

ALGORITMO GENÉTICO PARA LA UBICACIÓN ÓPTIMA DE SENSORES EN UN ROBOT SEGUIDOR DE LÍNEA.

Genetic algorithm to the optimal location of sensors on a robot line follower.

RESUMEN

En este artículo se presenta un algoritmo genético empleado para ubicar de forma óptima los sensores en un robot seguidor de línea.

PALABRAS CLAVES: Algoritmo genético, robot seguidor de línea.

ABSTRACT

In this article is presented a genetic algorithm used to locate in an optimal way, the sensors in a robot line follower.

KEYWORDS: *Algoritmo genético, robot seguidor de línea.*

CARLOS RESTREPO PATIÑO

Ingeniero Electricista, M. Sc.
Estudiante de Doctorado en
Ingeniería Electrónica
Universitat Rovira I Virgili
carlos.restrepo@urv.cat

JASON MOLINA VARGAS

Ingeniero Electricista, M. Sc (c).
Analista de Tecnología.
Xm los expertos en mercado
jmolina@xm.com.co

CARLOS ANDRÉS TORRES P.

Ingeniero Electricista, M. Sc.
Estudiante de Doctorado en
Ingeniería Electrónica
Universitat Rovira I Virgili
carlosandres.torres@urv.cat

**Grupo de Investigación en
Electrónica de Potencia
Universidad Tecnológica de
Pereira**

1. INTRODUCCIÓN

Uno de los problemas clásicos en la navegación de robots es seguir trayectorias. El problema que surge al momento de implementar un seguidor de línea es determinar la posición en la que se deben ubicar los sensores para garantizar que el robot no pierda la trayectoria sin importar la forma que ella tenga.

Los algoritmos genéticos constituyen una herramienta matemática que se utiliza para resolver problemas asociados con la búsqueda de soluciones óptimas.

Se basan en la teoría de la evolución de los seres vivos en la naturaleza. La mecánica en este tipo de algoritmos se fundamenta en el hecho de partir de un conjunto posible de soluciones, donde a cada una de ellas se les conoce como individuos y al grupo en general como población. A esta población se le aplica un conjunto de operadores llamados operadores genéticos, de los cuales se consideran básicos los siguientes: la selección, la mutación y el cruce. Junto a estos el otro elemento fundamental es la función de evaluación la cual va a medir cual de los individuos involucrados en el proceso es más apto y por tanto se acerca más a la solución óptima. La principal característica de los algoritmos

genéticos, es la robustez, particularidad que otros métodos de optimización tradicionales como los métodos matemáticos basados en el cálculo o los heurísticos no tienen.

La naturaleza paralela es lo que les da esa potencialidad a los algoritmos genéticos ya que buscan la solución en varios puntos (individuos) de una población a la vez y no en uno solo; sumado a lo anterior, la información de los parámetros esta codificada, por lo que no tienen que trabajar directamente con ellos. Por último una ventaja adicional es que usan reglas probabilísticas y no determinísticas. Los algoritmos genéticos requieren que los parámetros involucrados en el proceso de optimización estén codificados en algún alfabeto; generalmente se prefiere por la naturaleza digital involucrada, que sean cadenas binarias, aunque también se trabaja con números reales.

El soporte matemático sobre el cual se sustentan los algoritmos genéticos, corresponde a lo que Holland, su creador, dio por nombre como esquema. Un esquema es un patrón que describe un conjunto de similitudes entre un grupo de cadenas ubicadas en ciertas posiciones de los individuos partícipes del proceso de optimización. Para comprender mejor porque funcionan los algoritmos

genéticos, se debe observar el efecto que tienen los operadores de selección, cruce y mutación sobre determinado esquema. El efecto de la selección sobre los esquemas, se resume en lo siguiente: para esquemas con función de adaptación por encima de la media, tienden a proliferar en las generaciones siguientes. El segundo efecto se da para aquellos esquemas con función de evaluación por debajo de la media, los cuales tienden a desaparecer. Los esquemas cortos tienen mayor probabilidad de sobrevivir después del proceso de cruce. Para la mutación, los esquemas de orden bajo, tienen mayores posibilidades de sobrevivir.

En este artículo se implementa un algoritmo genético para la ubicación óptima de los sensores en una plataforma móvil. Se desarrolla el modelamiento del movimiento de la plataforma y de la trayectoria que se recorrerá.

2. CONTENIDO

La evaluación de la función objetivo en el algoritmo genético implica que se tenga un modelo para el posicionamiento de la plataforma y otro modelo para la trayectoria que se recorrerá. Se desarrollará a continuación un modelo matemático, tanto para la posición del robot como para la trayectoria a seguir; los cuales serán utilizados posteriormente en el algoritmo genético.

2.1 Construcción de la trayectoria

La trayectoria con la que se evaluará cada una de las configuraciones deberá tener un recorrido muy complejo con diferentes curvas y ángulos de quiebre, como las que se presentan en los diferentes competencias de robótica. Otra característica de la trayectoria es que tiene un ancho constante en todo su recorrido.

La trayectoria que sigue la plataforma se ha definido mediante dos líneas, las cuales deben conservar un ancho igual en todos sus puntos. Para esto se construye primero una curva base que es la unión de:

- Un período de una curva sinusoidal
- Un período de una curva triangular con ángulo agudo.
- Un período de una curva triangular con ángulo recto.

Con base en la anterior curva, se construye otra, cuyos puntos tengan una distancia fija a la curva original. Para esto se utiliza un algoritmo que calcula cada punto de la curva utilizando el hecho de que la distancia perpendicular a la curva original sea constante. La distancia perpendicular se calcula con base en la recta normal a la curva original. De hecho, si la curva original es:

$$y = f(x)$$

Entonces la pendiente de la recta normal en el punto x es:

$$m = -\frac{1}{\frac{dy}{dx}} = -\frac{1}{f'(x)}$$

Lo anterior sirve como base para calcular el punto cuya distancia a la curva original es r , dado que calculando el ángulo θ que forma la recta normal con la horizontal se calcula el punto (x_1, y_1) de la otra curva. Lo que se utiliza es lo siguiente:

$$x_1 = x - r \cos(\arctan(m))$$

$$y_1 = y + r \cos(\arctan(m))$$

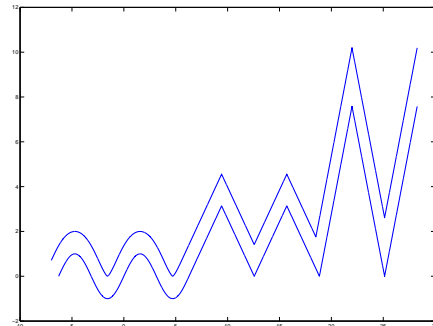


Figura 1. Trayectoria de prueba.

El proceso se repite con los puntos de la curva original hasta construir la otra curva; en la figura 1 se presenta la trayectoria obtenida. El proceso debe hacerse discretizando la curva original, dado que el problema de determinar una expresión analítica de la segunda curva, implica resolver una ecuación diferencial no lineal.

2.2 Movimiento del robot

Para localizar un cuerpo rígido en el espacio es necesario contar con una herramienta que permita la localización espacial de sus puntos. La forma más intuitiva y utilizada para especificar la posición de un punto son las coordenadas cartesianas. En un plano, el posicionamiento tiene dos grados de libertad, y por tanto la posición de un punto está definida por dos componentes independientes.

Para la descripción del movimiento de una plataforma móvil en un plano bidimensional se requieren varios aspectos:

Plantear un sistema de referencia fijo y un sistema de referencia móvil para la plataforma.

Formular una expresión con la cual se pueda determinar la posición del sistema de referencia móvil, en función de un giro y una traslación.

Especificar las posiciones de los elementos que constituyen la plataforma, referenciados al sistema móvil.

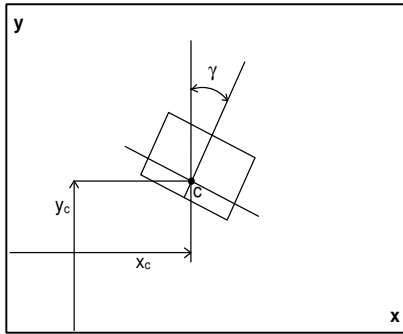


Figura 2. Posición de la plataforma.

El sistema de referencia fijo se escoge como el sistema de coordenadas cartesianas; por otro lado conviene especificar el sistema de referencia móvil como la posición de un punto representativo en la plataforma, referente a las coordenadas cartesianas y la posición angular de la plataforma respecto al sistema, como se indica en la figura 2.

El punto c es fijo en la plataforma y tiene coordenadas (xc,yc) y el ángulo γ, se toma con referencia al eje vertical y se define positivo el sentido horario.

Una nueva posición del centro c puede ser determinada por un cambio en el ángulo γ y un desplazamiento en dicha dirección, como:

θ = Desplazamiento angular de la plataforma, o giro.

l = Longitud recorrida por la plataforma.

La figura 3 presenta una descripción de lo anterior. Las coordenadas de c2 son (xc2,yc2) y se calculan así:

$$x_{c2} = x_{c1} + l \cos\left(\frac{\pi}{2} - (\gamma_1 - \theta)\right)$$

$$y_{c2} = y_{c1} + l \sin\left(\frac{\pi}{2} - (\gamma_1 - \theta)\right)$$

Luego

$$x_{c2} = x_{c1} + l \cos(\gamma_1 + \theta)$$

$$y_{c2} = y_{c1} + l \sin(\gamma_1 + \theta) \tag{1}$$

y la nueva inclinación γ se da por:

$$\gamma_2 = \gamma_1 + \theta \tag{2}$$

De esta forma se puede describir la cinemática de la plataforma, dado que el desplazamiento puede darse directamente, o en términos de velocidades; con el fin de hallar el giro θ y el desplazamiento l y evaluar (1). Con las ecuaciones (1) y (2) se puede calcular además la posición de cualquier elemento en la plataforma, siguiendo el método de girar y luego avanzar.

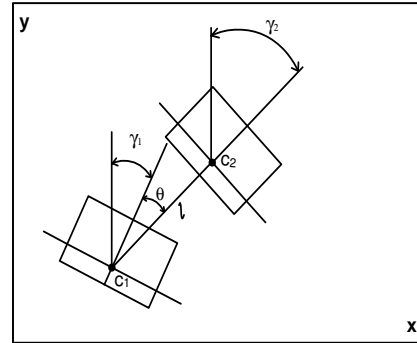


Figura 3. Movimiento de la plataforma.

3. FORMULACIÓN DEL PROBLEMA

La ubicación óptima de los sensores en la plataforma móvil conlleva a que ella no se saldrá de la trayectoria, no regresara por la trayectoria ya recorrida, cuando el sensor principal pierda línea sólo alguno de los otros dos lo respaldará para corregir la orientación. Todas estas condiciones ocasionan que la plataforma realice un recorrido considerado en la trayectoria ya analizada. Cada una de las configuraciones de la población de soluciones recorre la pista, y se determina la longitud l recorrida. Por lo anterior, el modelo toma la forma:

$$\max \mathcal{L} = \sum_{i=1}^K l_i$$

Donde K es el número de pasos realizados por la plataforma.

Sujeto a:

- Si $S_1 \notin R \Rightarrow$ existe sólo un $j \neq 1$ tal que $S_j \in R$.
- Para $i = 1 : S_1 \in R, y, S_j \notin R \forall j \neq 1$.
- $x_c(i+1) \geq x_c(i)$

Siendo R la región comprendida entre las dos curvas, i es el número del paso actual de la plataforma ($1 \leq i \leq k$) y S_j es el j -ésimo sensor.

4. ALGORITMO GENÉTICO

En la naturaleza, los individuos mejor dotados genéticamente tienen mayor capacidad de supervivencia cuando los recursos son escasos o cuando cambian las condiciones del medio ambiente. Mejor dotados significa mejor capacidad de supervivencia, y, en última instancia, esta cualidad o capacidad es determinada por el contenido genético de cada individuo.

Las alteraciones y la diversificación del material genético constituyen la esencia de la evolución. El algoritmo genético usa una población de individuos, que en los problemas combinatoriales representan un conjunto de configuraciones, para resolver un problema de optimización complejo.

A continuación se presenta cada una de las características del algoritmo para la ubicación óptima de los sensores en la plataforma móvil.

4.1 Codificación

La codificación decimal, para representar variables enteras, tiene como ventaja el usar un vector de tamaño reducido, mejorando los problemas de memoria para almacenamiento de información y velocidad de procesamiento porque generalmente la función objetivo esta en función de las variables enteras originales.

El principal inconveniente de la codificación decimal es que teóricamente no son necesariamente verdaderas todas las propiedades teóricas y las características de convergencia encontradas en el algoritmo genético usando la codificación binaria. También en la denominada codificación decimal tiene que redefinirse el operador mutación.

La codificación que mejor se adapta a este problema es la decimal, con la cual se representa la posición de cada uno de los tres sensores en coordenadas cilíndricas. Esta codificación tiene como ventaja que usa un vector muy reducido para cada sensor, de sólo dos componentes, el ángulo y la distancia, obteniendo de esta forma una mayor velocidad de procesamiento.

Cada una de las posibles configuraciones está formada por un vector C_i de seis componentes, en donde cada par de componentes representa la posición de uno de los sensores como se muestra en la figura 4. De esta forma se tiene que la población de soluciones esta conformada así:

$$C_i = [l_1^i \quad \theta_1^i \quad l_2^i \quad \theta_2^i \quad l_3^i \quad \theta_3^i]$$

4.2 Población inicial

Se determinará el tamaño de la población inicial N teniendo como criterio la complejidad de la trayectoria a recorrer. La forma en la cual se genera la población inicial es aleatoria controlada.

$$C = \begin{bmatrix} l_1^1 & \theta_1^1 & l_2^1 & \theta_2^1 & l_3^1 & \theta_3^1 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ l_1^N & \theta_1^N & l_2^N & \theta_2^N & l_3^N & \theta_3^N \end{bmatrix}$$

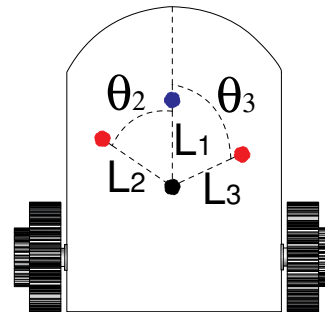


Figura 4. Configuración de sensores.

4.3 Cálculo de la función objetivo

La función objetivo determina la calidad de una configuración para de esta manera tener una selectividad entre las configuraciones.

Para cada una de las configuraciones del vector de soluciones se determina la longitud de la trayectoria recorrida. La configuración con mayor longitud será el diseño que podrá tener más probabilidad de tener más descendientes en la próxima generación.

Las longitudes correspondientes a cada configuración son calculadas con el algoritmo de la figura 5. Obteniendo el vector:

$$L = [L_1 \quad L_2 \quad \dots \quad L_{N-1} \quad L_N]^T$$

La función de adaptación que se emplea en este algoritmo es igual a la función objetivo (f.o) de cada configuración menos un factor k , el cual se define como:

$$K = 0.9 \cdot \min(L)$$

de esta forma se tiene la función de adaptación:

$$FA = [L_1 \quad L_2 \quad \dots \quad L_{N-1} \quad L_N]^T - K$$

4.4 Selección

Este operador genético permite seleccionar las configuraciones de la población actual que deban participar en la generación de las configuraciones de la nueva población. Se emplea un esquema de selección proporcional, es decir, los descendientes a los que tiene derecho cada configuración es proporcional al valor de su función de adaptación. Para determinar el número de descendientes de cada configuración se hace una relación entre la función de adaptación y la media de la función objetivo. El inconveniente radica en que los descendientes en su mayoría pueden ser números no enteros. Para tener un número de descendientes enteros se emplea el esquema de la ruleta.

4.5 Recombinación

El operador de recombinación consiste en escoger dos configuraciones en forma aleatoria entre las que tienen derecho a descendencia y generar a partir de ellas dos nuevas configuraciones que tienen partes de las configuraciones originales. La recombinación en este algoritmo es de punto simple. Existe una tasa de recombinación del 90%, y la forma de recombinar es generar un número aleatorio entre [0,1] y si éste es menor a 0.9 se realiza la recombinación. De lo contrario, a la próxima generación se pasan las dos configuraciones sin recombinación.

4.6 Mutación

Se analiza cada componente de las configuraciones de la población inicial, se asigna un factor de mutación del 5% y se genera un número aleatorio; si dicho número es menor o igual a la tasa de mutación se realiza una mutación en esa posición. La mutación que se emplea es de tipo simple, es decir, se convierte el número a un valor una tolerancia mayor o menor a él, para esto se genera otro número aleatorio para conocer si la mutación aumenta o disminuye el número seleccionado. Se analiza si cada una de las configuraciones de la nueva generación cumple con un recorrido mínimo, en caso contrario se continua haciendo mutaciones sucesivas hasta que se cumpla un recorrido mínimo como se ilustra en el figura 5.

4.6 Criterio de Parada

El algoritmo finalizará una vez se halla realizado un número específico de generaciones.

5. RESULTADOS

En la tabla I se presenta los resultados obtenidos por el algoritmo genético con un Δl y un $\Delta\theta$ fijos, las variaciones que se aprecian en estos valores son debido a mutaciones. G es la generación en la que se obtuvo la configuración y L es la longitud recorrida por ella. Para todos los procesos se empleo una población de 20 individuos y 1000 generaciones. En la figura 6 se observa la solución grafica de la mejor configuración obtenida en la tabla I.

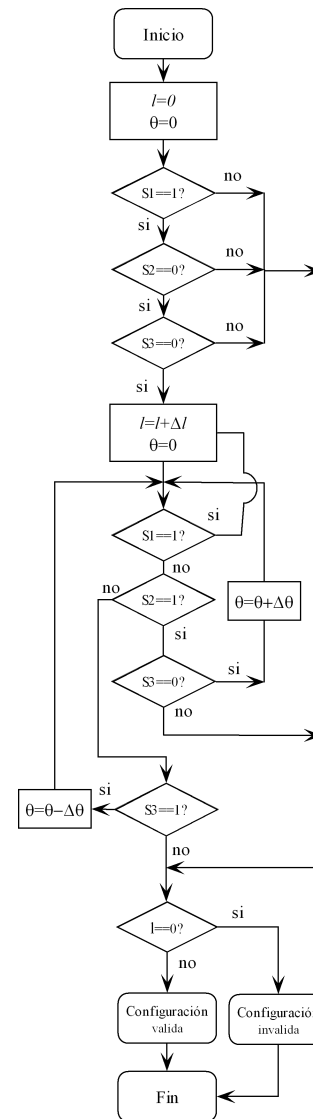


Figura 5. Diagrama de flujo del recorrido.

Tabla I. Resultados con Δl y un $\Delta\theta$ fijos.

Δl	$\Delta\theta$	l_1	θ_1	l_2	θ_2	l_3	$\theta_3(-)$	G	L
0.10	0.10	0.40	0	0.80	36.5	0.5	121.0	1	13,20

0.10	0.10	0.36	0	0.80	36.5	0.5	121.0	4	13,30
0.11	0.1	0.36	0	0.80	36,5	0,5	111.0	7	13,31
0.11	0.10	0.36	0	0.70	97.5	0.3	27.2	12	17,27
0.09	0.10	0.32	0	0.72	97.5	0.3	27.2	19	17,37

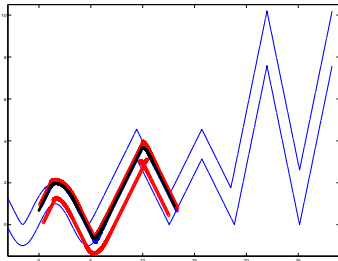


Figura 6. Mejor solución Tabla I.

En la tabla II se presentan los resultados optimizando, además de la configuración de sensores, el paso al que revisa los sensores Δl y el ángulo que girará una vez pierda línea $\Delta \theta$. La generación en la que fue hallada cada configuración como la longitud recorrida por ella se muestran en la tabla III. En las figuras 7 y 8 se observa la solución grafica de las dos mejores configuraciones de la tabla II.

Tabla II. Resultados con Δl y un $\Delta \theta$ variables.

Δl	$\Delta \theta$	$t1$	$\theta1$	$t2$	$\theta2$	$t3$	$\theta3(-)$
0.36	1.093	0.6	0	0.8	132.9	0.9	31.7
0.36	1.093	0.5	0	0.7	54.8	0.4	106.8
0.324	10.056	0.2	0	0.7	97.5	0.4	106.8
0.324	1.093	0.2	0	0.77	105.3	0.4	115.34
0.2916	10.056	0.18	0	0.77	113.72	0.4	115.34
0.3564	1.093	0.22	0	0.7623	105.3	0.44	124.57
0.36	1.093	0.2	0	0.3	85.847	0.8	142.77

Tabla III. Generación y recorrido de tabla II.

G	1	3	10	15	34	50	187
L	6.84	13.68	29.48	40.18	40.53	40.63	48.87

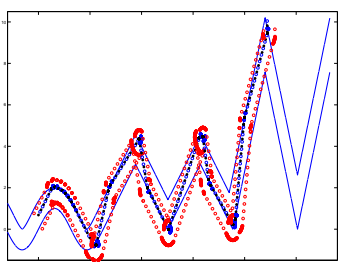


Figura 7. Segunda mejor solución Tabla II.

6. CONCLUSIONES

Los modelos matemáticos desarrollados para definir la trayectoria y el movimiento del robot y a su vez el algoritmo para seguir línea permitieron obtener una función objetivo muy selectiva para cada una de las configuraciones. El algoritmo genético encontró una buena solución que permitió realizar un recorrido considerable en una trayectoria de alta dificultad.

Se han formulado las expresiones correspondientes a la ubicación de la plataforma móvil, a partir de un giro y una longitud recorrida; las cuales son necesarias para que el algoritmo tome la decisión en lo concerniente al movimiento; además se ha planteado la manera de adecuar trayectorias generales sobre las cuales se tenga la forma de trayectorias reales. Debe notarse que mientras más general sea la forma de modelar el problema del movimiento de la plataforma, más general será el resultado que proporcione el algoritmo, lo cual indica que el resultado obtenido funcionaría para casos similares a los modelados.

7. BIBLIOGRAFÍA

- [1] Gallego Ramón - Escobar Antonio - Romero Rubén. Técnicas de optimización combinatorial, Primera edición, 176 páginas, Universidad Tecnológica de Pereira, Colombia, 2006.
- [2] Goldberg David. Genetic Algorithms, in search optimization and machina learning, Primera edición, 412 páginas, Addison Wesley, ISBN 0201157675, 1989.
- [3] Glover F.- Kochenberger G. A. Handbook of metaheuristic. Kluwer Academic Publishers. Norwell. M.A. 2003.
- [4] Barrientos Antonio - Peñin Luis Felipe - Balaguer Carlos - Aracil Rafael. Fundamentos de robótica. McGraw-Hill, España, 1997.
- [5] Restrepo Carlos – Granada Mauricio. Algoritmos genéticos aplicados al diseño óptimo de mallas de tierra de subestaciones, Energía y Computación, Volumen XII, edición 23, 2005.
- [6] González Fabio Alfonso – Barrero Pérez Jaime. Algoritmos genéticos aplicados al planeamiento de trayectorias de robots móviles. Modelado y simulación , Energía y Computación, Volumen XIII, edición 24, 2005.
- [7] Ashlock Daniel. Evolutionary computation for modeling and optimization, Springer, Primera Edicion, 2006, United States of America. ISBN 0-387-22196-4
- [8] Menon Anil Daniel. Frontiers of evolutionary computation, Kluwer Academic Publisher, Primera Edicion, 2004, United States of America. ISBN 1-4020-7782-3.
- [9] Spall James C. Introduction to stochastic search and optimization, estimation, simulation and control, Wiley-Interscience, Primera Edicion, 2003, United States of America.