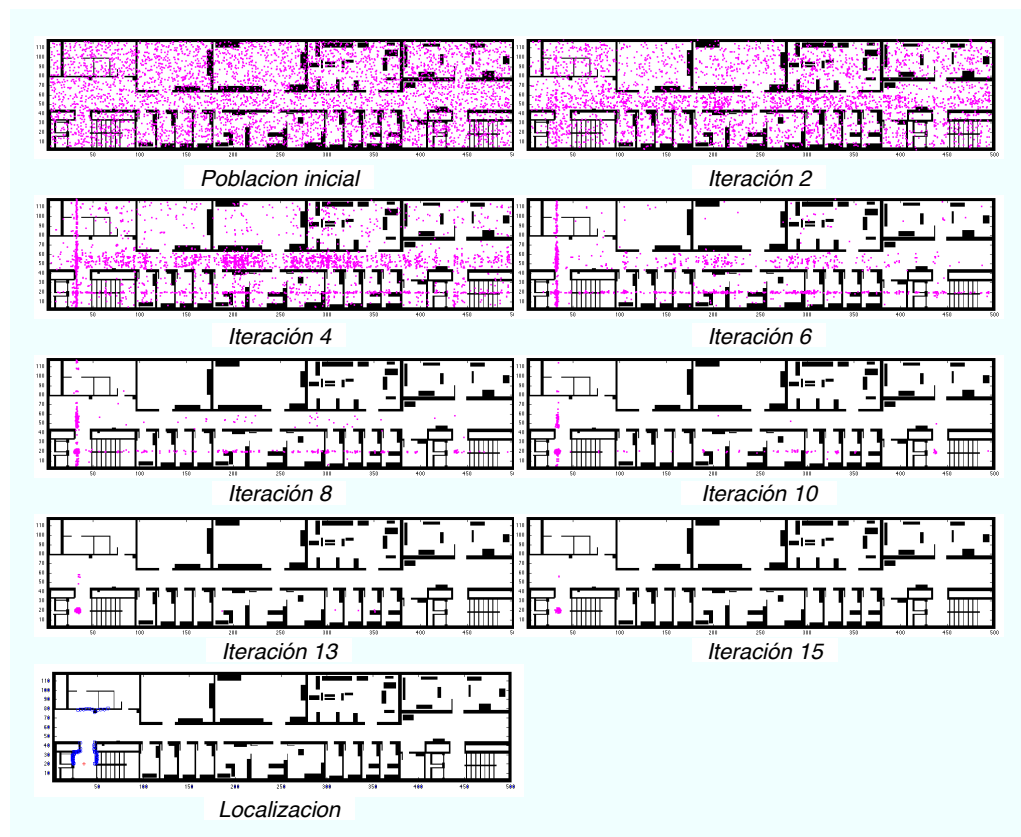


Figura 5.25: Distribución de una población de 5000 individuos durante 15 iteraciones para la localización en $x=20$, $y=35$, $\theta=90$. A lo largo del pasillo existen múltiples mínimos locales próximos al mínimo global del espacio, de ahí la distribución de la población, y la longitud que alcanza la distribución en cruz producto del operador de cruce. Debido a la estrecha localización del mínimo global, mínimo valor de la función de salud, que implica una mutación en entorno, la población se agrupa en el entorno cercano de los brazos.

Comenzamos con una población uniformemente distribuida, y en la iteración 2 podemos observar como ya los individuos van abandonando los espacios abiertos y se centran en aquellos cuyas dimensiones son parecidas a la de nuestro objetivo: el pasillo y los despachos. En la iteración 4 se comienza a apreciar el efecto del operador de cruce. Este punto posee un mínimo local muy estrecho en el espacio, por lo que cuando las soluciones están cerca de la solución la amplitud de la mutación da un salto brusco, por lo tanto los brazos de esa distribución en cruz poseen una dispersión espacial menor que la que vimos en el ejemplo de la localización en $x=40$, $y=60$ $\theta=0$. La razón de que los brazos se extiendan a lo largo de todo el espacio es la recombinación de los padres con toda la población: ésta se encuentra distribuida a lo largo de gran parte del entorno. A partir de la iteración 8 los individuos de la solución dominante van haciendo converger a toda la población hacia el área de relevancia, quedando pequeños reductos que desaparecen en unas pocas iteraciones.

Capítulo 6

Conclusiones

Se ha demostrado la viabilidad de la utilización de Algoritmos Genéticos Canónicos como solución eficiente al problema de la localización global en entornos conocidos.

Nos hemos centrado en la localización a partir de un solo muestro del terreno obteniendo resultados que resultan prometedores, obteniendo un alto grado de localizaciones correctas en los casos de baja multiplicidad espacial del entorno de localización. En cualquier caso se logra evitar la convergencia hacia un mínimo local, manteniendo la población distribuida a lo largo de las áreas que provocan la incertidumbre, situación que permite, mediante incorporación del movimiento, resolverla.

Para la incorporación del movimiento no es necesaria ninguna modificación en el algoritmo, solo requiere, una vez realizado el desplazamiento, la dispersión aleatoria y local de los individuos respecto a las posiciones anteriormente obtenidas, y volver a lanzar la localización. Partimos de este modo de una población concentrada alrededor de las áreas candidatas a la posición real, se consigue una rápida convergencia y un descarte de las áreas que, desde el nuevo punto de vista de los sensores, ya no sean equivalentes.

Comparado con otros métodos de localización basados en partículas, como

Monte Carlo, vemos que al igual que ellos la robustez tiene un alto grado de dependencia con el tamaño de la población. El test de Jensfelt's con el método de localización basado en Montecarlo con un tamaño de población de 10000 individuos y un promedio de 33 iteraciones, solo un 15 % de las localizaciones era satisfactoria. En nuestros peores entornos obtenemos, con $N=10000$ una efectividad del 33 %, y en promedio hemos necesitado 13 iteraciones. La posibilidad de trabajar directamente con datos en bruto del sensor y de acomodar altos niveles de ruido con robustez también ha sido demostrada, y brinda la posibilidad de abaratar los costes y complejidades del sistema de adquisición de datos del entorno, así como incrementar la fiabilidad global del sistema.

La elección de los parámetros de funcionamiento del algoritmo no es un asunto trivial, y, en este caso nos enfrentamos con un espacio de soluciones no continuo, esto es lo que provoca que el parámetro de mutación, $P_m = 0.4$ sea elevado. El algoritmo así adquiere el carácter estocástico necesario para salvar estas discontinuidades y explorar de forma efectiva el espacio de soluciones, sin perder a su vez las características que hacen del algoritmo un algoritmo evolutivo: la posibilidad de que el conocimiento adquirido a través de las diferentes generaciones sea heredado por las siguientes, llevándonos con unas razonables tasas de éxito a la convergencia en la solución adecuada.

El auge de los procesadores multinúcleo y de las tarjetas gráficas con multiprocesadores abre posibilidades a la implementación de este tipo de algoritmos, cuyo funcionamiento es intrínsecamente paralelo, en dichos sistemas utilizando el paradigma de programación paralela, lo que permitirá alcanzar grandes velocidades de ejecución, aun con poblaciones de un elevado número de individuos, en sistemas autónomos.

Aun queda mucho trabajo por hacer en la aplicación de los algoritmos genéticos en el problema de la localización global, nuestro objetivo era demostrar la viabilidad práctica de la aplicación a este tipo de problemas. Para ello lo hemos

implementado en su forma canónica, y hemos obtenido unos resultados satisfactorios que abren la puerta a un desarrollo futuro de esta línea de aplicación.

Capítulo 7

Lineas Futuras

Como líneas futuras para la extensión de este trabajo proponemos:

- Su uso en conjunto con otros métodos iniciales de localización, como balizas, que permitan inicializar las poblaciones en el entorno cercano a la solución, que por lo tanto podrían ser poblaciones de un tamaño reducido, podría aportar precisión y rapidez a la convergencia.
- Incorporación del movimiento para poder resolver las incertidumbres debidas a los mínimos locales
- Hemos comprobado como cuando existen múltiples entornos equivalentes la respuesta del sistema oscila entre unos y otros, distribuyendo las respuestas a lo largo de ellos. Un análisis estadístico de múltiples lanzamientos del algoritmo podría ofrecer respuestas mas exactas acerca de nuestra posición, mejorando la eficiencia computacional frente a la solución de incrementar el tamaño de las poblaciones.
- Se sugieren pruebas de dinamización de tamaños de población, y de parámetros de mutación y combinación para mejorar la velocidad. La evolución del porcentaje de aciertos, junto con un análisis estadístico de las respuestas que se van obteniendo pueden ofrecer información de como ha de ser

este cambio en el tiempo para maximizar su eficiencia

- Implementación en procesadores multinúcleo o tarjetas gráficas con arquitectura paralala: Hemos observado como cuando nos enfrentamos a entornos que poseen múltiples mínimos locales equivalentes el sistema tiene dificultades para ofrecer una respuesta correcta con poblaciones pequeñas, pero las grandes poblaciones requieren también una gran potencia de cálculo. Una implementación en un entorno multiprocesador ofrecería una posible solución a este incremento de la necesidad de capacidad de cálculo. Asimismo proponemos el desarrollo de una interfaz remota, que permita al robot utilizar sistemas externos, accesibles mediante tecnologías wireless en el caso de necesitar mayor potencia de cálculo de la disponible localmente.
- Clusterización de la población: La división de la población en islas de individuos, que exploren de forma independiente el espacio de soluciones, y compitan entre ellas cada cierto tiempo a fin de descartar unas y potenciar otras en función de su grado medio de salud podría ofrecer una mejora de funcionamiento
- Ajustes de parámetros en múltiples entornos. En nuestro trabajo hemos seleccionado los parámetros óptimos basándonos en un entorno de baja dificultad a la hora de la localización. Es de esperar que diferentes entornos con diferentes características tengan unos parámetros óptimos diferentes. Una vez conocidos estos parámetros servirán para dinamizar su valor en función de la evolución del algoritmo.
- Probar el algoritmo propuesto en un entorno cambiante: La localización global debe ser robusta ante cambios ligeros en el entorno, como la existencia de nuevos obstáculos como personas o muebles en sus inmediaciones, el hecho de que el algoritmo pueda acomodar elevados niveles de ruido nos hace esperar buenos resultados en este sentido.

Bibliografía

A. E. Eiben, E. H. L. A., & Hee, K. M. V. (1991). *Global convergence of genetic algorithms: A markov chain analysis*. Parallel Problem Solving from Nature, H.-P. Schwefel and R. Manner, Eds. Berlin and Heidelberg: Springer, pp. 4-12.

D Fox, S. T., W Burgard. (1999). *Markov localization for mobile robots in dynamic environments*. Journal of Artificial Intelligence Research.

D.J. Austin, P. J. (2000). *Using multiple gaussian hypotheses to represent probability distributions for mobile robot localization*. Proc. of the Int. Conference on Robotics and Automation ICRA-00, San Francisco, CA, USA, pp. 1036–1041.

F Dellaert, W. B. S. T., D Fox. (1999). *(monte carlo localization for mobile robots*. Robotics and Automation, Proceedings. 1999 IEEE International Conference on.

Fogel, D. B. (1992). *Evolving artificial intelligence*. PhD dissert., San Diego: University of California.

Holland, J. H. (1968). *Hierarchical descriptions of universal spaces and adaptive systems*. Technical Report ORA projects 01252 and 08226. University of Michigan, Department of Computer and Communications Sciences.

Holland, J. H. (1975). *Adaptation in natural and artificial systems*. The University of Michigan Press.

- I.J. Cox, J. L. (1994). *Modeling a dynamic environment using a bayesian multi hypothesis approach*. Artificial Intelligence 66 311–344.
- J.Reuter. (2000). *Mobile robot self-localization using pdab*. Robotics and Automation. Proceedings. ICRA '00. IEEE International Conference on.
- K. Arras, R. S., J. A Castellanos. (2002). *Feature-based multi-hypothesis localization and tracking form mobile robots using geometric constraints*. Proc. of the Int. Conference on Robotics and Automation, ICRA-02Washington DC, USA ,pp. 1371–1377.
- L. Moreno, M. M., S. Garrido. (2006). *Evolutionary filter for robust mobile robot global localization evolutionary filter for robust mobile robot global localization evolutionary filter for robust mobile robot global localization evolutionary filter for robust mobile robot global localization evolutionary filter for robust mobile robot global localization*. Robotics and Autonomous Systems 54 (2006) 590–600.
- M.H. Kalos, P. W. (1986). *Monte carlo methods*. Basics, vol I, John Wiley and Sons.
- P Jensfelt, D. A. M. A., O Wijk. (2000). *Experiments on augmenting condensation for mobile robot localization*. Robotics and Automation, 2000. Proceedings. ICRA '00. IEEE International Conference on 2000 vol. 3 pp. 2518 - 2524 vol.3.
- P. Jensfelt, S. K. (1999). *Active global localisation for a mobile robot using multiple hypothesis tracking*. Proc. IJCAI Workshop on Reasoning with Uncertainty in Robot Navigation, Stockholm, Sweden, pp. 13–22.
- Porto, D. F. y. L. F., Vincent. (1995). *Alternative neural network training methods*. IEEE Expert, vol.10, no.3, p.16-22.
- R. Storn, K. P. work with the robot. (1997). *Differential evolution—a simple and efficient adaptive scheme for global optimization over continuous spaces* (K. A. publishers, Ed.). Journal of Global ReferencesOptimization 11 ,341–359.

Sato, K. O. A. T. H. S. Y. A. y. H. K., S. (2002). *Applying genetic algorithms to the optimum design of a concert hall*. Journal of Sound and Vibration, vol.258, no.3, p. 517-526.

Sebastian Thrun, W. B. F. D., Dieter Fox. (2001). *Robust monte carlo localization for mobile robots*. Artificial Intelligence 128 (2001) 99–141.

S.I. Roumeliotis, G. B. (2000). *Bayesian estimation and kalman filtering: A unified framework for mobile robot localization*. Proc. IEEE International Conference on Robotics and Automation, ICRA-2000, San Francisco, pp. 2985–2992.

Thrun, S., Bennewitz, M., Burgard, W., Cremers, A., Dellaert, F., Lakemeyer, G., et al. (1999). *Experiences with two deployed interactive tour-guide robots*. Artificial Intelligence 114 (1–2) 3–55.

Thrun, S., Fox, D., Burgard, W., & Dellaert, F. (2001). *Robust monte carlo localization for mobile robots*. Artificial Intelligence.

Wolfram Burgard, D. H. T. S., Dieter Fox. (1996). *Position tracking with position probability grids*. Proc. of the National Conference on Artificial Intelligence AAAI-96, Potland, Oregon, USA pp. 896–901.

Xiaofeng Qi, F. P. (1994). *Theoretical analysis of evolutionary algorithms with an infinite population size in continous space partii: Analysis of the diversification role of crossover*. IEEE Transactions on neural Networks, Vol 5, No 1, January.

Z. Michalewicz, C. Z. J. (1991). *Handling constraints in genetic algorithms*. Proceedings of the Fourth International Conference on Genetic Algorithms; pp. 151.