



Algoritmo para Resolver la Cinemática Directa de un Robot tipo Ápodo con Articulaciones Rotacionales

Martínez Gómez Carlos Alberto [✉] (1), Vargas Soto José Emilio (1), Pérez Soto Gerardo Israel (1), Ramírez Granados Luisa (1), Sandoval Castro Xóchitl Yamile (2),

Universidad Autónoma de Querétaro, Campus Cerro de las Campanas (1)
Centro de Investigación en Ciencia Aplicada y Tecnología Avanzada, Unidad Querétaro (2)
[✉]carlos.martho@hotmail.com

Resumen

Los robots ápodos poseen una estructura morfológica y principios de locomoción inspirados en las serpientes biológicas. En años recientes han originado líneas de investigación que buscan aprovechar sus cualidades en el desplazamiento a través de superficies irregulares y con espacios reducidos con la finalidad de emplearlos en diversas tareas como la localización de personas atrapadas en sismos, caracterización del entorno, inspección de ductos y plantas de generación de energía. El presente trabajo propone un algoritmo para resolver la cinemática directa de un robot móvil ápodo de múltiples grados de libertad con desplazamiento unidireccional, empleando propiedades matemáticas, como las relaciones de recurrencia y funciones continuas a trazos, presentando como resultados un modelo matemático que generaliza la solución para n articulaciones, su implementación en código C++ y la simulación del movimiento del robot.

Palabras clave: Robot, Algoritmo, Locomoción, Cinemática, Modelo Matemático, implementación.

Abstract

Snake robots have a morphological structure and locomotion principles inspired in biological snakes. In recent years, they have originated research lines that search for take advantage of their qualities for displace through irregular surfaces and narrow spaces having a purpose to use them in many tasks as localization of people trapped in earthquakes, characterization of the environment, duct inspection and safety of power generation plants. This work proposes an algorithm to solve the problem of direct kinematics problem of a unidirectional displacement multiple degrees of freedom snake robot, using mathematical properties like recurrence relations and piecewise functions, presenting as results a mathematical model that generalizes the solution for n -articulations, its implementation in C++ code and movement simulation of robot.

Keywords: Robot, Algorithm, Locomotion, Kinematics, Mathematical Model, Implementation.

1. Introducción

La robótica de servicios se enfoca en desarrollar robots que ayuden al ser humano en labores no necesariamente industriales, la Federación Internacional De Robótica (IFR) Define a un robot de servicios como “Robot que opera de forma parcial o totalmente autónoma al servicio del bienestar de los seres humanos y de equipamientos, excluyendo operaciones manufactureras” [1].



Este tipo de robots pueden ser fijos o móviles. En cuanto a los fijos, han sido empleados en labores que requieren precisión como asistentes en cirugías, atención médica, diagnóstico de enfermedades, apoyo a personas con capacidades diferentes, asistente personal, entre muchas otras.

Sin embargo, la mayoría de los robots de servicios son móviles, su capacidad para desplazarse en el espacio facilita la realización de tareas, sobre todo en áreas donde el tránsito del ser humano es complicado ya sea por las condiciones del terreno, o es en extremo riesgoso por otros factores implicados (radioactividad, concentración de gases tóxicos, carencia de oxígeno).

La realización de estas tareas por parte de los robots de servicios, ha traído como beneficio la continua generación de conocimiento, caracterización del entorno en el que el ser humano no podría acceder, así como reducción del riesgo a que un ser humano normalmente se expondría de realizar dichas tareas. Los robots móviles a su vez pueden clasificarse por su tipo de locomoción. Los cuáles pueden ser:

- Con Ruedas.
- Con cintas de deslizamiento.
- Robots Caminantes.
- Robots Ápodos o Modulares
- Otros.

El diseño de algoritmos de locomoción de robots bioinspirados constituye una línea de investigación en robótica. La selección natural, a través de los procesos evolutivos, ha mejorado el proceso de locomoción de las especies animales a través de millones de años [2]. Imitar dichos patrones morfológicos y de desplazamiento, trae por ende como resultado, eficiencia energética, estabilidad y adaptación al entorno.

En el caso de los robots ápodos, estos poseen un patrón de movimiento empleado por reptiles y algunos anfibios, el cual ya ha sido caracterizado matemáticamente, a través de la ecuación serpentinoide [3], El objetivo de la investigación en ese entonces, fue caracterizar el movimiento de las serpientes a través de un modelo matemático.

Como resultado de estas investigaciones previas, Hirose descubrió que las curvas que hacen las serpientes al moverse también seguían una cierta ley de la naturaleza, la cual pudo generalizar mediante una ecuación matemática. Esto dio lugar a su establecimiento de una fórmula, que llamó "Curva Serpenoide". La cual, funciona como un generador senoidal, y puede caracterizar dicho patrón de locomoción, el cual ha sido aplicado a distintos robots con morfologías bio inspiradas en estos animales, con aplicaciones diversas, como lo son: Exploración, Inspección en ductos y tanques industriales, búsqueda y rescate.

La "Curva Serpenoide de Hirose, se expresa de la siguiente forma:

$$K(s) = \frac{-2\pi k}{l} \alpha \sin\left(\frac{2\pi K}{l} S\right) \quad (1)$$

Donde:

l: longitud de la curva (mm).

s: distancia a lo largo de la curva (mm).

k: número de ondulaciones, $K > 0$

α : Angulo de serpenteo (radianes).

Mediante esta función, podemos generar senoidales conocidas como serpentinoides, lo que equivale a determinar las trayectorias senoidales que seguirá el robot dependiendo de la distancia y del tiempo.

Una vez establecidos dichos resultados, Hirose se enfocó en construir el primer robot bioinspirado en serpientes. El cual fue desarrollado y puesto en funcionamiento en el Instituto de Tecnología De Tokio en 1976 [3].

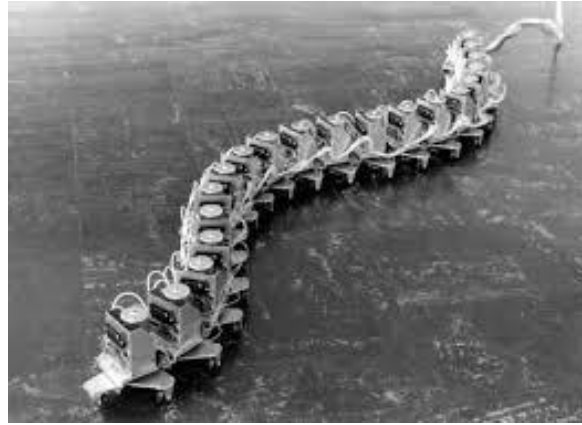


Figura 1: ACM III's, primer robot ápedo desarrollado por Shigeo Hirose [3]

El poder caracterizar el movimiento de una serpiente mediante ecuaciones matemáticas, también permitió modificar esta ecuación de onda para modelar los diversos patrones de locomoción de una serpiente biológica, estos pueden categorizarse en 4 tipos, los cuales son:

- Serpentinoide
- Rectilínea
- Concertina
- Ondulación Lateral

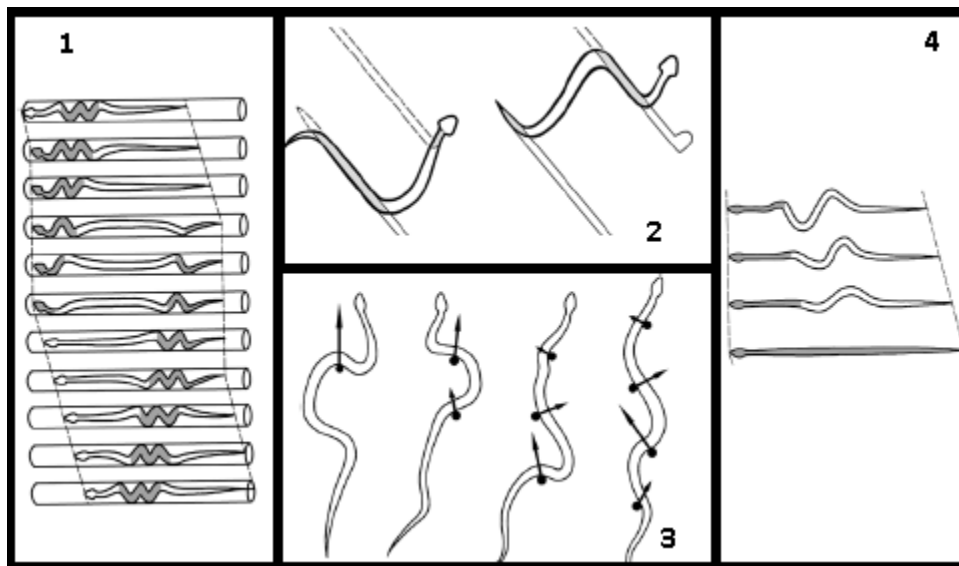


Figura 2: Patrones de locomoción serpentinoide. 1: Concertina, 2: Ondulación lateral, 3: Serpentinoide, 4: Rectilínea [4].



Los robots ápodos, poseen propiedades interesantes, ya que el hecho de no contar con extremidades, no presenta impedimento para que estas se desplacen por cualquier terreno, así sea irregular o reducido. Por lo cual, nos permiten proponerlos como solución al problema de cómo moverse a través de espacios sumamente reducidos, y con superficies de desplazamiento irregulares.

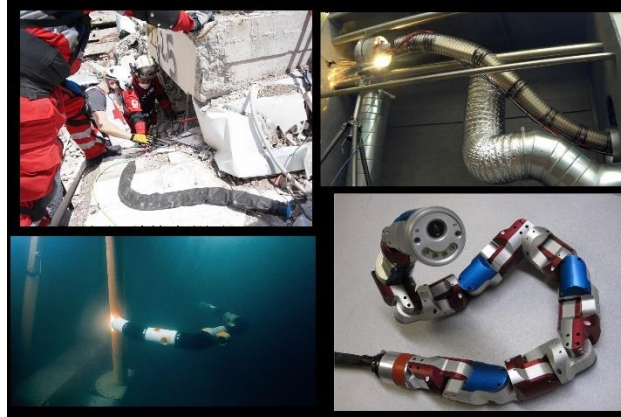


Figura 3: Robots Ápodos y sus aplicaciones en tareas diversas, de izquierda a derecha, arriba abajo: Robot serpiente trabajando en labores de rescate [5], Robot inspector de ductos [6], Robot Snake de inspección subacuática [7], Robot de Desplazamiento en 3 Dimensiones [5].

El problema de la locomoción de un robot de estas características, radica en la complejidad que representa el caracterizar el movimiento de una serpiente a través de un modelo matemático, que depende de la longitud en las articulaciones, estos pueden ser continuos o discretos.

En el modelo continuo, se hace la suposición de que la distancia entre cada una de las articulaciones tiende a 0, haciendo la analogía en una serpiente biológica la cual tiene una distancia entre sus vertebras (análogas a los eslabones), ya que es flexible, por lo cual, basta con la ecuación (1), para describir el comportamiento en el espacio del robot. Por contraparte, una serpiente robótica tiene articulaciones rígidas que preservan una métrica en el espacio, por lo cual, para modelar su movimiento, es necesario plantear un modelo discreto, Sandoval Castro [8], propone el siguiente modelo matemático:

$$\alpha_i(\varphi) = \alpha \cos\left(\varphi + \frac{2\pi k}{M}(i - 1)\right) \quad (2)$$

Donde:

$\alpha_i(\varphi)$: Ángulo de la i-ésima articulación.

α : Amplitud, delimita la altura, tomada por el robot (radianes).

φ : Desfasamiento, o ángulo inicial (radianes).

k: Número de ondulaciones

M: Número total de articulaciones

i: Número de articulación, para $i = 1, 2, 3 \dots M$.

El cual funciona como un generador sinusoidal, donde la i-ésima onda, describe la ubicación en el espacio de la i-ésima articulación. para localizar cada uno de estos puntos, debemos solucionar el problema de cinemática directa del robot, el cual se plantea de la siguiente forma:

Sea.

$$Q = \{q_1, q_2, q_3, q_4 \dots q_M\} , q_M \in \mathbb{R} \quad (3)$$



El conjunto de vectores de variables articulares q_n , que expresa la dirección en el espacio de las M articulaciones, y sea:

$$P = (x, y) \in \mathbb{R}^2 \quad (4)$$

La posición final del robot, considerando que estamos trabajando en el modelo unidireccional.

Enunciado: Dada una colección de elementos de Q, El problema de cinemática directa consiste en encontrar una función continua:

$$f: Q \subset \mathbb{R}^M \rightarrow P \subset \mathbb{R}^2 \quad (5)$$

En el presente trabajo, se unificará el modelo matemático de la ecuación (2), con una f de la ecuación (5) a encontrar para tener un modelo más compacto, el cual se implementará en código C++. Una vez realizada esta parte, se describirá un algoritmo para simular el movimiento del robot serpiente en software, basado en el modelo propuesto.

2. Obtención del Modelo Matemático.

Para desarrollar el modelo matemático se siguió la siguiente metodología.

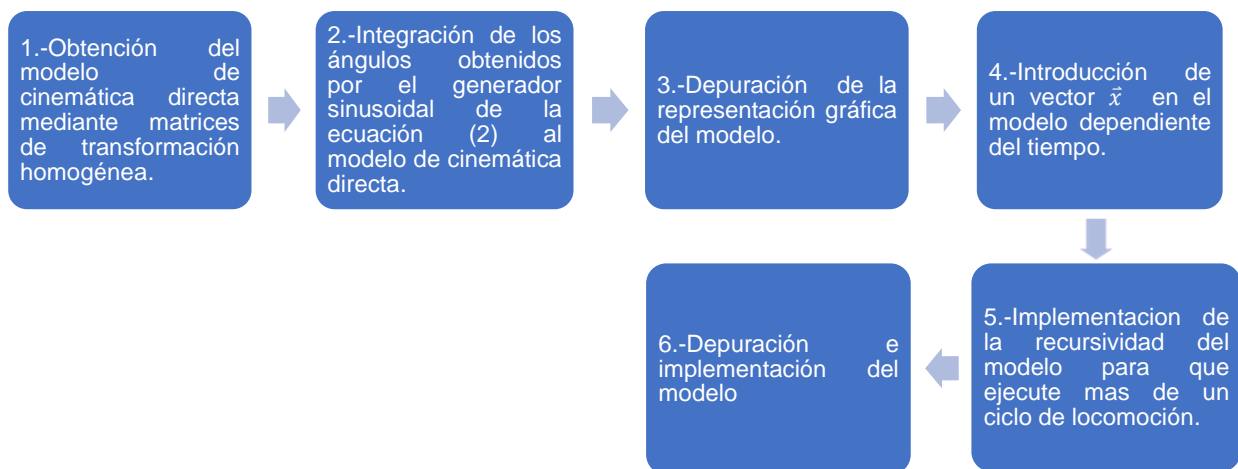


Figura 4: Metodología a desarrollar.

2.1 Obtención del modelo de cinemática directa del robot:

El modelo de cinemática directa se obtuvo tratando la estructura morfológica del robot como una cadena cinemática abierta con la primera articulación con la base fija, con el objetivo de poder utilizar la convención Denavit-Hartenberg.

Usando la notación D-H, se buscó la matriz de transformación homogénea para n articulaciones.

Para $M=1$:



$${}^0H_1 = \begin{bmatrix} \cos(q_1) & -\text{sen}(q_1) & 0 & l * \cos(q_1) \\ \text{sen}(q_1) & \cos(q_1) & 0 & l * \text{sen}(q_1) \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \quad (6)$$

Donde:

q_i : ángulo de la articulación i

l : longitud de los eslabones

Para $M=2$:

$${}^0H_2 = \begin{bmatrix} \cos(q_1+q_2) & -\text{sen}(q_1+q_2) & 0 & l[\cos(q_1) + \cos(q_1+q_2)] \\ \text{sen}(q_1+q_2) & \cos(q_1+q_2) & 0 & l[\text{sen}(q_1) + \text{sen}(q_1+q_2)] \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (7)$$

Para $M=3$:

$${}^0H_3 = \begin{bmatrix} \cos(q_1+q_2+q_3) & -\text{sen}(q_1+q_2+q_3) & 0 & l[\cos(q_1) + \cos(q_1+q_2) + \cos(q_1+q_2+q_3)] \\ \text{sen}(q_1+q_2+q_3) & \cos(q_1+q_2+q_3) & 0 & l[\text{sen}(q_1) + \text{sen}(q_1+q_2) + \text{sen}(q_1+q_2+q_3)] \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (8)$$

Podemos notar que se forma un patrón o relación de recurrencia, el cual nos permite generalizar el modelo cinemático para M articulaciones:

$${}^0H_M = \begin{bmatrix} \sum_{i=1}^M \cos(q_i) & -\sum_{i=1}^M \text{sen}(q_i) & 0 & l \sum_{i=1}^M \cos(\sum_{i=1}^M q_i) \\ \sum_{i=1}^M \text{sen}(q_i) & \sum_{i=1}^M \cos(q_i) & 0 & l \sum_{i=1}^M \text{sen}(\sum_{i=1}^M q_i) \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (9)$$

Por tanto, la función que nos indica la posición de cada una de las articulaciones, está dada por la siguiente expresión vectorial:

$$\begin{bmatrix} x_i \\ y_i \end{bmatrix} = \begin{bmatrix} l \sum_{i=1}^M \cos(\sum_{i=1}^M q_i) \\ l \sum_{i=1}^M \text{sen}(\sum_{i=1}^M q_i) \end{bmatrix} \quad (10)$$

2.1 Modelo cinemático obtenido

El modelo descrito en la ecuación (2) genera ángulos absolutos, denotados por q_i , es decir, el ángulo de la articulación i , con respecto al eje x , mientras que el modelo cinemático obtenido mediante matrices de transformación homogénea bajo la convención Denavit-Hartenberg, trabaja con ángulos relativos, es decir, el ángulo formado por la articulación i , con respecto a la articulación $i-1$. Por lo cual el siguiente paso a realizar es transformar los ángulos absolutos en relativos y poder integrarlos en nuestra solución de la cinemática directa.

$$q_i = \alpha_i - \alpha_{i-1} \quad (11)$$

Por lo cual, sustituyendo (11) en (10), queda de la siguiente manera:

$$\begin{bmatrix} x_i \\ y_i \end{bmatrix} = \begin{bmatrix} l \sum_{i=1}^M \cos(\sum_{i=1}^M (\alpha_i - \alpha_{i-1})) \\ l \sum_{i=1}^M \text{sen}(\sum_{i=1}^M (\alpha_i - \alpha_{i-1})) \end{bmatrix} \quad (12)$$

Graficamos esta función con los siguientes parámetros en la ecuación (2).



$$\begin{aligned} \alpha &= 60^\circ \\ \Phi &= -90^\circ \\ K &= 2 \\ M &= 10 \end{aligned}$$

Se aprecia en la figura (5), que modelo gráfico obtenido, es una aproximación discreta de una función senoidal, pero que, por la naturaleza de las estas funciones, pasa por lugares menores a cero, por tanto, será necesario hacer una traslación en el plano, con la finalidad de que el gráfico se proyecte por sobre el eje X, el cual será nuestro sistema de referencia.

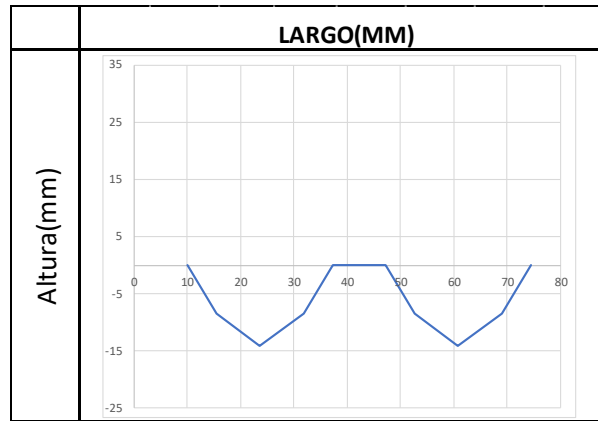


Figura 5: Grafico de posición inicial de un robot ápedo de 10 G.D.L

La traslación propuesta será la siguiente:

$$y_i = y_i - \min\{y_i\} \tag{13}$$

Sustituyendo (13) en (12):

$$\begin{bmatrix} x_i \\ y_i \end{bmatrix} = \begin{bmatrix} l \sum_{i=1}^M \cos\left(\sum_{i=1}^M (\alpha_i - \alpha_{i-1})\right) \\ l \sum_{i=1}^M \sin\left(\sum_{i=1}^M (\alpha_i - \alpha_{i-1})\right) - \min\left\{l \sum_{i=1}^M \sin\left(\sum_{i=1}^M (\alpha_i - \alpha_{i-1})\right)\right\} \end{bmatrix} \tag{14}$$

Aplicando esta traslación, la curva Serpenoide siempre estará sobre nuestra superficie de trabajo, como muestra la gráfica (6).

Basta hacer variar a ϕ , de la ecuación (2), para que el robot, realice un serpenteo, sin embargo, este se realizará sobre su misma posición, para completar el modelo, es necesario implementar una ecuación de paso.

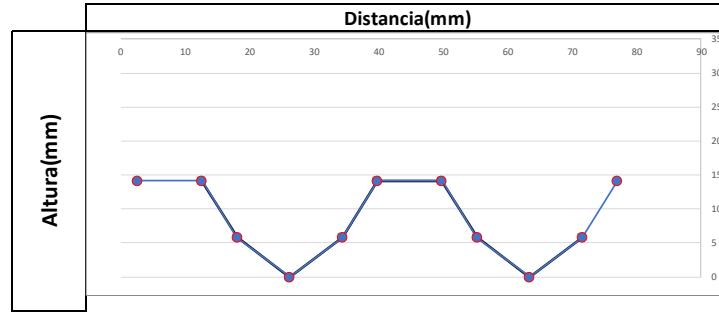


Figura 6: Gráfico de posición inicial de un robot ápedo de 10 GDL, Usando la ecuación (14).

Una vez implementadas las ecuaciones de Hirose en nuestro modelo, nos dispondremos a generar un ciclo de locomoción, el cual dará movimiento a través del eje x al robot, para lo cual necesitaremos hacer variar el parámetro φ de la ecuación (2), con la finalidad de que los ángulos que entrarán a la solución de la cinemática directa y nos darán una locomoción deseada.

Tomando como referencia un patrón de locomoción similar propuesto en[9], el presente modelo matemático propone variar φ , de la siguiente manera:

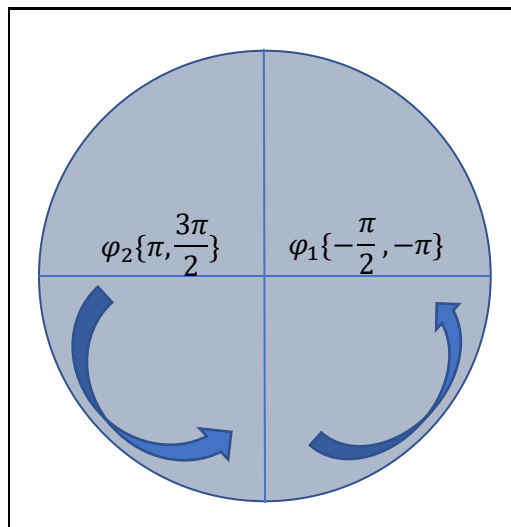


Figura 7: Secuencia de valores que tomará la variable φ

La variable φ , puede representarse como una función continua a trazos, que, en la primera parte del algoritmo, va de un ángulo a otro, y en la segunda recorre otra sección del rango de trabajo.

Definamos al vector \vec{x} , como el vector que expresa el avance del robot con respecto al eje x, en un instante de tiempo t, y dependiendo de un parámetro Δ_x , que mide el avance del robot en un ciclo de trabajo, tomando en cuenta las variables de diseño del robot que son amplitud(α), longitud(l), número de ondulaciones(k).

$$\vec{x} = \Delta_x t \tag{15}$$

El avance sobre el eje x, está determinado por los parámetros del robot, mediante un análisis de como influyen cada uno de estos, proponemos esta constante haciendo las siguientes reflexiones de cómo se da la locomoción de las serpientes:



1. A mayor amplitud (α), el organismo ápodico tiende a generar una zancada mas larga, por tanto, es proporcional al avance.
2. A mayor longitud (l), es esperable que un organismo ápodico tenga un ciclo de locomoción con mayor avance.
3. A mayor número de ondulaciones (K), el organismo, bajara más articulaciones (vertebras al suelo, y por tanto sus zancadas serán muchas, pero dentro de un mismo ciclo de trabajo, por tanto, esta variable es inversamente proporcional.

Por tanto, el parámetro Δ_x , estará determinado por el siguiente cociente:

$$\Delta_x = \frac{l\alpha}{k} \quad (16)$$

Como un ciclo de trabajo estará comprendido por el rango de φ_i , el cual es π radianes, dividiremos el ciclo de trabajo t , en $\pi/180$ instantes, de los cuales, el modelo trabajara con φ_1 y la segunda mitad, con φ_2 , entonces, definiremos las siguientes funciones continuas a trazos:

$$\varphi(C) = \begin{cases} \frac{-\pi(C+90)}{180}, & C < 90 \\ \frac{\pi(C+90)}{180}, & C \geq 90 \end{cases} \quad (17)$$

Donde $C \in \mathbb{Z}^+ 0,1,2 \dots 180$.

De igual forma, t convergerá a 1, en función del crecimiento de c , por tanto, podemos hacer el siguiente cambio de variable en la ecuación (15).

$$\vec{x} = \frac{c}{180} \Delta_x \quad (18)$$

Por tanto, nuestro modelo matemático quedará comprendido de la siguiente forma:

$$\begin{bmatrix} x_i \\ y_i \end{bmatrix} = \begin{bmatrix} l \sum_{i=1}^M \cos\left(\sum_{i=1}^M (\alpha_i - \alpha_{i-1})\right) + \vec{x} \\ l \sum_{i=1}^M \sin\left(\sum_{i=1}^M (\alpha_i - \alpha_{i-1})\right) - \min\left\{ \sum_{i=1}^M \sin\left(\sum_{i=1}^M (\alpha_i - \alpha_{i-1})\right) \right\} \end{bmatrix} \quad (19)$$

Donde los ángulos α_i , se obtienen mediante la ecuación (2).

2.2 Recursividad

Si se desea implementar este algoritmo para más de un ciclo de trabajo, entonces hay que generar una recursividad en el modelo, es decir, que cuando el instante $t=1$, en el cual un ciclo de trabajo ha finalizado, el algoritmo se reinicie tomando como nueva posición inicial, la obtenida en dicho instante t .

Por tanto:

Si $C > 180$, entonces: $\tau = t + 1$

Donde τ , es el ciclo de trabajo en el que se está ejecutando el algoritmo

La recursividad tendrá final cuando $\tau = T$, donde T es el número total de ciclos de trabajo.



2.3 Implementación

Se presenta el diagrama de flujo de la figura 8. Cuyos pasos conducen a la implementación del algoritmo de locomoción, basado en el modelo matemático desarrollado en el presente trabajo.

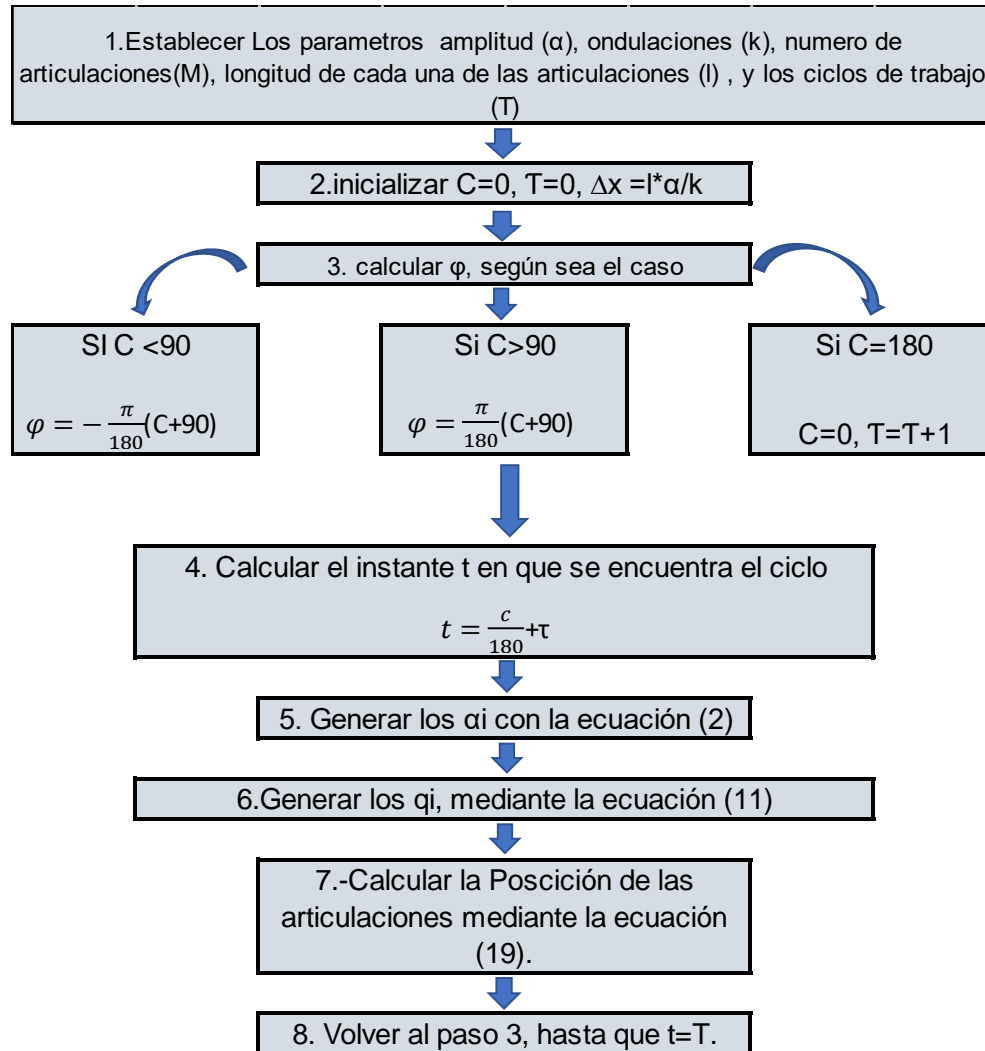


Figura 8. Diagrama de flujo para la implementación del algoritmo de locomoción.

3. Resultados

Para valorar la validez del modelo, se presenta en la tabla 1, una tabla de comparación con otros dos modelos semejantes para robots ápodos, haciendo énfasis en el rango de valores abarcados por la ecuación (17), y la metodología para posicionar al robot en el espacio, ya que estos son, dada la investigación previa, los que influyen directamente en el costo de cómputo al ejecutarlo en un algoritmo.

Trabajo	Rango de $\varphi(c)$ (Grados)	Método de Posicionamiento del robot móvil en el espacio
---------	--------------------------------	---



Algoritmo Presentado	180 °	Vectorial.
Garzón Oviedo [10]	No especificado	Estructura Virtual.
González Gómez [9]	360°	Estructura Virtual.

Tabla 1: Comparación de metodologías para resolver la cinemática directa de robots ápodos 1R.

Se desarrolló una animación basado en los datos obtenidos en hoja de cálculo de Excel, y se implementó además el algoritmo en código C++, para validar los cálculos obtenidos. los parámetros del modelo son los que se implementaron para graficar la ecuación (12). Obteniéndose una constante de la ecuación de paso Δ_x de 5.2359 milímetros por ciclo de trabajo. Para realizar estos cálculos, se utilizó un procesador Intel Celeron 1.60 GHz, con memoria instalada de 4 GB. El tiempo de ejecución del algoritmo es de 5.8172 segundos por ciclo de locomoción. El desarrollo de la locomoción de un algoritmo para 6 ciclos de trabajo, se muestra en la figura 9.

El patrón de locomoción a detalle, durante un ciclo de trabajo se muestra en la figura 10. La gráfica se obtuvo de los datos obtenidos mediante hoja de cálculo Excel, la columna izquierda muestra la evolución del robot cuando la variable C de la ecuación (16), cuando esta es menor a $\pi/2$, la columna derecha, muestra la evolución del movimiento cuando la variable C, es mayor a $\pi/2$, los incrementos de esta gráfica se muestran cada $\Delta\phi = \pi/12$.

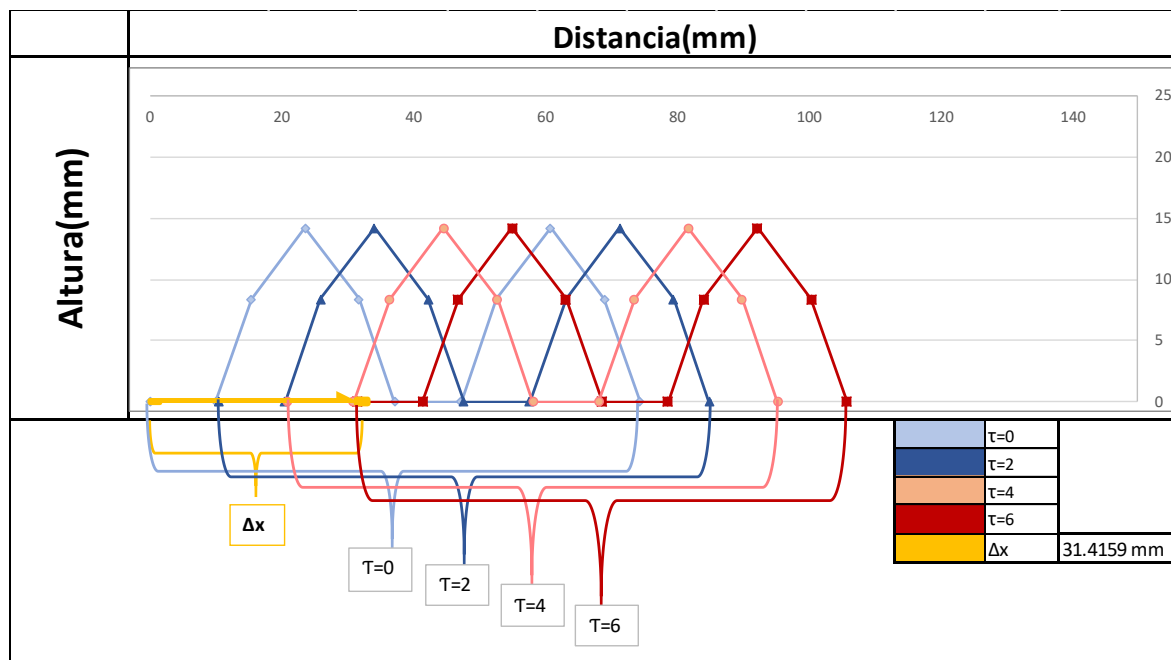


Figura 9: Locomoción programada para 6 ciclos de trabajo a lo largo de una línea recta.

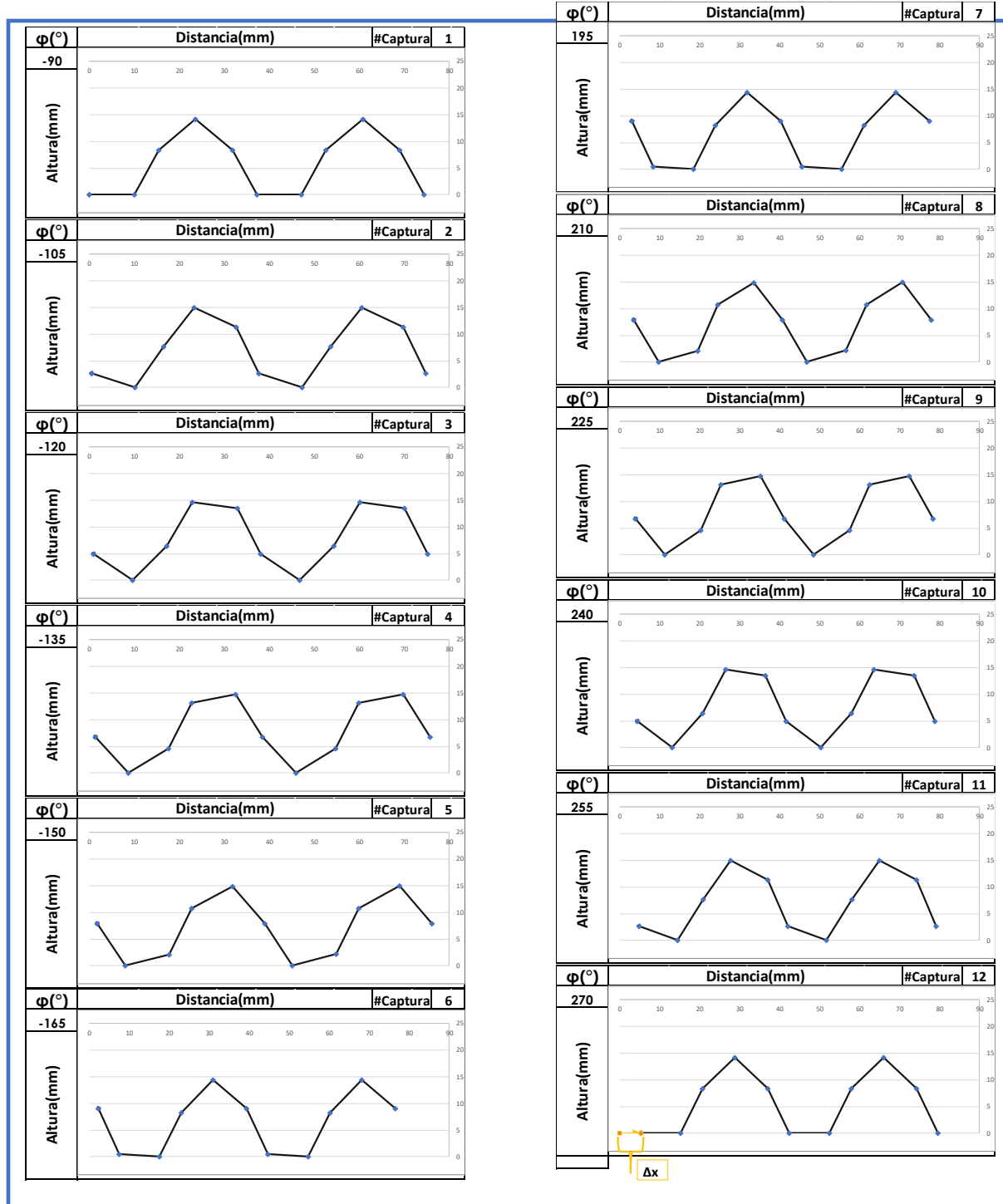


Figura 10: Simulación de la locomoción del robot en un ciclo.

3.1 Análisis del error:



Debido a que el algoritmo no toma el valor exacto de π , es de esperar un error de aproximación, y otro ocasionado por la discretización de ondas sinusoidales, como se presenta en [8]. La figura 11, muestra el valor esperado mediante cálculo, de la posición de la última articulación del robot, con respecto al obtenido en el algoritmo implementado.

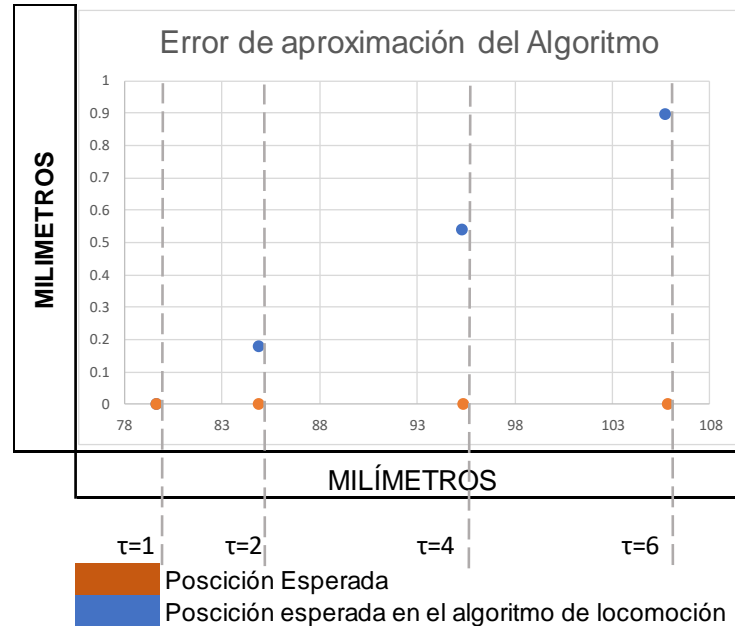


Figura 11: Comparativa entre valor esperado y el obtenido por el algoritmo.

Después del primer ciclo de trabajo, comienza a generarse un error, en forma de desfaseamiento, horizontal y vertical, estos son lineales respecto al ciclo de trabajo τ , y mediante ecuaciones de la recta, se presentan de la siguiente manera:

$$\begin{bmatrix} X_{alg} \\ Y_{alg} \end{bmatrix} = \begin{bmatrix} X_{esp} - 0.0291\tau \\ Y_{esp} + 0.01791\tau \end{bmatrix} \quad (20)$$

Donde:

X_{alg} = Ubicación en X, obtenida por el algoritmo de locomoción, del último eslabón

Y_{alg} = Ubicación en Y, obtenida por el algoritmo de locomoción, del último eslabón

X_{esp} = Ubicación esperada en X, del último eslabón del robot.

y_{esp} = Ubicación esperada en Y, del último eslabón del robot.

3.2 Tabla de resumen de variables y parámetros.

Símbolo	Tipo.	Definición.	Unidades	Dominio.
α_i	Variable	Ángulo de la articulación i, con respecto al eje x.	Radianes.	$\{0; \frac{2\pi}{3}\}$
α	Parámetro	Amplitud de onda	Radianes.	$\{0; \frac{2\pi}{3}\}$
$\Phi(c)$	Función	proporciona el ángulo Φ , para cada instante t	Radianes.	$\{\frac{\pi}{2}; -\pi, \pi; \frac{3\pi}{2}\}$
K	Parámetro	Numero de ondulaciones.	Escalar.	$K > 2$
M	Parámetro	Número de articulaciones.	Escalar.	$M \in \mathbb{Z}^+, M > 2$.



L	Parámetro	Longitud del Eslabón.	Milímetros	$l \in \mathbb{Q}^+$
q_i	Variable	Angulo de la articulación i, con respecto a la articulación i-1.	Radianes.	$\{-\pi, \pi\}$
\vec{x}	Vector	Vector que depende del instante t y de la constante de paso.	Milímetros	$\{0: \frac{l\alpha}{k}\}$
Δ_x	Parámetro	Avance del robot en un ciclo de trabajo.	Milímetros	$\frac{l\alpha}{k}$
t(c)	Función	Instante t en que se encuentra el ciclo de locomoción.		$\{0: T\}$
C	Variable	Contador	Escalar	$\{0: 180\}$
τ	Constante	Ciclo de trabajo.	Escalar	$\tau \in \mathbb{Z}^+, \tau > 1.$
T	Constante	Ciclos de trabajo a reproducir por el algoritmo.	Escalar	$T \in \mathbb{Z}^+, T > 1.$

Tabla 2. Resumen de Variables y Parámetros Implementados en el Modelo

4. Conclusiones

Durante el trabajo presentado, se desarrolló conocimiento matemático, para entender cómo se comporta este tipo de robots, lo cual permitió su validación mediante un diagrama de flujo y su implementación en código C++, aportando un modelo propuesto para describir la locomoción del robot, proponiendo una secuencia para el generador sinusoidal que disminuye el rango de iteraciones comparado con otros métodos presentados.

Actualmente se está desarrollando bajo esta misma metodología un modelo y un algoritmo de locomoción que logre desplazar un robot ápedo en el espacio, en una superficie con inclinación, por lo que este trabajo sirvió como parteaguas para comprender e implementar las curvas serpenoides. Competencias útiles para trabajar con el modelo en el espacio que es, por definición más complejo de obtener y validar.

Referencias

- [1] International Federation of Robotics. (2016). Service Robotics. Sitio Web: www.ifr.org/service-robots [Último acceso: febrero de 2019].
- [2] Darwin, C. (2009). El origen de las especies por medio de la selección natural. Editorial CSIC-CSIC Press. 1ª edición, Madrid, 2009.
- [3] Hirose, S., & Yamada, H. Snake-like robots: "Machine design of biologically inspired robots". IEEE Robotics and Automation Magazine, 16(1), 88–98. 2019.
- [4] Lilywhite, H., (2014). How Snakes Work: Structure, Function and Behavior of the World's Snakes. OXFORD. University Press, Nueva York, 2014.
- [5] Esta robo-serpiente buscó sobrevivientes tras el sismo de la CDMX. (04 de octubre de 2017). *El Financiero*. p.1
- [6] Atherton, K., (15 de diciembre de 2015). Snake Robot Slithers Through Pipes, Laser-Welds From Within. *Popular Science*. Recuperado de: <https://www.popsci.com/laser-wielding-snake-robot-wants-to-repair-pipes>.
- [7] SINTEF. (2016) A giant subsea snake robot. Sitio Web: <https://www.sintef.no> [Último acceso : Marzo de 2019].
- [8] Sandoval, X. Castillo E., "Desarrollo de un Robot Gusano Capaz de Desplazarse en Terreno Natural". Robótica y Mecatrónica, Asociación Mexicana de Mecatrónica, 1ª edición, 2018
- [9] González, J. (2008). *Robótica Modular y Locomoción: Aplicación a Robots Ápodos* (Tesis Doctoral). Universidad Autónoma de Madrid, Madrid, España
- [10] Garzón, M. (2009). *Estrategias Bio-Inspiradas para Locomoción de Robots Ápodos*. (Tesis De Maestría). Universidad Politécnica de Madrid, Madrid, España.