

Cálculo de trayectorias para sistemas sujetos a restricciones no holónomas

David Rodríguez, Gustavo Arechavaleta
Robótica y Manufactura Avanzada-CINVESTAV, Saltillo, México

Abstract—En este artículo proponemos una estrategia numérica simple, eficiente y válida para encontrar las trayectorias de sistemas sin deriva con restricciones diferenciales no integrables. Específicamente, consideramos el problema de minimizar la norma L_2 del control. Primero convertimos este problema de control óptimo a una formulación de optimización numérica no lineal con restricciones. Esta formulación nos permite encontrar las trayectorias utilizando la Programación Cuadrática Secuencial (PCS). Comparamos nuestra estrategia con el algoritmo desarrollado por Fernandes *et al.* [1], en términos de convergencia y tiempo de cálculo, utilizando varios modelos cinemáticos de robots móviles con ruedas y remolques.

Palabras clave: robótica móvil, sistemas no holónomos, control óptimo, optimización numérica.

I. INTRODUCCIÓN

En robótica, la planificación de movimientos ha contribuido al desarrollo de robots autónomos. Inicialmente, surgieron algoritmos eficientes para la generación de rutas libres de colisiones. Estos algoritmos simplificaban el problema *completo* de movimiento, que involucra cinemática y dinámica, a un problema puramente geométrico. Sin embargo, varios robots móviles con ruedas, manipuladores espaciales, entre otros, están sujetos a características cinemáticas y dinámicas restrictivas que llamamos no holónomas (i.e. restricciones diferenciales no integrables). El estudio de esta clase de sistemas ha generado interés en la comunidad de matemáticas (e.g., [2]), teoría de control (e.g., [3]) y robótica (e.g., [4]).

Algunos métodos para generar trayectorias válidas para esta clase de sistemas están basados en la teoría del control óptimo. Aunque la solución óptima válida para cualquier sistema no holónomo es desconocida, existen algunas excepciones para caracterizar los caminos más cortos entre cualquier par de configuraciones en $\mathbb{R}^2 \times \mathbb{S}^1$ cuando la curvatura está acotada [5], [6]. En [7] y [8] utilizaron el Principio Máximo de Pontryagin (PMP) [9] para resolver el mismo problema. En [10] y [11] utilizaron el PMP para resolver la síntesis completa de los caminos más cortos para el mismo caso particular. También han sido reportadas algunas variantes del problema siguiendo el mismo enfoque [12], [13] [14], [15] y [16].

Debido a la complejidad que representa encontrar la solución general óptima para desplazar cualquier sistema no holónomo, la alternativa es la utilización de métodos numéricos. Entre las estrategias numéricas que han sido propuestas en la literatura se encuentra el algoritmo de Fernandes *et al.* [1]. La idea subyacente del método se encuentra en la representación del espacio de control mediante una serie de

Fourier truncada. De esta forma, el problema de naturaleza variacional lo transforman en un problema de optimización no lineal. Siguiendo este enfoque, han sido reportadas algunas variantes y extensiones [19], [20], [21], [22], [23]. Todos estos métodos dependen de una formulación adecuada de la estrategia de optimización y de su sensibilidad numérica para lograr un tiempo de convergencia mínimo. En consecuencia, el desempeño de los métodos reposa en la adquisición y manejo de sofisticados paquetes informáticos existentes en el mercado.

En este trabajo proponemos una estrategia numérica simple y eficiente basada en la PCS. Esta estrategia representa la contribución principal de este trabajo. El algoritmo que proponemos permite generar trayectorias para sistemas sin deriva con restricciones no holónomas. En particular, calculamos las trayectorias que minimizan la norma L_2 del control.

En la sección II exploramos la estructura del problema mediante herramientas analíticas. Posteriormente en la sección III revisamos la formulación numérica del problema y describimos la estrategia que proponemos usando PCS. En la sección IV mostramos el desempeño de la estrategia utilizando varios sistemas con diferente grado de complejidad y comparamos el método con el algoritmo reportado en [1]. En la sección V presentamos algunos comentarios finales.

II. FORMULACIÓN GENERAL VÍA EL CONTROL ÓPTIMO

Consideramos un sistema de ecuaciones diferenciales de la forma

$$\dot{q} = f(q, u) \quad (1)$$

donde $q \in \mathbb{R}^n$, que habita en la variedad M , representa el estado del sistema y $u \in \mathbb{R}^m$ el control. Entonces, para cualesquiera dos estados $q_0, q_f \in \mathbb{R}^n$, deseamos caracterizar, entre todas las leyes de control que llevan al sistema de q_0 a q_f , una (si existe) que minimice la funcional:

$$J = \int_0^T \mathcal{L}(q(\tau), u, u(\tau)) d\tau \quad (2)$$

donde $\mathcal{L}(q, u)$ es una función continua y diferenciable con respecto a sus parámetros. En particular, nos interesamos en sistemas de control de la forma

$$\dot{q} = B(q)u, \quad B(q) = g_1(q), \dots, g_m(q) \quad (3)$$

donde $B(q) \in \mathbb{R}^{n \times m}$ es una distribución suave de dimensión m (i.e. en cada punto $q \in M$ la familia de vectores

$g_1(q), \dots, g_m(q)$ genera un subespacio lineal $\Delta(q)$ del espacio tangente T_qM . Cada g representa una transformación suave que asigna a cada q un vector tangente $g(q) \in T_qM$.

II-A. Minimización de la norma L_2 del control

El criterio que deseamos minimizar es

$$J = \frac{1}{2} \int_0^T \langle u(\tau), u(\tau) \rangle d\tau \quad (4)$$

sujeto a las restricciones dadas por (3). Entonces, para cualquier par de configuraciones $q_0, q_f \in R^n$, deseamos encontrar el conjunto de entradas $u(\tau) \in R^m, \tau \in [0, T]$, que minimicen a J y lleven al sistema de q_0 a q_f . Entonces, la función Hamiltoniana se define como

$$H(q, u, \psi) = \frac{1}{2} u^t u + \psi^t \sum_{i=1}^m g_i(q) u_i \quad (5)$$

II-B. Análisis con PMP

Minimizando la función Hamiltoniana obtenemos el control óptimo

$$u_i^* = -\psi^t g_i(q) \quad (6)$$

y la Hamiltoniana óptima es

$$H^*(q, \psi) = -\frac{1}{2} \sum_{i=1}^m (\psi^t g_i(q))^2 \quad (7)$$

El sistema satisface las siguientes ecuaciones

$$\begin{aligned} \dot{q} &= -\sum_{i=1}^m g_i(q) (\psi^t g_i(q)) \\ \dot{\psi} &= \sum_{i=1}^m \frac{\partial g_i}{\partial q} \psi^t (\psi^t g_i(q)) \end{aligned} \quad (8)$$

suponemos que $q(0) = q_0$ y $q(T) = q_f$. La estructura del control óptimo de este problema ha sido estudiada en [24]. Los autores encontraron un resultado interesante al diferenciar la ecuación (6), esto es

$$\dot{u}_i = -\dot{\psi}^t g_i(q) - \psi^t \frac{\partial g_i}{\partial q} \dot{q} \quad (9)$$

Entonces, utilizando la ecuación de la función Hamiltoniana para $\dot{\psi}_j$ dada por

$$\dot{\psi}_j = -\sum_{i=1}^m \psi^t \frac{\partial g_i}{\partial q_j} u_i \quad (10)$$

podemos obtener la siguiente ecuación diferencial

$$\dot{u}_i = \sum_{j=1}^m \psi^t [g_i, g_j] u_j \quad (11)$$

donde $[g_1, g_2]$ es el corchete de Lie de dos campos vectoriales. Este hecho establece que la norma de la entrada óptima es constante para todo τ , esto es

$$\|u(\tau)\|^2 = \|u(0)\|^2 \quad (12)$$

Este resultado arroja información parcial. Por lo tanto, a partir de esta información no podemos encontrar las trayectorias óptimas.

III. MÉTODOS DE OPTIMIZACIÓN NUMÉRICA

Consideramos el sistema dinámico (3) junto con la función de costo (4). Definimos una base ortonormal $\{e_j\}_{j=1}^{\infty}$ para $L_2([0, T])$ y suponemos que la señal de control $u \in L_2([0, T])$ es una función continua C^1 por partes definida en $[0, T]$. De esta forma podemos escribir u en términos de una serie de Fourier

$$u = \sum_{j=1}^{\infty} (\alpha_j e^{i \frac{2j\pi t}{T}} + \beta_j e^{-i \frac{2j\pi t}{T}})$$

Entonces podemos aproximar u truncando la serie hasta N . La nueva ley de control u y la función objetivo J ahora pueden expresarse como

$$u = \sum_{j=1}^N \alpha_j e_j \implies J \simeq \sum_{j=1}^N |\alpha_j|^2$$

donde $\alpha = (\alpha_1, \alpha_2, \dots, \alpha_N) \in \mathbb{R}^N$ debe encontrarse. La configuración $q(T)$ es la solución en el tiempo T aplicando la ley de control u . Entonces, $q(T)$ está en función de α . Por lo tanto, $f(\alpha) = q(T)$. De esta forma, el problema de optimización ahora es el siguiente: dados un tiempo final T y el par (q_0, q_f) , encontrar $\alpha \in \mathbb{R}^N$ tal que la función de costo $J(\alpha)$ sea mínima.

Debido a que $f(\alpha)$ normalmente no es conocida, necesitamos integrar numéricamente el sistema (3). Al finalizar la integración del sistema también podemos obtener la Jacobiana $\frac{\partial f}{\partial \alpha} \in \mathbb{R}^{n \times N}$. Podemos escribir el sistema (3) tal que $\forall(\tau, \alpha) \in [0, T] \times [0, +\infty)$

$$\frac{\partial q}{\partial \tau}(\tau, \alpha) = \sum_{i=1}^m u_i(\tau, \alpha) g_i(q(\tau, \alpha)) \quad (13)$$

Diferenciando (13) obtenemos:

$$\frac{\partial^2 q}{\partial \tau \partial \alpha}(\tau, \alpha) = X(\tau, \alpha) Y(\tau, \alpha) + B(\tau, \alpha) e(\tau, \alpha) \quad (14)$$

donde $Y(\tau, \alpha) = \frac{\partial q}{\partial \alpha}(\tau, \alpha)$ y $e(\tau, \alpha) = \frac{\partial u}{\partial \alpha}(\tau, \alpha)$. $X(\tau, \alpha)$ es una matriz de $n \times n$

$$X(\tau, \alpha) = \sum_{i=1}^m (u_i(\tau, \alpha) \frac{\partial g_i}{\partial q}(q(\tau, \alpha))) \quad (15)$$

y $B(\tau, \alpha)$ es una matriz de $n \times m$. Entonces, el sistema (14) es de hecho el sistema linealizado (3) en $\tau \rightarrow q(\tau, \alpha)$.

Utilizando las ecuaciones anteriores, $f(\alpha)$ y $\frac{\partial f}{\partial \alpha}$ se obtienen evaluando (3) y (14) en $\tau = T$ respectivamente.

III-A. Estrategia de optimización sin restricciones

En Fernandes *et al.* utilizan una variación del método de Newton para llevar al sistema a q_f . Primero añaden un término adicional a la función de costo

$$J(\alpha) = \frac{1}{2} \sum_{j=1}^N \|\alpha_j\|^2 + \gamma \|f(\alpha) - q_f\|^2 \quad (16)$$

donde $q(T) = f(\alpha)$ y γ es un parámetro que el usuario debe manipular durante el proceso de optimización. Para

minimizar la función no lineal $J(\alpha)$, deben calcular los siguientes términos

$$\frac{\partial J}{\partial \alpha} = \alpha + \gamma Y^t(f(\alpha) - q_f) \quad (17)$$

y

$$\frac{\partial^2 J}{\partial \alpha^2} = I + \gamma Y^t Y + \gamma \sum_{i=1}^n (f_i(\alpha) - q_i^f) H_i \quad (18)$$

donde

$$Y = \frac{\partial f}{\partial \alpha} \in R^{n \times N} \quad y \quad H_i = \frac{\partial^2 f_i}{\partial \alpha^2}, \quad i = 1, \dots, n \quad (19)$$

son la Jacobiana y las Hessianas de f . Entonces las iteraciones k utilizando el enfoque de búsqueda lineal con un paso de Newton aproximado son de la forma

$$\alpha_{k+1} = \alpha_k - \mu[\rho I + Y^t Y]^{-1}[\rho \alpha_k + Y^t(f(\alpha_k) - q_f)] \quad (20)$$

donde $\rho = 1/\gamma$, $0 < \mu < 1$ y $(\rho I + Y^t Y)$ es una matriz positiva definida que reemplaza el cálculo de la Hessiana de J con respecto y evita el cálculo de las Hessianas H_i . De esta forma resolvemos un problema de optimización no lineal sin restricciones. La convergencia hacia la solución exacta depende de N , μ y γ (ver [1]).

III-B. Estrategia de optimización con restricciones

En la estrategia anterior es necesario incluir un término en la función de costo original. Sin embargo, la naturaleza del segundo término es la de una restricción de igualdad. Además, la forma de regular los parámetros γ y μ no es clara. Entre las debilidades del método anterior se encuentra el hecho de omitir la parte que no es positiva definida de la matriz Hessiana para calcular la dirección de Newton. En esta sección, proponemos una solución efectiva y aún más eficiente. La idea es quitar el segundo término en (16) de tal forma que la función de costo sea convexa

$$J(\alpha) = \frac{1}{2} \sum_{j=1}^N \|\alpha_j\|^2 \quad (21)$$

y agregar al problema una restricción de igualdad

$$c(\alpha) = \frac{1}{2} \|f(\alpha) - q_f\|^2 \quad (22)$$

Entonces, planteamos el problema de la siguiente manera

$$\mathcal{L}(\alpha, \lambda) = J(\alpha) - \sum_{i \in I} \lambda_i c_i(\alpha) \quad (23)$$

donde \mathcal{L} es la Lagrangiana del problema, λ_i son los multiplicadores de Lagrange y $c(\alpha_i)$ son las restricciones. Utilizando la PCS, el paso de Newton está dado por

$$\begin{bmatrix} \alpha_{k+1} \\ \lambda_{k+1} \end{bmatrix} = \begin{bmatrix} \alpha_k \\ \lambda_k \end{bmatrix} + \begin{bmatrix} p_k \\ p_\lambda \end{bmatrix} \quad (24)$$

donde p_k y p_λ resuelven el sistema KKT-Newton

$$\begin{bmatrix} \left(\frac{\partial^2 \mathcal{L}}{\partial \alpha^2} \right)_k & - \left(\frac{\partial c}{\partial \alpha} \right)_k^t \\ \left(\frac{\partial c}{\partial \alpha} \right)_k & 0 \\ - \left(\frac{\partial J}{\partial \alpha} \right)_k & + \left(\frac{\partial c}{\partial \alpha} \right)_k^t \lambda_k \\ & - c_k \end{bmatrix} \begin{bmatrix} p_k \\ p_\lambda \end{bmatrix} = \quad (25)$$

Este sistema está bien definido sí y sólo si la matriz Jacobiana de las restricciones $\frac{\partial c}{\partial \alpha}$ es de rango completo y la matriz $\frac{\partial^2 \mathcal{L}}{\partial \alpha^2}$ es positiva definida. Conociendo la dirección, entonces calculamos el tamaño de paso μ . Cuando el problema cuenta con restricciones, las funciones de mérito ayudan a converger hacia el mínimo siempre y cuando no se violen las restricciones. La función que utilizamos es

$$\phi(\alpha, \delta) = J(\alpha) + \delta \|c(\alpha)\| \quad (26)$$

Reemplazamos el cálculo explícito de la matriz $\frac{\partial^2 \mathcal{L}}{\partial \alpha^2}$ mediante la fórmula estandar conocida como BFGS [25] para construir una matriz B_k . Otro ingrediente importante es la evaluación de la condición de *curvatura* antes de actualizar B_{k+1} dada por

$$\begin{aligned} s_k^t y_k &> 0 \\ s_k &= \alpha_{k+1} - \alpha_k, \\ y_k &= \frac{\partial \mathcal{L}}{\partial \alpha}(\alpha_{k+1}, \lambda_{k+1}) - \frac{\partial \mathcal{L}}{\partial \alpha}(\alpha_k, \lambda_k) \end{aligned} \quad (27)$$

III-C. Algoritmo

El algoritmo (1) está basado en la PCS utilizando búsqueda lineal para la generación de trayectorias de sistemas no holonómos sin deriva.

IV. GENERACIÓN DE TRAYECTORIAS

Los métodos de la sección III, los implementamos inicialmente en *Matlab* versión 7.5. Realizamos comparaciones entre estos algoritmos y la función *fmincon* de *Matlab* para optimización no lineal con restricciones. Posteriormente realizamos la implementación en lenguaje ANSI C. Todos los resultados que mostramos se llevaron a cabo en PC con Linux. Utilizamos *BLAS* para el cálculo de las operaciones simples de álgebra lineal y *LAPACK* para resolver operaciones complejas como la inversión de matrices. La biblioteca dinámica está disponible gratuitamente en el repositorio: <http://www.netlib.org/>. Para graficar las trayectorias utilizamos *OpenGL*.

IV-A. Modelos cinemáticos

Utilizamos los siguientes sistemas para probar los algoritmos:

$$\begin{pmatrix} \dot{q}_1 \\ \dot{q}_2 \\ \dot{q}_3 \end{pmatrix} = \begin{pmatrix} \cos q_3 \\ \sin q_3 \\ 0 \end{pmatrix} u_1 + \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix} u_2 \quad (28)$$

$$\begin{pmatrix} \dot{q}_1 \\ \dot{q}_2 \\ \dot{q}_3 \\ \dot{q}_4 \end{pmatrix} = \begin{pmatrix} \cos q_3 \\ \sin q_3 \\ \frac{\tan q_4}{L} \\ 0 \end{pmatrix} u_1 + \begin{pmatrix} 0 \\ 0 \\ 0 \\ 1 \end{pmatrix} u_2 \quad (29)$$

Algorithm 1: Algoritmo numérico basado en PCS para el cálculo de trayectorias

Entrada: 1. Configuraciones inicial y final: q_0 y $q_f \in R^n$

2. Una distribución $B(q) \in R^{n \times m}$

Salida : El control $u(\tau) \in R^m$, $\tau \in [0, T]$, que lleva al sistema de q_0 a q_f

begin

Paso 0: Escoger una base ortonormal Φ ;

Paso 1: Iniciar con $\alpha_0 \neq 0$ mediante algún proceso aleatorio y $\lambda_0 = 1$;

Paso 2: Escoger $\eta \in (0, \frac{1}{2})$ y $\rho \in (0, 1)$;

Paso 3: Calcular p_k con (25) ;

Paso 4: Escoger $\delta_k > \|\lambda_{k+1}\|_\infty$;

Paso 5: Asignar $\mu = 1$;

Mientras $\phi(\alpha_k + \mu_k p_k; \delta_k) > \phi(\alpha_k; \delta_k) + \eta \alpha_k D(\phi(\alpha_k; \delta_k) p_k)$
asigna $\mu_k = \rho \mu$ para algún $\rho \mu \in (0, \rho]$;

Paso 7: Asignar $\alpha_{k+1} = \alpha_k + \mu_k p_k$ y

$\lambda_{k+1} = \lambda_k + \mu_k p_\lambda$;

Paso 8: Evaluar $J_{k+1}, (\frac{\partial J}{\partial \alpha})_{k+1}, c_{k+1}, (\frac{\partial c}{\partial \alpha})_{k+1}$;

Si la condición (27) se cumple **Entonces**

obtener la matriz B_{k+1} actualizando

B_k mediante BFGS ;

Si $TOL < \|c\|$ **o** num. máximo de iteraciones

Entonces

Fin ;

De lo contrario

regresar al **Paso 3** ;

end

$$\begin{pmatrix} \dot{q}_1 \\ \dot{q}_2 \\ \dot{q}_3 \\ \dot{q}_4 \end{pmatrix} = \begin{pmatrix} \cos q_3 \\ \sin q_3 \\ q_4 \\ 0 \end{pmatrix} u_1 + \begin{pmatrix} 0 \\ 0 \\ 0 \\ 1 \end{pmatrix} u_2 \quad (30)$$

$$\begin{pmatrix} \dot{q}_1 \\ \dot{q}_2 \\ \dot{q}_3 \end{pmatrix} = \begin{pmatrix} \frac{1}{2} \cos q_3 \\ \frac{1}{2} \sin q_3 \\ \frac{1}{L} \end{pmatrix} u_1 + \begin{pmatrix} \frac{1}{2} \cos q_3 \\ \frac{1}{2} \sin q_3 \\ \frac{-1}{L} \end{pmatrix} u_2 \quad (31)$$

$$\begin{pmatrix} \dot{q}_1 \\ \dot{q}_2 \\ \dot{q}_3 \\ \dot{q}_4 \\ \dot{q}_5 \end{pmatrix} = \begin{pmatrix} \cos q_3 \\ \sin q_3 \\ 0 \\ \frac{\sin q_4}{L_1} \\ \frac{-\sin q_4}{L_1} + \frac{\sin q_6 \cos q_5}{L_2} \end{pmatrix} u_1 + \begin{pmatrix} 0 \\ 0 \\ 1 \\ 1 \\ 0 \end{pmatrix} u_2 \quad (32)$$

Una consideración importante es el análisis de controlabilidad de los sistemas. Aplicando la condición de rango del álgebra de Lie (LARC [26]) en todos los sistemas verificamos que son controlables, esto quiere decir que podemos encontrar trayectorias para ir de cualquier configuración inicial a cualquier otra final (ver [4]). Las condiciones

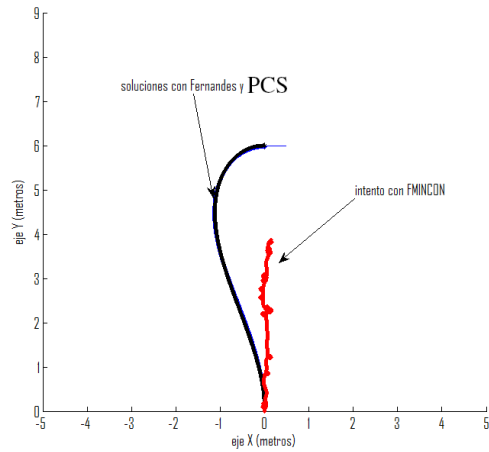


Fig. 1. Trayectoria calculada con FMINCON en rojo y con Fernandes y PCS en negro.

necesarias para encontrar trayectorias óptimas también han sido estudiadas y verificadas para estos sistemas [27]. Si deseamos posteriormente utilizar los algoritmos numéricos para otros sistemas sin deriva que no estén descritos en este trabajo, entonces es necesario considerar previamente el análisis de controlabilidad y las condiciones necesarias sobre la existencia de trayectorias óptimas. De otra manera, no será posible garantizar la convergencia de los algoritmos.

En la Figura (1) mostramos un ejemplo representativo del resultado obtenido con la función *fmincon*. Evidentemente los resultados no son satisfactorios aún cuando las trayectorias son relativamente simples. En la Figura (2) mostramos trayectorias más complejas. El algoritmo de Fernandes *et al.* tardó 24 iteraciones en llegar al resultado.

IV-B. Parámetros de entrada

Los parámetros necesarios para ejecutar los algoritmos son básicamente los siguientes:

- α_0 , es necesario proporcionar una solución inicial
- La dimensión de α
- El tiempo final T
- la matriz $B(q)$

Los algoritmos son naturalmente sensibles al parámetro inicial α_0 . Si lo escogemos adecuadamente, esto es, razonablemente cerca de la solución deseada, entonces, los algoritmos convergen rápidamente (ver Figura (2)). De lo contrario, si α_0 está muy alejada de la solución óptima, entonces, el algoritmo de la sección III-A no logrará converger adecuadamente a menos que, en cada iteración, modifiquemos inteligentemente el tamaño de paso y la ponderación para regular la minimización de la energía y el error en distancia entre la configuración final en la iteración actual y la configuración final deseada.

Las trayectorias en negro y en azul representan la integración del sistema con α^* utilizando el método de Fernandes *et al.* y el propuesto por nosotros respectivamente. Visualmente la diferencia no es relevante entre las trayectorias. Sin embargo, el algoritmo de Fernandes tardó 24 iteraciones para

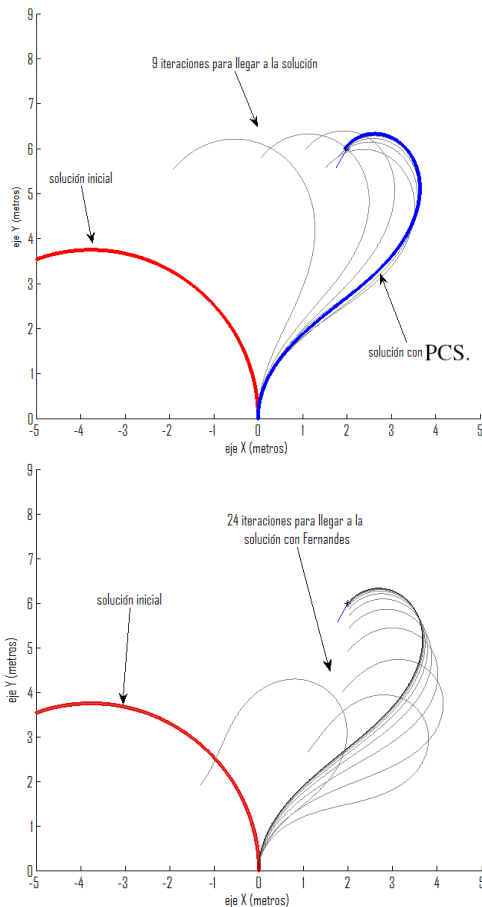


Fig. 2. Trayectorias generadas con el método de Fernandes *et al* y con PCS

encontrar la solución mientras que el método que utiliza PCS sólo iteró 9 veces. Cabe mencionar que ambos algoritmos no emplean el mismo criterio de paro. El algoritmo de Fernandes *et al.* se detiene cuando la longitud de paso de Newton es igual o menor a una tolerancia. En este caso la tolerancia la fijamos en $1.0e-5$. El método con PCS se detiene cuando la norma de las restricciones es igual o menor a una tolerancia. En este caso la tolerancia es $1.0e3$. Por supuesto, ambos algoritmos paran si exceden el número máximo de iteraciones. En este caso es 50. El número de variables (componentes del vector α) fue 20.

La ventaja del algoritmo (1), es justamente el manejo automático de la longitud de paso y la satisfacción de las restricciones. Esto ocasiona un comportamiento robusto. Por lo tanto, aún cuando α_0 está alejada de la solución óptima, podemos garantizar que el algoritmo converja a un óptimo local respetando al máximo las restricciones

En la Figura (3) mostramos un ejemplo con el sistema (30) que ilustra la convergencia de los algoritmos cuando la solución inicial está alejada de la solución óptima. La trayectoria en rojo representa la integración del sistema con α_0 , esto es, la solución inicial. Las trayectorias en negro y azul representan la integración del sistema con α^* utilizando el método de Fernandes *et al.* y con PCS

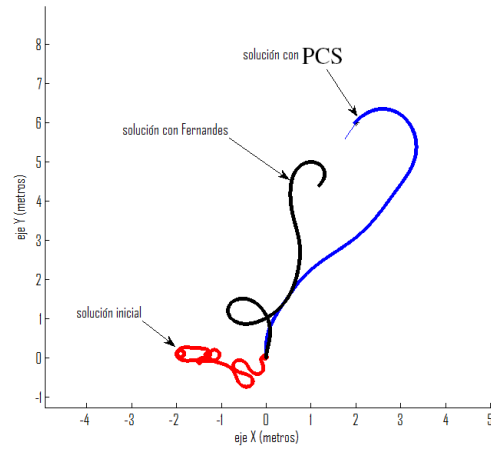


Fig. 3. Generación de trayectorias cuando la solución inicial está alejada

respectivamente. El algoritmo de Fernandes *et al.* llegó a las 50 iteraciones sin obtener la solución. El método con PCS tardó 33 iteraciones para encontrar la solución. En la cuarta iteración identificó que la matriz Hessiana dejó de ser positiva definida y no la actualizó en esa iteración.

El tiempo final T depende de la distancia que existe entre la configuración inicial y la final. Si la distancia es muy grande y T pequeña entonces no existe manera de converger adecuadamente si el tiempo de muestreo es siempre el mismo. También existe el caso contrario cuando T es muy grande para una distancia corta. En este caso, la energía se propaga a lo largo de la trayectoria dando como resultado una trayectoria con vueltas amplias y no necesarias.

Finalmente en la Figura (4) mostramos algunos ejemplos cuando la posición y la dirección final cambian.

Después de afinar varios detalles en la implementación que realizamos en ANSI C, logramos obtener el mismo número de iteraciones que en *Matlab*. Por lo tanto, la precisión numérica es similar. Sin embargo, los rangos con respecto al tiempo de cálculo usando ANSI C son

1. PCS, algoritmo (1): varía de 0.8 a 1.2 segundos.
2. Fernandes *et al.*: varía 1.8 de 2.7 segundos.

El procesador que usamos fue un Intel Core2 Duo con 1.5GHz en cada núcleo.

V. CONCLUSIONES

En este trabajo presetamos una estrategia numérica eficiente para generar trayectorias de sistemas sin deriva y con restricciones diferenciales no integrables. Primero seguimos el enfoque analítico para obtener información parcial sobre las trayectorias óptimas según la minimización de la norma L_2 . Debido a que esta información no es suficiente para obtener dichas trayectorias, entonces recurrimos a los métodos de optimización numérica. Proponemos resolver el problema de optimización no lineal con restricciones de igualdad mediante un algoritmo simple basado en la PCS con búsqueda lineal. Implementamos la estrategia en lenguaje ANSI C para que sea transparente su utilización con robots móviles. Desde los años ochenta existen trabajos reportados

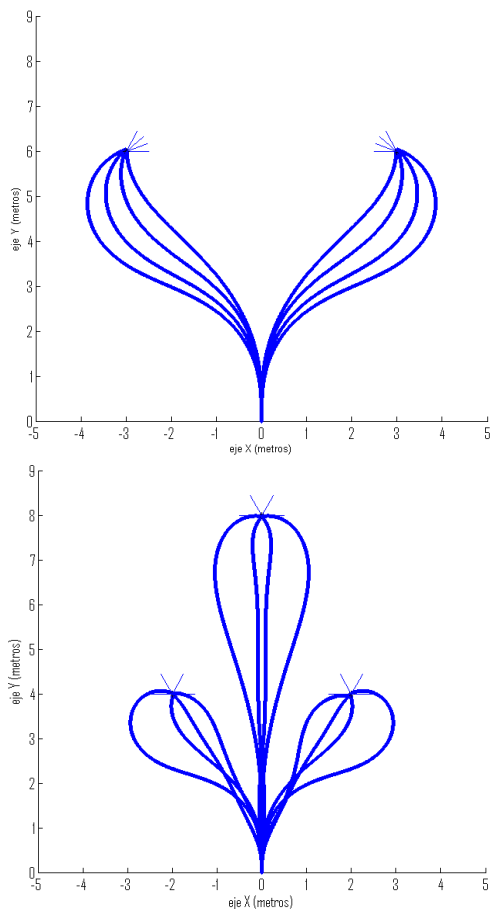


Fig. 4. Simetría de las trayectorias cuando variamos la configuración final.

sobre la planificación de movimientos para sistemas no holonómicos y actualmente continua siendo una línea de investigación vigente. De hecho, existen estudios recientes sobre las trayectorias de marcha humana y humanoide basados en este tipo de sistemas [28], [29].

VI. AGRADECIMIENTOS

Agradecemos el soporte financiero del CONACyT por medio del proyecto No. 84855 para desarrollar este trabajo.

REFERENCES

- [1] C. Fernandes, L. Gurvits, and Z. Li, "Near-optimal nonholonomic motion planning for a system of coupled rigid bodies," *IEEE Transactions on Automatic Control*, vol. 39(3), pp. 450–463, 1994.
- [2] A. Bellaïche and J.-J. Risler, *SubRiemannian Geometry*. Progress in Math. 144, Birkhauser, Basel, 1996.
- [3] Z. Li and J. Canny, *Nonholonomic Motion Planning*. Kluwer, Boston, 1993.
- [4] J.-P. Laumond, S. Sekhavat, and F. Lamiroux, *Robot Motion Planning and Control*. Lectures Notes in Control and Information Sciences 229, Springer, 1998, ch. Guidelines in Nonholonomic Motion Planning for Mobile Robots.
- [5] L.-E. Dubins, "On curves of minimal length with a constraint on average curvature and with prescribed initial and terminal positions and tangents," *American Journal of Mathematics*, vol. 79, pp. 497–516, 1957.
- [6] J.-A. Reeds and R.-A. Shepp, "Optimal paths for a car that goes both forward and backwards," *Pacific Journal of Mathematics*, vol. 145 (2), pp. 367–393, 1990.

- [7] H.-J. Sussmann and W. Tang, "Shortest paths for reeds-shepp car: a worked out example of the use of geometric techniques in nonlinear control," *Report SYCON-91-10*, Rutgers University, 1991.
- [8] J.-D. Boissonnat, A. Cerezo, and J. Leblong, "Shortest paths of bounded curvature in the plane," in *IEEE Int. Conf. on Robotics and Automation*, Nice, France, 1992.
- [9] L.-S. Pontryagin, V.-G. Boltyanskii, R.-V. Gamkrelidze, and E.-F. Mishchenko, *The Mathematical Theory of Optimal Processes*. Pergamon Press, 1964.
- [10] P. Souères and J.-P. Laumond, "Shortest path synthesis for a car-like robot," *IEEE Transactions on Automatic Control*, vol. 41(5), pp. 672–688, 1996.
- [11] T. Pecsvaradi, "Optimal horizontal guidance law for aircraft in the terminal area," *IEEE Transactions on Automatic Control*, vol. 17(6), pp. 763–772, 1972.
- [12] D. Balkcom and M. Mason, "Time optimal trajectories for bounded velocity differential drive vehicles," *International Journal of Robotics Research*, vol. 21(3), pp. 199–217, 2002.
- [13] D. Balkcom, P.-A. Kavathekar, and M. Mason, "Fastest trajectories for an omni-directional vehicle," *International Workshop on Algorithmic Foundations of Robotics*, July 2006.
- [14] S. Bhattacharya, R. Murrieta-Cid, and S. Hutchinson, "Optimal paths for landmark-based navigation by differential drive vehicles with field-of-view constraints," *IEEE Transactions on Robotics*, vol. 23, no. 2, pp. 47–59, February 2007.
- [15] J.-B. Hayet, C. Esteves, and R. Murrieta-Cid, "A motion planner for maintaining landmark visibility with a differential drive robot," in *Proc. WAFR 2008 to be published in Springer Tracts in Advanced Robotics*, 2009.
- [16] W. Huifang, C. Yangzhou, and P. Souères, "A geometric algorithm to compute time-optimal trajectories for a bidirectional steered robot," *IEEE Transactions on Robotics*, vol. 25(1), pp. 399–413, 2009.
- [17] J.-D. Boissonnat, A. Cerezo, and J. Leblong, "A note on shortest paths in the plane subject to a constraint on the derivative of the curvature," *INRIA, Nice-Sophia-Antipolis, Research Report 2160*, 1994.
- [18] V.-P. Kostov and E.-V. Degtiarova-Kostova, "The planar motion with bounded derivative of the curvature and its suboptimal paths," *Acta Mathematica Universitatis Comeianae*, vol. 64, pp. 185–226, 1995.
- [19] J. Ostrowski, J.-P. Desai, and V. Kumar, "Optimal gait selection for nonholonomic locomotion systems," *International Journal of Robotics Research*, vol. 19, pp. 225–237, 1997.
- [20] A. Divelbiss and J. Wen, "A path space approach to nonholonomic motion planning in the presence of obstacles," *IEEE Transactions on Robotics and Automation*, vol. 13(3), pp. 443–451, 1997.
- [21] E. Sontag, "Control of systems without drift via generic loops," *IEEE Transactions on Automatic Control*, vol. 40, pp. 1210–1219, 1995.
- [22] O. L. F. Lamiroux, D. Bonnafous, "Reactive path deformation for nonholonomic mobile robots," *IEEE Transactions on Robotics*, vol. 20, no. 6, pp. 967–977, Dec 2004.
- [23] A. K. T. Howard, "Optimal rough terrain trajectory generation for wheeled mobile robots," *Optimal Rough Terrain Trajectory Generation for Wheeled Mobile Robots*, vol. 26, no. 1, pp. 141–166, February 2007.
- [24] S.-S. Sastry and R. Montgomery, "The structure of optimal controls for a steering problem," in *IFAC Workshop on Nonlinear Control*, 1992, pp. 385–390.
- [25] J. Nocedal and S.-J. Wright, *Numerical Optimization*. Springer-Verlag, 1999.
- [26] H.-J. Sussmann, *Nonlinear Controllability and Optimal Control*. New York: Marcel Dekker, Inc., 1990.
- [27] L. Cesari, *Optimization, theory and applications*. Springer-Verlag, 1983.
- [28] G. Arechavaleta, J.-P. Laumond, H. Hicheur, and A. Berthoz, "An optimality principle governing human walking," *IEEE Transactions on Robotics*, vol. 24(1), pp. 5–14, 2008.
- [29] K. Mombaur, J.-P. Laumond, and E. Yoshida, "An optimal control model unifying holonomic and nonholonomic walking," in *8th IEEE-RAS International Conference on Humanoid Robots*, 2008, pp. 646–653.