Optimización de Trayectorias para Sistemas Sujetos a Restricciones No Holónomas

Trajectory Optimization for Systems Under Nonholonomic Constraints

Gustavo Arechavaleta
Robótica y Manufactura Avanzada, CINVESTAV – Unidad Saltillo
Carretera Saltillo-Monterrey Km. 13.5, C.P. 25900, Ramos Arizpe, Coah. México
garechav@cinvestav.edu.mx

Artículo recibido en Enero 15, 2010; aceptado en Junio 11, 2010

Resumen. Presentamos una estrategia numérica para calcular trayectorias válidas para sistemas sin deriva con restricciones diferenciales no integrables que minimicen el consumo de energía expresado como la norma L_2 del control. Utilizamos herramientas de la teoría del control óptimo y la programación no lineal para formular y resolver el problema de optimización. Primero analizamos las condiciones necesarias que debe satisfacer el control óptimo. Posteriormente convertimos el problema de dimensión infinita a un problema de optimización no lineal de dimensión finita. Esta formulación nos permite generar las trayectorias deseadas utilizando una estrategia simple y eficiente basada en la Programación Cuadrática Secuencial (PCS).
Comparamos la estrategia propuesta con el algoritmo desarrollado por [Fernandes, et al., 1994], en términos de convergencia y tiempo de cálculo, utilizando varios modelos cinemáticos de robots móviles con ruedas y remolques y también un modelo dinámico de robot espacial.
Palabras clave: sistemas no holónomos, control óptimo, optimización numérica, robótica móvil.

Abstract. This paper presents a numerical strategy to compute feasible trajectories for driftless systems under nonintegrable differential constraints that minimize the L_2 norm of the control. We made use of optimal control tools and nonlinear programming to formulate and solve the optimization problem. First, we analyze the necessary conditions to be satisfied by the optimal control. Then, we transform the infinite-dimensional problem into a finite-dimensional nonlinear optimization problem. This formulation allows us to generate the desired trajectories by using a simple and efficient strategy based on the Sequential Quadratic Programming (SQP).
We compare the proposed strategy with the algorithm developed by [Fernandes, et al., 1994], in terms of convergence and computational time, by using various kinematic models of mobile robots with wheels, chained systems and a dynamic model of space robot.
Keywords. Nonholonomic systems, optimal control, numerical optimization, mobile robotics.

1 Introducción

La planificación de movimientos es un área de la robótica que ha contribuido sustancialmente a la concepción y desarrollo de algoritmos eficientes para generar automáticamente la secuencia de movimientos que los robots deben efectuar para realizar una o varias tareas asignadas [Latombe, 1991]. Estas tareas pueden representar restricciones de igualdad (bilaterales) y/o desigualdad (unilaterales) en el espacio de configuración (CS) del mecanismo compuesto por una serie de cuerpos rígidos articulados. La dimensión y la topología del CS están determinadas por el número de grados de libertad (g.d.l) y el tipo de articulaciones del mecanismo respectivamente. En el CS la secuencia de desplazamientos necesarios para unir dos configuraciones del mecanismo se reduce al cálculo de rutas factibles que un punto debe recorrer sobre una variedad diferencial[1].

En la versión más simple del problema de planificación de movimientos deseamos desplazar al mecanismo en cuestión de una configuración inicial a otra final sin colisionar con los obstáculos presentes en el entorno. Cabe mencionar que la estructura cinemática y la geometría del mecanismo y de los obstáculos en el entorno son conocidas. Entonces, encontrar una posible solución implica calcular eficientemente rutas libres de colisión (si existen) en las regiones factibles del CS. En este caso, el problema *completo* de movimiento, donde intervienen la cinemática y la

[1] Una variedad (resp. variedad diferencial) es un espacio topológico tal que cualquier punto $q \in M$ contiene una vecindad abierta homeomorfa (resp. difeomorfa) a un conjunto abierto en $\mathbb{R}^n$ donde n es la dimensión de M.

dinámica, se simplifica hasta reducirlo a un problema puramente geométrico. Sin embargo, en esta versión simplificada aún encontramos varias dificultades, entre ellas se encuentran:

1) las restricciones unilaterales principalmente relacionadas con la evasión de obstáculos y con los límites articulares propios del mecanismo,
2) la estructura mecánica del sistema se complica de acuerdo al número de g.d.l. que posee. En consecuencia, la dimensión del CS es mayor y su topología más compleja.

En la literatura existen estrategias relativamente eficientes basadas en el muestreo del CS para capturar su topología en una estructura tipo árbol o grafo [Choset, et al., 2005; LaValle, 2006]. Los nodos representan configuraciones válidas libres de colisión y las aristas que unen a los nodos representan desplazamientos articulares del mecanismo. Siguiendo este enfoque, varios ingredientes son de gran importancia:

1) la *técnica de muestreo* que permite generar configuraciones en las regiones factibles del CS,
2) el *detector de colisiones* en el espacio de trabajo para validar que las configuraciones no están en colisión,
3) el *planificador local* eficiente que permita generar las aristas entre nodos del grafo. En otras palabras, las geodésicas en el CS que unan pares de configuraciones.

Cuando el sistema cuenta con restricciones diferenciales, la estructura del espacio de búsqueda se vuelve más compleja y el componente crítico para generar trayectorias factibles en el espacio de estados (Q) es el *planificador local*. Ahora Q está compuesto por CS junto con su fibrado tangente (TCS) donde habitan las velocidades.

En la planificación de movimientos de robots, dos clases de restricciones diferenciales han sido estudiadas ampliamente: las integrables y las que no son integrables. También se llaman *holónomas* y *no holónomas* respectivamente. Las restricciones *holónomas*, al momento de integrarlas, permiten reducir la dimensión del CS, y por lo tanto, es posible realizar la planificación de las rutas en una subvariedad sin restricciones contenida en CS y de la misma dimensión que la del espacio de control. Sin embargo, existen varios robots móviles con ruedas y manipuladores espaciales, entre otros, que están sujetos a características cinemáticas y/o dinámicas restrictivas que no se pueden integrar, esto es, *no holónomas*. El estudio de esta clase de sistemas ha generado interés en la comunidad de matemáticas (e.g., [Bellaiche y Risler, 1996]), teoría de control (e.g., [Li y Canny, 1993]) y robótica (e.g., [Laumond, et al., 1998]). Algunos métodos para generar trayectorias válidas para esta clase de mecanismos están basados en la teoría del control óptimo. Aunque la solución factible y óptima para cualquier sistema *no holónomo* es desconocida, existen algunas excepciones para caracterizar analíticamente las rutas más cortas entre cualquier par de configuraciones en la variedad $\mathbb{R}^2 \times \mathbb{S}^1$ cuando la curvatura está acotada [Dubins, 1957; Reeds y Shepp, 1990]. En [Sussmann y Tang, 1991] y [Boissonnat, et al., 1992] utilizaron el Principio Máximo de Pontryagin (PMP) [Pontryagin, et al., 1964] para resolver el mismo problema. En [Pecsvardi, 1972] y [Souères y Laumond, 1996] utilizaron el PMP para resolver la síntesis completa de las rutas más cortas para el mismo caso particular. También han sido reportadas algunas variantes del problema siguiendo el mismo enfoque [Boissonat, et al., 1994; Kostov y Degtiariova-Kostova, 1995; Balkcom y Mason, 2002; Balkcom, et al., 2006; Bhattacharya, et al., 2007; Hayet, et al., 2010; Huifang, et al., 2009; Salaris et al., 2010].

Debido a la complejidad que representa encontrar la solución óptima global para desplazar cualquier sistema *no holónomo*, la alternativa radica en la utilización de métodos numéricos [Betts, 1998]. Entre las estrategias numéricas que han sido propuestas en la literatura se encuentra el algoritmo de [Fernandes, et al., 1994]. La idea subyacente del método consiste 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 [Sontag, 1995; Divelbiss y Wen, 1997; Ostrowski, et al., 2000; Lamiraux, et al., 2004; Howard y Kelly, 2007]. 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 estos métodos numéricos 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 hecho, el algoritmo que proponemos permite generar trayectorias para sistemas sin deriva con restricciones *no holónomas* a nivel cinemático y dinámico. En particular, calculamos las trayectorias que minimizan la norma L_2 del control. En la sección 2 exploramos la estructura de las restricciones diferenciales mediante herramientas analíticas. Posteriormente en la sección 3 describimos la formulación variacional del problema de optimización. En la sección 4 revisamos la formulación numérica del problema y describimos la estrategia que proponemos usando PCS. En la sección 5 mostramos el desempeño de la estrategia utilizando varios sistemas con diferente grado de complejidad y comparamos el método con el algoritmo reportado en [Fernandes, et al., 1994] y con la función *fmincon* que proporciona el paquete informático *MATLAB*. En la sección 6 presentamos algunos comentarios finales.

2 Estructura de las restricciones diferenciales

Sabemos que varios sistemas robóticos están sujetos a restricciones diferenciales inherentes a su estructura mecánica. Estas restricciones también pueden surgir como consecuencia de la tarea de movimiento asignada. Si las restricciones pueden expresarse como $n-m$ restricciones algebraicas en el CS, donde n representa el número de g.d.l del sistema robótico y m la dimensión del espacio de control, entonces decimos que dichas restricciones son *integrables* y, por lo tanto, *holónomas*. En este trabajo nos interesamos en restricciones diferenciales que no se pueden integrar, esto es, *no holónomas*. Básicamente, este tipo de restricciones aparecen a nivel cinemático cuando existe contacto entre cuerpos que ruedan pero no se deslizan. A nivel dinámico estas restricciones aparecen debido a la conservación del momento angular en sistemas multicuerpo. Los ejemplos representativos de restricciones *no holónomas* a nivel cinemático son: 1) la manipulación fina de objetos mediante una mano robótica; 2) varios robots móviles con ruedas. En aplicaciones con manipuladores espaciales también las encontramos pero a nivel dinámico. La forma implícita y general para representarlas es:

$$G(q,\dot{q}) = 0 \tag{1}$$

Sin embargo, para propósitos relacionados con el cálculo de trayectorias factibles de mecanismos sujetos a este tipo de restricciones es deseable encontrar una representación paramétrica de la expresión 1. En otras palabras, la forma de la restricción en la ecuación 1 debe representarse mediante un sistema de ecuaciones diferenciales de la forma:

$$\dot{q} = f(q,u) \tag{2}$$

donde $q \in \mathbb{R}^n$, que habita en la variedad diferencial M, representa el estado del sistema y $u \in \mathbb{R}^m$ el control. En particular, la forma de los sistemas de control sin deriva que abordaremos es la siguiente:

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

donde $B(q) \in \mathbb{R}^{n\times m}$ es una distribución diferencial $\Delta(q)$. En otras palabras, en cada punto $q \in M$ la familia de vectores $g_1(q), \dots, g_m(q)$ genera un subespacio lineal $\Delta(q)$ en el espacio tangente T_qM. Si $\Delta(q)$ es *regular* entonces su dimensión no varía con respecto a q. Cada $g \in \mathbb{R}^n$ representa una transformación diferencial que asigna en cada q de $\mathbb{R}^n$ un vector tangente $g(q) \in T_qM$.

Para identificar si un mecanismo está sujeto a restricciones *no holónomas* es necesario introducir algunas nociones y herramientas elementales de geometría diferencial [Brockett, 1976].
Esencialmente, nos interesamos en el análisis de *integrabilidad* de las restricciones diferenciales que precisamos en la primera parte de esta sección. En la segunda parte, describimos las propiedades de *controlabilidad* de mecanismos expresados mediante la ecuación 3 para decidir *a priori* si existen soluciones factibles que unan pares de configuraciones o estados.

2.1 Restricciones holónomas y no holónomas

La *integrabilidad* de restricciones diferenciales definidas mediante una distribución Δ está relacionada con la dimensión del espacio alcanzable del sistema en cuestión. Consideremos una distribución Δ de dimensión m en M. Si M

puede descomponerse en subvariedades y $\Delta(q)$ forma una base del espacio tangente T_qM de la subvariedad que pasa por q, entonces Δ es *integrable*. Si $m = n$ quiere decir que q puede moverse libremente en M. Si $m < n$ entonces $\Delta(q)$ es un subespacio lineal de T_qM en q. En este caso, si $\Delta(q)$ puede integrarse de tal forma que localmente la dimensión de M se reduzca a m, entonces el sistema es *holónomo*. En el caso opuesto, $\Delta(q)$ no se puede integrar y, por lo tanto, el sistema no es *holónomo*.

Para decidir si una distribución Δ es integrable es necesario utilizar el teorema de Frobenius. De hecho, basta probar que Δ esté cerrada bajo el operador del corchete de Lie definido como

$$[g_\mathrm{i}, g_\mathrm{j}] = \frac{\partial g_\mathrm{j}}{\partial q} g_\mathrm{i} - \frac{\partial g_\mathrm{i}}{\partial q} g_\mathrm{j} \tag{4}$$

donde g_i y g_j son elementos base de Δ. En otras palabras, si no es posible generar un elemento g_k linealmente independiente, $g_\mathrm{k} = [g_\mathrm{i}, g_\mathrm{j}]$, entonces Δ es integrable. Es importante mencionar que no es necesario realizar todas las combinaciones entre los elementos $g_1(q), \dots, g_m(q)$ gracias a la propiedad antisimétrica $[g_\mathrm{i}, g_\mathrm{j}] = -[g_\mathrm{j}, g_\mathrm{i}]$ del corchete de Lie.

Ejemplo 1: Restricción diferencial integrable: Consideremos la siguiente restricción implícita

$$q_1\dot{q}_1 + q_2\dot{q}_2 + q_3\dot{q}_3 = 0 \tag{5}$$

Integrando la ecuación 5 con respecto al tiempo obtenemos que

$$q_1^2 + q_2^2 + q_3^2 - c^2 = 0 \tag{6}$$

Por lo tanto, la restricción diferencial 5 puede expresarse como una restricción algebraica en $CS = \mathbb{R}^3$. La subvariedad sin restricciones es $\mathbb{S}^2$. El siguiente sistema es una representación paramétrica de la restricción 5

$$\begin{pmatrix} \dot{q}_1 \\ \dot{q}_2 \\ \dot{q}_3 \end{pmatrix} = \begin{pmatrix} q_2 \\ -q_1 \\ 0 \end{pmatrix} u_1 + \begin{pmatrix} q_3 \\ 0 \\ -q_1 \end{pmatrix} u_2 \tag{7}$$

donde $(u_1, u_2) \in \mathbb{R}^2$ representan las variables de control. Aplicando el operador de la ecuación 4 entre los campos vectoriales del sistema 7 es posible verificar que $\det(g_1\ g_2\ [g_1, g_2]) = 0$, para cualquier q. Los vectores columna son

$$g_1 = \begin{pmatrix} q_2 \\ -q_1 \\ 0 \end{pmatrix}; g_2 = \begin{pmatrix} q_3 \\ 0 \\ -q_1 \end{pmatrix};\ [g_1, g_2] = \begin{pmatrix} 0 \\ q_3 \\ -q_2 \end{pmatrix} \tag{8}$$

En consecuencia podemos concluir que la restricción diferencial expresada en la ecuación 5 es *holónoma*.

Ejemplo 2: Restricción diferencial no integrable: Consideremos ahora el movimiento cinemático del monociclo sobre el plano. La posición está determinada mediante las coordenadas, $(q_1, q_2) \in \mathbb{R}^2$, relativas al contacto del artefacto con el plano. La dirección está definida mediante el ángulo, $q_3 \in \mathbb{S}^1$, existente entre el monociclo y el eje horizontal. La siguiente restricción diferencial indica que el monociclo rueda sin deslizarse

$$\sin q_3\, \dot{q}_1 - \cos q_3\, \dot{q}_2 = 0 \tag{9}$$

La ecuación 9 puede expresarse mediante el siguiente conjunto de ecuaciones diferenciales

$$\begin{pmatrix} \dot{q}_1 \\ \dot{q}_2 \\ \dot{q}_3 \end{pmatrix} = \begin{pmatrix} \cos q_3 \\ \sin q_3 \\ 0 \end{pmatrix} u_1 + \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix} u_2 \tag{10}$$

donde $(u_1, u_2) \in \mathbb{R}^2$ representan la velocidad lineal y angular respectivamente. Aplicando el operador de la ecuación 4 entre los campos vectoriales del sistema 10 es posible probar que $\det(g_1\ g_2\ [g_1, g_2]) \neq 0$, donde

$$g_1 = \begin{pmatrix} \cos q_3 \\ \sin q_3 \\ 0 \end{pmatrix};\ g_2 = \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix};\ [g_1, g_2] = \begin{pmatrix} \sin q_3 \\ -\cos q_3 \\ 0 \end{pmatrix} \tag{11}$$

En consecuencia podemos concluir que la restricción diferencial expresada en la ecuación 9 no es integrable o, de forma equivalente, *no holónoma*.

2.2 Existencia de trayectorias factibles

Las propiedades de *controlabilidad* en sistemas robóticos ayudan a decidir si cualquier par de estados puede unirse mediante una trayectoria (ver [Sussmann 1990; Laumond et al., 1998]). En este

trabajo nos interesamos en la generación de trayectorias óptimas para sistemas definidos por un conjunto de ecuaciones diferenciales (ver ecuación 3). Sin embargo, las trayectorias óptimas representan un subconjunto de trayectorias factibles. Por lo tanto, *la controlabilidad* de los sistemas es una condición necesaria que permite verificar que el conjunto de trayectorias factibles para dichos sistemas no esté vacío. En este contexto, *la condición de rango del algebra de Lie* (LARC, ver [Sussmann, 1990]) nos indica si el sistema es controlable. El *algebra de Lie*, $LA(g_i)$, está compuesta por el conjunto de campos vectoriales $g_1(q), \dots, g_m(q)$ junto con el operador del corchete de Lie (ver ecuación 4). Si $[g_\mathrm{i}, g_\mathrm{j}] \neq 0$ entonces significa que g_i y g_j no conmutan y si $g_\mathrm{k} = [g_\mathrm{i}, g_\mathrm{j}]$ no es una combinación lineal de $g_1(q), \dots, g_m(q)$ entonces g_k representa una nueva dirección de movimiento para el sistema. Si consideramos una distribución $\Delta = \mathrm{gen}(g_1(q), \dots, g_m(q))$, entonces la prueba de *controlabilidad* consiste en generar elementos linealmente independientes para completar la base del espacio tangente para cualquier q tal que $\overline{\Delta} = \mathrm{gen}(g_1(q), \dots, g_m(q), [g_\mathrm{i}, g_\mathrm{j}], [g_\mathrm{k}, [g_\mathrm{i}, g_\mathrm{j}]], \dots)$, donde $LA(\Delta) = \overline{\Delta}$ y $\dim(\overline{\Delta}) = \dim(T_q M)$.

Regresando al ejemplo 2 del monociclo, podemos verificar que el sistema es controlable mediante la siguiente operación $LA(\Delta) = \mathrm{gen}([\cos q_3 \;\; \sin q_3 \;\; 0]^\mathrm{T}, [0\ 0\ 1]^\mathrm{T}, [\sin q_3 \;\; -\cos q_3 \;\; 0]^\mathrm{T})$.

3 Enfoque variacional del problema de optimización

En esta sección formulamos el problema de optimización y mostramos la información parcial que logramos obtener utilizando herramientas analíticas. Consideramos sistemas definidos por la ecuación 3. 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 exsite) que minimice la funcional:

$$J = \int_0^T \mathcal{L}\left(q(\tau, u), u(\tau)\right) d\tau \tag{12}$$

donde $\mathcal{L}(q, u)$ es una función continua y diferenciable con respecto a sus parámetros y T es el tiempo final deseado.

La función Hamiltoniana que deseamos minimizar se define como

$$H(q, u, \psi) = \mathcal{L}(q, u) + \langle \psi, f(q, u) \rangle \tag{13}$$

donde ψ representa el estado adjunto asociado a q y $\langle \cdot, \cdot \rangle$ el producto interno en $\mathbb{R}^n$. El sistema satisface las siguientes ecuaciones

$$\dot{q} = \frac{\partial H}{\partial \psi}(q, \psi) \tag{14.a}$$

$$\dot{\psi} = -\frac{\partial H}{\partial q}(q, \psi) \tag{14.b}$$

$$\frac{\partial H}{\partial u} = 0 \tag{14.c}$$

junto con las condiciones de frontera $q(0) = q_0$ y $q(T) = q_f$. En particular, deseamos minimizar la norma L_2 del control definida por la siguiente funcional

$$J = \frac{1}{2} \int_0^T < (u(\tau), u(\tau)) > d\tau \tag{15}$$

sujeta a las restricciones dadas en la ecuación 3. En consecuencia, H posee la siguiente estructura

$$H(q, u, \psi) = \frac{1}{2} u^T u + \psi^T \sum_{i=1}^{m} g_i(q) u_i \tag{16}$$

Utilizando las ecuaciones 14.c y 16, obtenemos el control óptimo

$$u_i^* = -\psi^T g_i(q) \tag{17}$$

y la Hamiltoniana óptima

$$H^*(q, \psi) = -\frac{1}{2} \sum_{i=1}^{m} \left(\psi^T g_i(q)\right)^2 \tag{18}$$

La estructura del control óptimo para este problema ha sido estudiada en [Sastry y Montgomery, 1992]. Los autores encontraron un resultado interesante al diferenciar la ecuación 17 con respecto al tiempo, esto es

$$\dot{u}_i = -\dot{\psi}^T g_i(q) - \psi^T \frac{\partial g_i}{\partial q} \dot{q} \tag{19}$$

y utilizando la ecuación 14.b que define a la función Hamiltoniana para $\dot{\psi}_j$ dada por

$$\dot{\psi}_j = -\sum_{i=1}^{m} \psi^T \frac{\partial g_i}{\partial q_j} u_i \tag{20}$$

es posible llegar a la siguiente expresión diferencial

$$\dot{u}_i = \sum_{j=1}^{m} \psi^T [g_i, g_j] u_j \tag{21}$$

donde $[g_i, g_j]$ es el corchete de Lie entre dos campos vectoriales (ver ecuación 4). Este hecho establece que la norma de la entrada óptima es constante para $\tau \in [0, T]$ y puede expresarse como

$$\|u(\tau)\|^2 = \|u(0)\|^2 \tag{22}$$

Este resultado arroja información parcial, es decir, a partir de la ecuación 22 no es posible calcular las trayectorias óptimas.

4 Estrategias numéricas

Debido a la dificultad que representa encontrar la solución óptima para mover cualquier sistema *no holónomo*, la alternativa recae en la optimización numérica [Betts, 1998]. La primera parte de esta sección describe la transformación del problema de naturaleza variacional de dimensión infinita en uno de dimensión finita. Posteriormente describimos dos estrategias basadas en la programación no lineal. La primera estrategia, propuesta por [Fernandes, et al., 1994], consiste en resolver un problema no lineal sin restricciones empleando una aproximación del paso de Newton. La segunda estrategia está basada en la PCS con restricciones de igualdad.

4.1 Representación numérica del problema de optimización

Consideramos el sistema dinámico de la ecuación 3 junto con la funcional 15. Definimos una base ortonormal $\{\Theta(\tau)_j\}_{j=1}^{\infty}$ para $L_2[0, T]$ y suponemos que la señal de control $u \in [0, T]$ es una función continua C^1 por partes definida en $[0, T]$. De esta forma podemos escribir a u en términos de una serie de Fourier

$$u = \sum_{j=1}^{\infty} \left(\alpha_j \boldsymbol{e}^{i\frac{2j\pi t}{T}} + \beta_j \boldsymbol{e}^{-i\frac{2j\pi t}{T}} \right) \tag{23}$$

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

$$u(\tau) = \sum_{j=1}^{N} \alpha_j \Theta(\tau)_j \Rightarrow J \simeq \frac{1}{2} \sum_{j=1}^{N} |\alpha_j|^2 \tag{24}$$

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 original ahora se convierte en el siguiente: dados un tiempo final T y el par $(q_0, q_f) \in \mathbb{R}^n$, 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 $\frac{\partial f}{\partial \alpha} \in \mathbb{R}^{n \times N}$ necesaria durante la optimización. Podemos escribir el sistema 3 tal que $\forall (\tau, \alpha) \in [0, T] \times (-\infty, +\infty)$

$$\frac{\partial q}{\partial \tau}(\tau, \alpha) = \sum_{i=1}^{m} u_i(\tau, \alpha) g_i(q(\tau, \alpha)) \tag{25}$$

Diferenciando la ecuación 25 con respecto a α obtenemos

$$\frac{\partial^2 q}{\partial \tau \partial \alpha}(\tau, \alpha) = X(\tau, \alpha) Y(\tau, \alpha) + B(\tau, \alpha) \Theta(\tau) \tag{26}$$

donde $Y(\tau, \alpha) = \frac{\partial q}{\partial \alpha}(\tau, \alpha)$, $\Theta(\tau) = \frac{\partial u}{\partial \alpha}(\tau, \alpha)$, $X(\tau, \alpha)$ es una matriz de $n \times n$ de la forma

$$X(\tau, \alpha) = \sum_{i=1}^{m} \left(u_i(\tau, \alpha) \frac{\partial g_i}{\partial q}(q(\tau, \alpha)) \right) \tag{27}$$

y $B(\tau, \alpha)$ es una matriz de $n \times m$. Entonces, el sistema 26 es de hecho el sistema linealizado 3 en $\tau \to q(\tau, \alpha)$. Utilizando las ecuaciones anteriores, $f(\alpha)$ y $\frac{\partial f}{\partial \alpha}$ se obtienen evaluando las ecuaciones 25 y 26 en $\tau = T$ respectivamente.

4.2 Estrategia de optimización no lineal sin restricciones

En [Fernandes, et al., 1994] utilizan una variación del método de Newton para llevar al sistema hasta q_f. Primero le añaden un término adicional a la función de costo

$$J(\alpha) = \frac{1}{2}\left(\alpha^T\alpha + \gamma\left\|f(\alpha) - q_f\right\|^2\right) \tag{28}$$

donde $f(\alpha) = q(T)$ y γ es un parámetro que relaciona los términos del criterio 28 mediante una ponderación que el usuario debe manipular durante el proceso de optimización. Para minimizar la función no lineal $J(\alpha)$ con Newton, es necesario calcular el gradiente y la matriz Hessiana de $J(\alpha)$,

$$\frac{\partial J}{\partial \alpha} = \alpha + \gamma Y^T\left(f(\alpha) - q_f\right) \tag{29}$$

y

$$\frac{\partial^2 J}{\partial \alpha^2} = I + \gamma Y^T Y + \gamma \sum_{i=1}^{n}\left(f_i(\alpha) - q_i^f\right)H_i \tag{30}$$

donde

$$Y = \frac{\partial f}{\partial \alpha} \in \mathbb{R}^{n\times N} \quad \text{y} \quad H_i = \frac{\partial^2 f_i}{\partial \alpha^2} \quad i = 1, \dots, n \tag{31}$$

son la Jacobiana y las Hessianas de $f(\alpha)$. Suponemos que $J(\alpha)$ es una función escalar, continua y diferenciable al menos 2 veces. Las condiciones necesarias y suficientes para encontrar un mínimo local son

$$\frac{\partial J}{\partial \alpha} = 0 \quad \text{y} \quad \frac{\partial^2 J}{\partial \alpha^2} > 0 \tag{32}$$

Entonces las k iteraciones del algoritmo de optimización para actualizar α utilizando el enfoque de búsqueda lineal con un paso de Newton aproximado son de la forma

$$\alpha_{k+1} = \alpha_k - \mu\left[\rho I + Y^T Y\right]^{-1}\left[\rho\alpha_k + Y^T\left(f(\alpha_k) - q_f\right)\right] \tag{33}$$

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 $J(\alpha)$ y evita el cálculo de las Hessianas H_i de $f(\alpha)$. De esta forma resolvemos un problema de optimización no lineal sin restricciones. La convergencia hacia la solución exacta depende de N, μ y γ.

4.3 Estrategia de optimización no lineal con restricciones

En la estrategia anterior es necesario incluir un segundo 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 μ y γ 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 28 de tal forma que la función de costo sea convexa

$$J(\alpha) = \frac{1}{2}\alpha^T\alpha \tag{34}$$

y agregar al problema una restricción de igualdad

$$c(\alpha) = \frac{1}{2}\left\|f(\alpha) - q_f\right\|^2 \tag{35}$$

Entonces, el problema posee la siguiente forma

$$\mathcal{L}(\alpha, \lambda) = J(\alpha) - \sum_{i\epsilon I}\lambda_i c_i(\alpha) \tag{36}$$

donde $\mathcal{L}$ es la lagrangiana del problema, λ_i son los multiplicadores de Lagrange y $c_i(\alpha)$ son las restricciones. En este caso, las condiciones necesarias de primer orden KKT para encontrar una solución óptima local satisfacen el siguiente sistema

$$\begin{pmatrix} \frac{\partial J}{\partial \alpha}(\alpha) - \left(\frac{\partial c}{\partial \alpha}(\alpha)\right)^T \lambda \\ c(\alpha) \end{pmatrix} = 0 \tag{37}$$

La Jacobiana del sistema 37 con respecto a α y λ está dada por

$$\begin{pmatrix} \frac{\partial^2 \mathcal{L}}{\partial \alpha^2}(\alpha,\lambda) & -\left(\frac{\partial c}{\partial \alpha}(\alpha)\right)^T \\ \frac{\partial c}{\partial \alpha}(\alpha) & 0 \end{pmatrix} \tag{38}$$

Utilizando la PCS, suponemos que la Jacobiana de la restricción $\frac{\partial c}{\partial \alpha}(\alpha)$ es de rango completo con respecto a sus renglones y que $\frac{\partial^2 \mathcal{L}}{\partial \alpha^2}(\alpha,\lambda) > 0$ en el espacio tangente de las restricciones. El paso de Newton está dado por

$$\begin{pmatrix} \alpha_{k+1} \\ \lambda_{k+1} \end{pmatrix} = \begin{pmatrix} \alpha_k \\ \lambda_k \end{pmatrix} + \begin{pmatrix} p_k \\ p_\lambda \end{pmatrix} \tag{39}$$

donde p_ky p_λ resuelven el sistema KKT-Newton

$$\begin{pmatrix} \left(\frac{\partial^2 \mathcal{L}}{\partial \alpha^2}\right)_k & -\left(\frac{\partial c}{\partial \alpha}\right)_k^T \\ \left(\frac{\partial c}{\partial \alpha}\right)_k & 0 \end{pmatrix} \begin{pmatrix} p_k \\ p_\lambda \end{pmatrix} = \begin{pmatrix} -\left(\frac{\partial J}{\partial \alpha}\right)_k + \left(\frac{\partial c}{\partial \alpha}\right)_k^T \lambda_k \\ -c_k \end{pmatrix} \tag{40}$$

Conociendo la dirección entonces calculamos el tamaño de paso μ. Debido a que el problema cuenta con restricciones, es necesario definir una función de mérito para garantizar la convergencia hacia un mínimo siempre y cuando no se violen las restricciones. La función que utilizamos es

$$\phi(\alpha,\delta) = J(\alpha) + \delta\|c(\alpha)\| \tag{41}$$

donde δ es un escalar positivo que debe ajustarse tal que p_k sea una dirección de descenso para ϕ. Esto es posible si $p_k \neq 0$, $\left(\frac{\partial^2 \mathcal{L}}{\partial \alpha^2}\right)_k$ es positive definida y $\delta > \|\lambda_{k+1}\|_\infty$. Si el valor de δ es muy pequeño entonces el algoritmo se puede alejar de la solución. En el caso opuesto, el tiempo de convergencia aumenta. En PCS con restricciones de igualdad utilizando búsqueda lineal, un paso μp_k adecuado debe satisfacer la siguiente condición de descenso suficiente:

$$\phi(\alpha_k + \mu p_k, \delta_k) \leq \phi(\alpha_k, \delta_k) + \eta\mu D(\phi(\alpha_k,\delta_k), p_k) \tag{42}$$

donde $\eta \in (0,\frac{1}{2}), \delta_\kappa = \| \lambda_\kappa \|_\infty$ y $D = p_k^T\left(\left(\frac{\partial J}{\partial \alpha}(\alpha)\right)_k - \left(\frac{\partial c}{\partial \alpha}(\alpha)\right)_k^T \lambda_k\right) - \delta_k\|c(\alpha)_k\|$ es la derivada direccional de ϕ en la dirección p_k. Reemplazamos el cálculo explícito de la matriz $\frac{\partial^2 \mathcal{L}}{\partial \alpha^2}(\alpha,\lambda)$ mediante la fórmula estándar conocida como BFGS [Nocedal y Wright, 2000] para construir una matriz $B_k \approx \left(\frac{\partial^2 \mathcal{L}}{\partial \alpha^2}\right)_k$. Otro ingrediente importante es la evaluación de la condición de *curvatura* antes de actualizar a B_{k+1}dada por

$$\begin{aligned} s_k^T y_k &> 0 \\ s_k &= \alpha_{k+1} - \alpha_k \\ y_k &= \frac{\partial \mathcal{L}}{\partial \alpha}(\alpha_{k+1},\lambda_{k+1}) - \frac{\partial \mathcal{L}}{\partial \alpha}(\alpha_k,\lambda_k) \end{aligned} \tag{43}$$

La estrategia numérica que proponemos permite generar trayectorias eficientemente para sistemas *no holónomos* sin deriva (ver Algoritmo 1).

5 Generación de trayectorias

Los métodos de la sección 4 los implementamos inicialmente en *MATLAB* versión 7.5. Realizamos comparaciones entre las estrategias numéricas sin restricciones (ver sección 4.2), con restricciones (ver sección 4.3) y la función *fmincon* que proporciona *MATLAB* para la optimización no lineal con restricciones. Posteriormente realizamos la implementación en lenguaje *ANSI C*. Todas las trayectorias que mostramos se generaron en una PC con el sistema operativo *Linux Fedora 10*. El procesador que usamos fue un *Intel Core2 Duo* con 1.5GHz en cada núcleo. Utilizamos *BLAS* para calcular las operaciones simples de álgebra lineal y *LAPACK* para resolver operaciones complejas como la inversión de matrices. Estas bibliotecas dinámicas están disponible gratuitamente en el repositorio: *http://www.netlib.org/*. Para graficar las trayectorias utilizamos funciones simples de *OpenGL*.

Entrada: 1. Configuraciones inicial y final: $q_0, q_f \in \mathbb{R}^n$
2. Una distribución $\Delta(q) \in \mathbb{R}^{n\times m}$

Salida: La señal de control óptima $u(\tau) \in \mathbb{R}^m, \tau \in [0,T]$, que lleva al sistema de q_0 a q_f

Inicio

Paso 0: Escoger una base ortonormal $(\mathcal{e}(\tau)_1\ \mathcal{e}(\tau)_2 \ldots \mathcal{e}(\tau)_N)$;

Paso 1: Iniciar con $\alpha_0 \neq 0$ mediante algún proceso aleatorio y $\lambda_0 = 1$;
Paso 2: Escoger $\eta \in \left(0, \frac{1}{2}\right)$ y $\rho \in (0,1)$;
Paso 3: Calcular p_k con 40 ;
Paso 4: Escoger $\delta_k > \|\lambda_{k+1}\|_\infty$;
Paso 5: Asignar $\mu = 1$;
Mientras la condición 42 no se cumpla ***entonces***
asignar $\mu = \rho_\mu \mu$ para algún $\rho_\mu \in (0, \rho)$;
Paso 7: Asignar $\alpha_{k+1} = \alpha_k + \mu p_k$ y $\lambda_{k+1} = \lambda_k + \mu p_\lambda$;
Paso 8: Evaluar a $J_{k+1}, \left(\frac{\partial J}{\partial \alpha}\right)_{k+1}, c_{k+1}, \left(\frac{\partial c}{\partial \alpha}\right)_{k+1}$;
Si la condición 43 se cumple ***entonces***
Obtener la matriz B_{k+1} actualizando a B_k mediante BFGS ;
Si $TOL < \|c\|$ ***o*** se cumple en número máximo de iteraciones ***entonces***
Terminar ;
De lo contrario
regresar al ***Paso 3*** ;
Fin

Algoritmo 1: estrategia numérica basada en PCS para calcular trayectorias de sistemas sin deriva y *no holónomos*

Una consideración importante es el análisis de *controlabilidad* de los sistemas (ver sección 2.2). Aplicando LARC [Sussmann, 1990] verificamos que todos los sistemas fueran controlables, esto quiere decir que podemos encontrar trayectorias para llevar a dichos sistemas de cualquier configuración inicial a cualquier otra final (ver [Laumond, et al., 1998]). Las condiciones necesarias para encontrar trayectorias óptimas también han sido estudiadas y verificadas [Cesari, 1983]. Si deseamos posteriormente utilizar los algoritmos numéricos de la sección 4 para otros sistemas sin deriva que no estén descritos en este trabajo (ver sección 5.1), entonces es necesario aplicar previamente el análisis descrito en la sección 2 y las condiciones necesarias sobre la existencia de trayectorias óptimas. De otra manera, no será posible garantizar la convergencia de los algoritmos.

5.1 Modelos cinemáticos y dinámicos

Para probar los algoritmos de la sección 4, utilizamos varios sistemas diferenciales de primer orden. Entre ellos se encuentra el monociclo (ver ecuación 10). También consideramos algunas variantes:

$$\begin{pmatrix} \dot{q}_1 \\ \dot{q}_2 \\ \dot{q}_3 \\ \dot{q}_4 \end{pmatrix} = \begin{pmatrix} \cos q_3 \\ \sin q_3 \\ q_4 \\ 0 \end{pmatrix} u_1 + \begin{pmatrix} 0 \\ 0 \\ 0 \\ 1 \end{pmatrix} u_2 \tag{44}$$

En este caso, el $CS = \mathbb{R}^2 \times \mathbb{S}^1 \times \mathbb{R}$, u_1 sigue siendo la velocidad lineal y u_2 ahora es la derivada de la curvatura con respecto al tiempo. Aplicando las herramientas de la sección 2, es posible mostrar que el sistema 44 es controlable. Esto es, el álgebra de Lie $LA(g_i)$ generada por g_1 y g_2 es de dimensión 4 en cada q:

$$\begin{aligned} g_1 = \begin{pmatrix} \cos q_3 \\ \sin q_3 \\ q_4 \\ 0 \end{pmatrix};\ g_2 = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 1 \end{pmatrix};\ [g_2, g_1] \\ = \begin{pmatrix} 0 \\ 0 \\ 1 \\ 0 \end{pmatrix};\ [[g_2, g_1], g_1] \\ = \begin{pmatrix} -\sin q_3 \\ \cos q_3 \\ 0 \\ 0 \end{pmatrix} \end{aligned} \tag{45}$$

y $\det\left(g_1\ g_2\ [g_2, g_1]\ [[g_2, g_1], g_1]\right) = -1$. También utilizamos el modelo cinemático del automóvil

$$\begin{pmatrix} \dot{q}_1 \\ \dot{q}_2 \\ \dot{q}_3 \\ \dot{q}_4 \end{pmatrix} = \begin{pmatrix} \cos q_3 \\ \sin q_3 \\ (\tan q_4)/L \\ 0 \end{pmatrix} u_1 + \begin{pmatrix} 0 \\ 0 \\ 0 \\ 1 \end{pmatrix} u_2 \tag{46}$$

donde L representa la distancia entre el eje trasero y delantero, (q_1, q_2) definen la posición del artefacto situada en el centro del eje trasero, q_3 es la dirección del automóvil con respecto al eje horizontal y q_4 es la dirección instantánea definida por el movimiento de las llantas delanteras, u_1 es la velocidad lineal caracterizada por el pedal y u_2 la razón de cambio del ángulo de giro del volante con respecto al tiempo. Este sistema es controlable:

$$\begin{aligned} g_1 = \begin{pmatrix} \cos q_3 \\ \sin q_3 \\ (\tan q_4)/L \\ 0 \end{pmatrix};\ g_2 = \begin{pmatrix} 0 \\ 0 \\ 0 \\ 1 \end{pmatrix};\ [g_1, g_2] \\ = \begin{pmatrix} 0 \\ 0 \\ -1/(L\cos^2 q_4) \\ 0 \end{pmatrix};\ [g_1, [g_1, g_2]] \\ = \begin{pmatrix} -\sin q_3/(L\cos^2 q_4) \\ \cos q_3/(L\cos^2 q_4) \\ 0 \\ 0 \end{pmatrix} \end{aligned} \tag{47}$$

y $\text{rango}\big(g_1\ g_2\ [g_1,g_2]\ [g_1,[g_1,g_2]]\big) = 4$ en cada q siempre que q_4 sea diferente de $\pm\pi/2$. Otro sistema similar que probamos está compuesto únicamente por dos ruedas y un eje:

$$\begin{pmatrix}\dot{q}_1\\ \dot{q}_2\\ \dot{q}_3\end{pmatrix} = \begin{pmatrix}\cos q_3/2\\ \sin q_3/2\\ 1/L\end{pmatrix}u_1 + \begin{pmatrix}\cos q_3/2\\ \sin q_3/2\\ -(1/L)\end{pmatrix}u_2 \tag{48}$$

donde (q_1, q_2) definen la posición del móvil situada en el centro del eje, L es una constante que representa la distancia entre las ruedas o la longitud del eje, q_3 es la dirección del artefacto con respecto al eje horizontal, u_1 y u_2 representan las velocidades angulares de cada rueda. El sistema 48 es controlable:

$$g_1 = \begin{pmatrix}\cos q_3/2\\ \sin q_3/2\\ 1/L\end{pmatrix};\ g_2 = \begin{pmatrix}\cos q_3/2\\ \sin q_3/2\\ 1/L\end{pmatrix};\ [g_1,g_2] = \begin{pmatrix}\sin q_3/L\\ -\cos q_3/L\\ 0\end{pmatrix} \tag{49}$$

y $\text{rango}(g_1\ g_2\ [g_1,g_2]) = 3$ en cada q. El último sistema cinemático que utilizamos está compuesto por 3 cuerpos articulados sujetos a la restricción 9. El primer cuerpo es justamente el monociclo. Su posición está determinada por $(q_1, q_2) \in \mathbb{R}^2$ y su dirección está definida mediante el ángulo $q_3 \in \mathbb{S}^1$ existente entre el monociclo y el eje horizontal. Los otros cuerpos representan vagones unidos por una articulación. La orientación del primer vagón con respecto al monociclo está definida por q_4 y la orientación del segundo vagón con respecto al primero está dada por q_5. L_1 y L_2 representan las distancias entre el monociclo y el primer vagón y éste con respecto al segundo respectivamente. El espacio de control es de dimensión 2 y el $CS = \mathbb{R}^2 \times (\mathbb{S}^1)^3$:

$$\begin{pmatrix}\dot{q}_1\\ \dot{q}_2\\ \dot{q}_3\\ \dot{q}_4\\ \dot{q}_5\end{pmatrix} = \begin{pmatrix}\cos q_3\\ \sin q_3\\ 0\\ \sin q_4/L_1\\ -\sin q_4/L_1 + (\sin q_5 \cos q_4)/L_2\end{pmatrix}u_1 + \begin{pmatrix}0\\0\\1\\1\\0\end{pmatrix}u_2 \tag{50}$$

donde $(u_1, u_2) \in \mathbb{R}^2$ representan la velocidad lineal y angular del monociclo respectivamente. Aplicando LARC verificamos que el sistema 50 es controlable. De hecho, la familia de campos vectoriales

$g_1, g_2, [g_1,g_2], [g_1,[g_1,g_2]], \big[g_1,[g_1,[g_1,g_2]]\big]$ *generan* el espacio tangente en cada q siempre que q_4 sea diferente de $\pm\pi/2$. Suponiendo que $L_1 = L_2 = 1$, entonces

$$\begin{aligned}
g_1 &= \begin{pmatrix}\cos q_3\\ \sin q_3\\ 0\\ \sin q_4\\ -\sin q_4 + \sin q_5 \cos q_4\end{pmatrix};\\
g_2 &= \begin{pmatrix}0\\0\\1\\1\\0\end{pmatrix};\quad [g_1,g_2] = \begin{pmatrix}\sin q_3\\ -\cos q_3\\ 0\\ \cos q_4\\ -\cos q_4 - \sin q_5 \sin q_4\end{pmatrix};\\
[g_1,[g_1,g_2]] &= \begin{pmatrix}0\\0\\0\\1\\-1-\cos q_5\end{pmatrix};\\
\big[g_1,[g_1,[g_1,g_2]]\big] &= \begin{pmatrix}0\\0\\0\\ \cos q_4\\ -2\cos q_4 - \cos q_4 \cos q_5\end{pmatrix}
\end{aligned} \tag{51}$$

Los sistemas 10, 44, 46, 48 y 50 están sujetos a la restricción 9. En otras palabras, la dirección de cada cuerpo debe corresponder con la dirección tangente a la trayectoria que están efectuando. Otra restricción *no holónoma* existente en sistemas de segundo orden es la conservación del momento angular. Este tipo de restricción aparece en aplicaciones con manipuladores espaciales principalmente. Aunque la causa es distinta a la de la restricción 9, la estructura del sistema resultante es semejante y, por lo tanto, los algoritmos de la sección 4 también sirven para generar trayectorias sin efectuarles modificación alguna. Consideramos el modelo simplificado de un robot espacial en el plano compuesto por 3 cuerpos unidos mediante articulaciones [Murray, et al., 1994]. El cuerpo central está unido a dos brazos rígidos con masa m en los extremos. La orientación del cuerpo central con respecto al eje horizontal es q_1. Las articulaciones están situadas a una distancia L_1 con respecto al centro del cuerpo principal, y la longitud de los brazos es L_2. La orientación de cada brazo con respecto al cuerpo central es q_2 y q_3. Si no consideramos el efecto de la gravedad y la fricción es mínima, el movimiento de los brazos causa que la orientación del cuerpo principal se modifique. Entonces, si no existen fuerzas externas y considerando la conservación del momento

angular que inicialmente la definimos en cero, la siguiente restricción implícita aparece:

$$\begin{aligned}(I + 2mL_2^2 + 2mL_1^2 \\ + 2mL_1 L_2(\cos q_2 \\ + \cos q_3))\dot{q}_1 \\ + (mL_2^2 + mL_1 \cos q_2)\dot{q}_2 \\ + (mL_2^2 + mL_1 \cos q_3)\dot{q}_3 = 0\end{aligned} \tag{52}$$

El siguiente sistema de control es una representación paramétrica de la restricción 52:

$$\begin{pmatrix}\dot{q}_1 \\ \dot{q}_2 \\ \dot{q}_3\end{pmatrix} = \begin{pmatrix} -\dfrac{(mL_2^2 + mL_1 \cos q_2)}{(I + 2mL_2^2 + 2mL_1^2 + 2mL_1 L_2(\cos q_2 + \cos q_3))} \\ 1 \\ 0 \end{pmatrix} u_1 + \begin{pmatrix} -\dfrac{(mL_2^2 + mL_1 \cos q_3)}{(I + 2mL_2^2 + 2mL_1^2 + 2mL_1 L_2(\cos q_2 + \cos q_3))} \\ 0 \\ 1 \end{pmatrix} u_2 \tag{53}$$

donde $u_1 = \dot{q}_2$, $u_2 = \dot{q}_3$ y el $CS = (\mathbb{S}^1)^3$. Este sistema es controlable porque el espacio tangente puede generarse mediante $g_1, g_2, [g_1, g_2]$ si $q_2 \neq q_3$.

5.3 Sensibilidad en los parámetros de entrada

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

- proporcionar una solución inicial α_0.
- las configuraciones $q_0, q_f \in \mathbb{R}^n$.
- La dimensión del vector α.
- El tiempo final deseado T.
- La distribución diferencial $\Delta(q)$.

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 1). De lo contrario, si α_0 está considerablemente alejada de la solución óptima, entonces, el algoritmo de la sección 4.2 no logrará converger adecuadamente a menos que en cada iteración modifiquemos inteligentemente la longitud del 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., 1994] y el propuesto por nosotros respectivamente. Visualmente la diferencia no es relevante entre las trayectorias. Sin embargo, el algoritmo de [Fernandes, et al., 1994] tardó 24 iteraciones para 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 la sección 4.2 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 2 mostramos un ejemplo con el sistema 44 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., 1994] y con PCS respectivamente. El primer algoritmo 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. Esto fue detectado automáticamente gracias a la condición de *curvatura*.

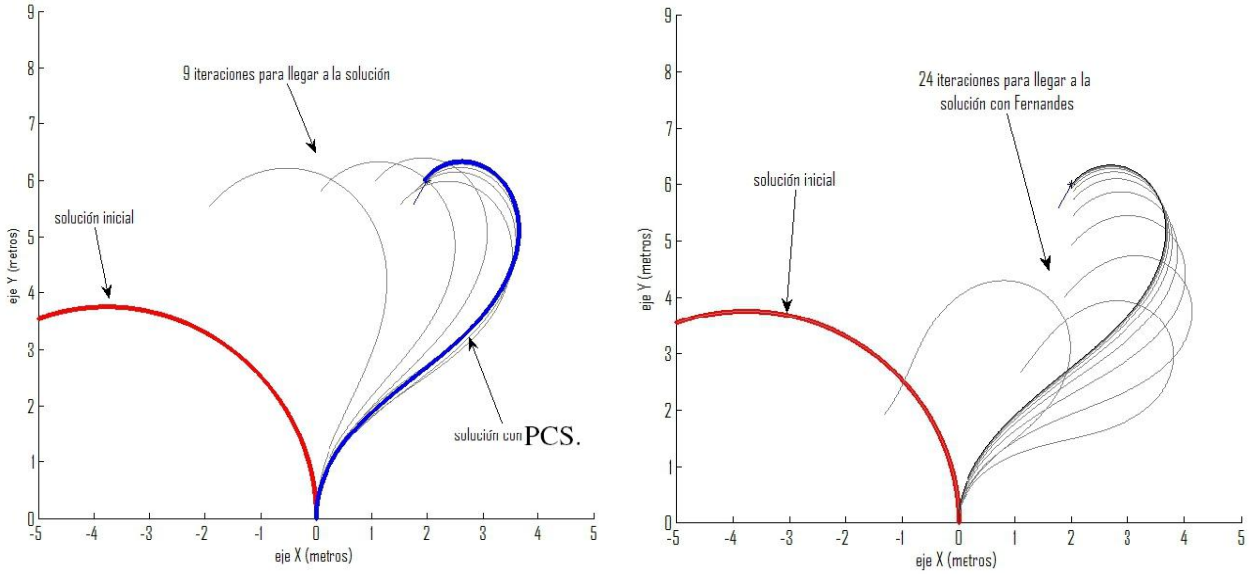

Fig.1. Ejemplo representativo del número de iteraciones en la optimización de trayectorias. Las trayectorias en rojo representan la integración del sistema 44 utilizando la solución inicial α_0. Las trayectorias en gris muestran las trayectorias correspondientes a α_k, donde k representa la iteración. Las trayectorias en negro y azul representan las trayectorias resultantes con α^* utilizando los métodos de la sección 4.2 y 4.3 respectivamente. El método de PCS necesitó 9 iteraciones para llevar al sistema 44 de q_0 a q_f minimizando la norma L_2 del control. El método de [Fernandes, et al., 1994] requirió 24 iteraciones

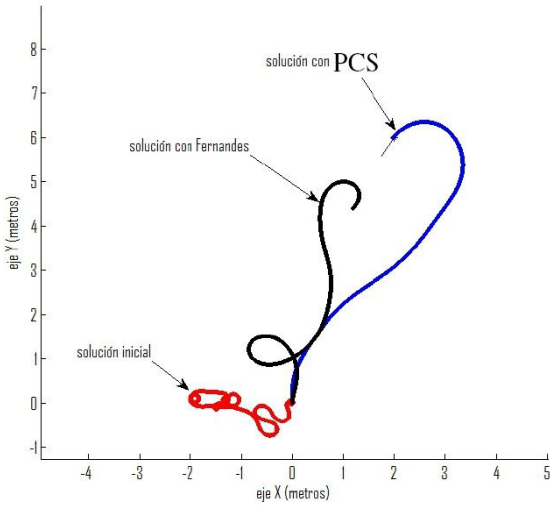

Fig. 2. Caso representativo de la generación de trayectorias cuando la solución inicial α_0 está alejada de la solución deseada α^*. El sistema que utilizamos fue el 44. En rojo mostramos la trayectoria integrada con α_0. La trayectoria en negro es la solución que logramos obtener con el método de la sección 4.2. En azul mostramos el resultado con la estrategia de la sección 4.3

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.

5.4 Convergencia y rendimiento

En la Figura 3 presentamos un ejemplo representativo que ilustra el resultado obtenido con la función *fmincon*. Evidentemente los resultados no son satisfactorios aún cuando las trayectorias son relativamente simples. En este caso, utilizamos el sistema del monociclo. La solución obtenida con los algoritmos de la sección 4 es semejante. En rojo mostramos la solución obtenida con la función *fmincon*. En la Figura 4 mostramos algunos ejemplos cuando la posición y la dirección final cambian.

La Figura 5 muestra la malla que definimos para generar de forma exhaustiva 90000 trayectorias óptimas en *MATLAB* utilizando los métodos descritos en la sección 4. Cada vértice de la malla representa una configuración final en posición y orientación. La configuración inicial siempre es la misma y está definida en el origen. Utilizando esta representación del espacio accesible, generamos 36 planos con respecto a la orientación final definida por el eje vertical. Esto es, en cada plano la orientación final es la misma para cualquier posición. En cada plano generamos 2500 trayectorias óptimas para unir el origen (configuración inicial) con cada vértice (configuración final). Este proceso lo realizamos con el sistema 10 y 50 aplicando el método en [Fernandes et al., 1994] y el Algoritmo 1 basado en PCS. Varios aspectos fueron necesarios:

- La solución inicial la generemos utilizando las curvas de Dubins (ver [Dubins, 1957]). De la ruta y los controles correspondientes calculamos la solución inicial α_0.
- El tiempo final lo definimos *a priori* considerando una velocidad constante de 1.3 m/s para Dubins. Utilizamos el mismo tiempo final para los métodos numéricos.
- La frecuencia de muestreo fue de 120 Hz. Por lo tanto, las trayectorias generadas contienen secuencias de 120 puntos por segundo.

Las Figuras 6 y 7 ilustran la comparación que realizamos después de aplicar el procedimiento anterior. De esta forma pudimos definir cotas con respecto a la convergencia y el rendimiento de los algoritmos. La cota inferior está definida utilizando el sistema 10 ya que representa el más sencillo que consideramos en este trabajo mientras que la cota superior la define el sistema 50 dado que es uno de los sistemas complicados que también consideramos.

Finalmente, 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 mejoran considerablemente usando *ANSI C* como era de esperarse. Para todos los sistemas de la sección 5.1, los siguientes rangos se cumplen:

1. PCS, Algoritmo 1: varía de 0.8 a 1.2 segundos.
2. [Fernandes, et al., 1994]: varía de 1.8 a 2.7 segundos.

6 Conclusiones

En este trabajo presentamos 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, recurrimos a los métodos de optimización numérica. Proponemos resolver el problema mediante la programación no lineal con restricciones de igualdad utilizando un algoritmo simple basado en la PCS con búsqueda lineal. Implementamos la estrategia en lenguaje *ANSI C* para que sea sencilla la implementación en plataformas robóticas. Desde los años ochenta existen trabajos reportados en planificación de movimientos para sistemas *no holónomos*. Actualmente este eje de investigación continua siendo vigente. De hecho, existen estudios recientes sobre las trayectorias de marcha humana y humanoide basados en este tipo de sistemas [Arechavaleta, et al., 2008], [Mombaur, et al.,

2008], [Hayet, et al., 2009]. Estamos trabajando actualmente en la adaptación de la estrategia numérica que proponemos para considerar sistemas con deriva y redundantes sujetos a restricciones diferenciales *holónomas* y *no holónomas*. También estamos extendiendo estos métodos para considerar restricciones unilaterales.

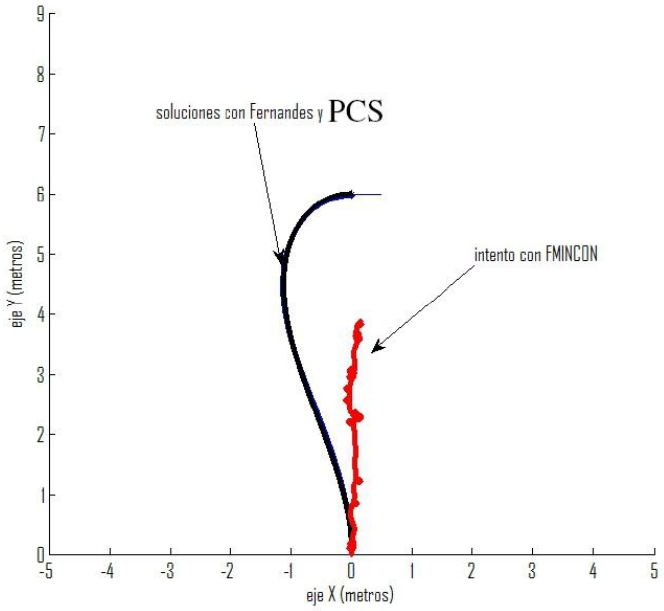

Fig. 3. Comparación entre los tres métodos de optimización numérica: en rojo mostramos el resultado obtenido con la función fmincon que proporciona MATLAB. En negro y azul mostramos las trayectorias generadas con los algoritmos de las secciones 4.2 y 4.3 respectivamente

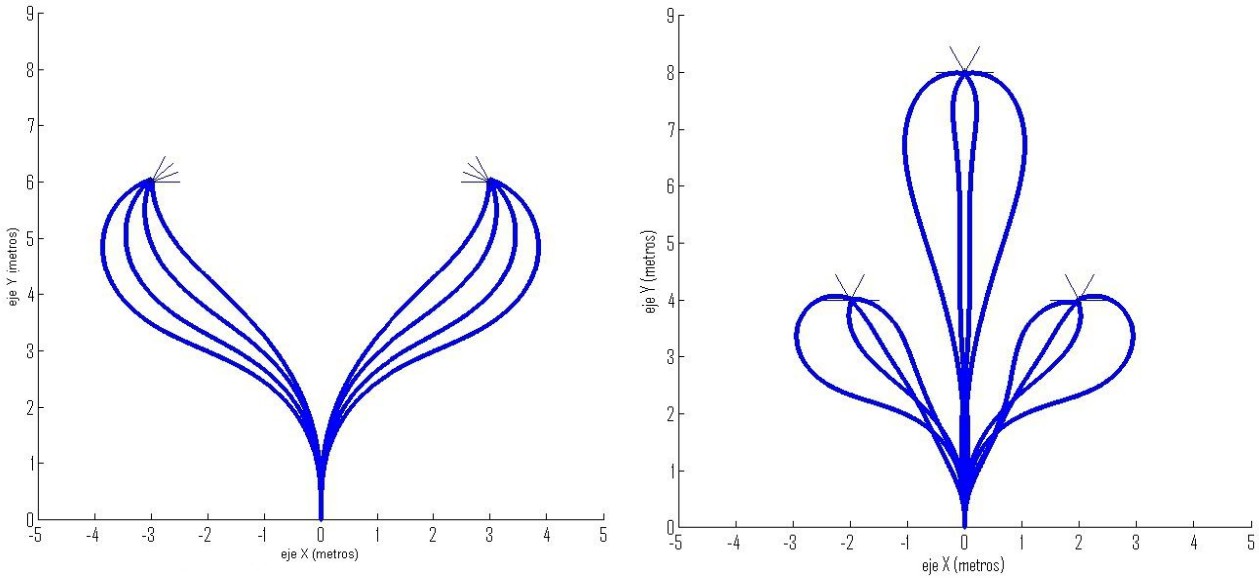

Fig. 4. Simetría de las trayectorias cuando la configuración final varía

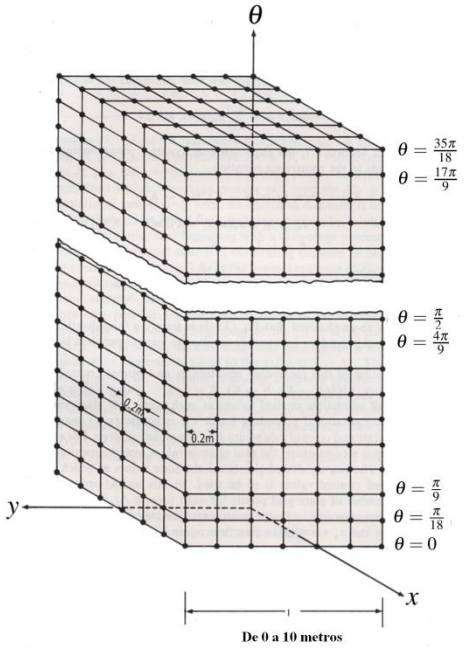

Fig. 5. Comparación exahustiva entre los métodos de optimización numérica con y sin restricciones variando la configuración final deseada en posición y orientación y utilizando un sistema no holónomo relativamente sencillo y otro complicado en términos de las restricciones diferenciales a nivel cinemático y de acuerdo al número de g.d.l. de dichos sistemas. Generamos una malla en 3 dimensiones con una resolución de $\left(0.2\mathrm{m} \times 0.2\mathrm{m} \times \frac{\pi}{18}\right)$ en un rango de $[0,10] \times [0,10]$ en posición (metros). El eje vertical define las orientaciones (radianes). La configuración inicial siempre es la misma y está en el origen (0,0,0)

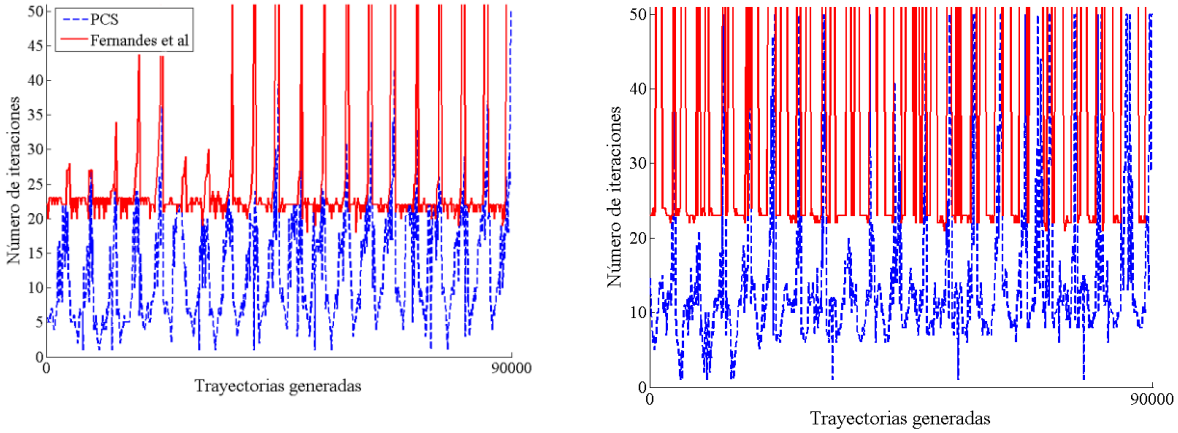

Fig. 6. Convergencia de los algoritmos numéricos utilizando la malla de la Figura 5. En rojo y azul mostramos las trayectorias generadas con los algoritmos de las secciones 4.2 y 4.3 respectivamente. La gráfica izquierda corresponde al sistema 10 y la derecha al sistema 50. Los puntos máximos normalmente corresponden a orientaciones finales complicadas y por lo mismo a trayectorias con formas complicadas

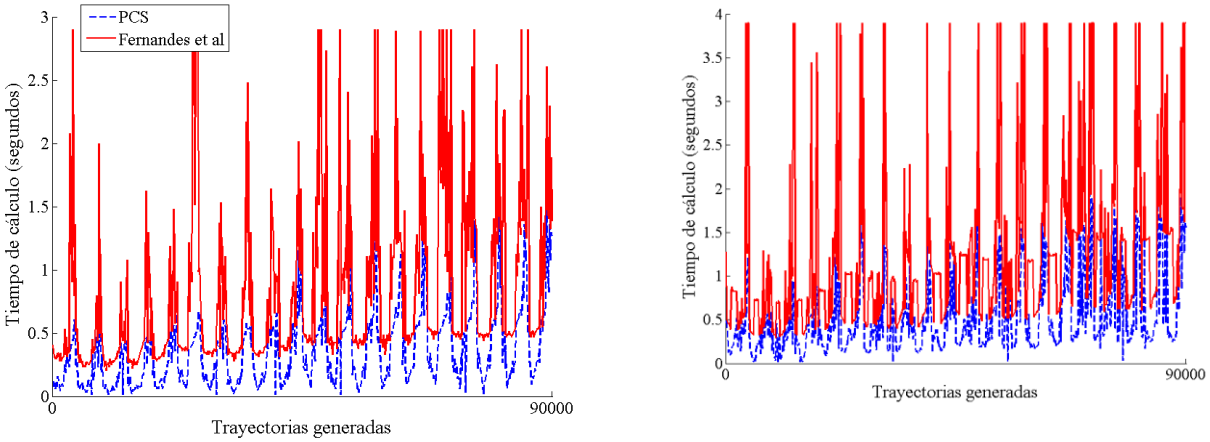

Fig. 7. Rendimiento de los algoritmos numéricos utilizando la malla de la Figura 5. En rojo y azul mostramos las trayectorias generadas con los algoritmos de las secciones 4.2 y 4.3 respectivamente. La gráfica izquierda corresponde al sistema 10 y la derecha al sistema 50

Agradecimientos

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

Referencias

1. **Arechavaleta, G., Laumond, J.P., Hicheur, H.** & **Berthoz, A. (2008).** An Optimality Principle Governing Human Walking. *IEEE Transactions on Robotics*, 24(1), 5-14.
2. **Balkcom, D.J., Kavathekar, P.A.** & **Mason, M.T. (2008).** The Minimum-Time Trajectories for an Omni-Directional Vehicle. In S. Akella, N.M. Amato, W.H. Huang & B. Mishra (Eds.), *Algorithmic Foundation of Robotics VII* (343-358). Berlin: Springer.
3. **Balkcom, D.J.** & **Mason, M.T. (2002).** Time Optimal Trajectories for Bounded Velocity Differential Drive Vehicles. *International Journal of Robotics Research*, 21(3), 199-217.
4. **Bhattacharya, S., Murrieta-Cid, R.** & **Hutchinson, S. (2007).** Optimal Paths for Landmark-based Navigation by Differential Drive Vehicles with Field-of-View Constraints. *IEEE Transactions on Robotics*, 23(1), 47-59.
5. **Bellaiche, A.** & **Risler, J. J. (1996).** *Sub-Riemannian Geometry*. Basel; Boston: Birkhäuser.
6. **Betts, J.T. (1998).** Survey of Numerical Methods for Trajectory Optimization. *Journal of Guidance, Control, and Dynamics*, 21(2), 193-207.
7. **Boissonnat, J.D., Cerezo A. & Leblong, J. (1992).** Shortest paths of bounded curvature in the plane. *IEEE International Conference on Robotics and Automation*, Nice, France, 3, 2315-2320.
8. **Boissonnat, J.D., Cerezo, A.** & **Leblong, J. (1994).** *A note on shortest paths in the plane subject to a constraint on the derivative of the curvature* (Rapport de recherche n 2160). Nice: Institut National De Recherche En Informatique Et Automatique.
9. **Brockett, R.W. (1976).** Nonlinear Systems and Differential Geometry. *Proceedings of the IEEE*, 64(1), 61-72.
10. **Cesari, L. (1983).** *Optimization, theory and applications: Problems with ordinary differential equations*. New York: Springer-Verlag.
11. **Choset, H., Lynch, K.M., Hutchinson, S., Kantor, G., Burgard, W., Kavraki, L.E.** & **Thrun, S. (2005).** *Principles of robot motion: Theory, algorithms and implementations*. Boston: MIT Press.
12. **Divelbiss, A.W.** & **Wen, J.T. (1997).** A path space approach to nonholonomic motion planning in the presence of obstacles. *IEEE Transactions on Robotics and Automation*, 13(3), 443-451.
13. **Dubins, L.E. (1957).** On curves of minimal length with a constraint on average curvature and with prescribed initial and terminal positions and tangents. *American Journal of Mathematics*, 79(3), 497-516.
14. **Fernandes, C., Gurvits, L.** & **Li Z. (1994).** Near-optimal nonholonomic motion planning for a system of coupled rigid bodies. *IEEE Transactions on Automatic Control*, 39(3), 450-463.
15. **Hayet, J.B., Esteves, C., Arechavaleta, G.** & **Yoshida, E. (2009).** Motion Planning for a Vigilant Humanoid Robot. 9th *IEEE-RAS International Conference on Humanoid Robots*, Paris, France, 196-201.
16. **Hayet, J.B., Esteves, C.** & **Murrieta-Cid, R. (2010).** A Motion Planner for Maintaining Landmark Visibility with a Differential Drive Robot. In G.S. Chirikjian, H. Choset, M. Morales & T. Murphey (Eds.), *Algorithmic Foundation of Robotics VIII* (333-347). Berlin: Springer.
17. **Howard, T.** & **Kelly, A. (2007).** Optimal Rough Terrain Trajectory Generation for Wheeled Mobile Robots. *International Journal of Robotics Research,* 26(2), 141-166.
18. **Huifang, W., Yangzhou, C.** & **Souères P. (2009).** A Geometric Algorithm to Compute Time-Optimal Trajectories for a Bidirectional Steered Robot. *IEEE Transactions on Robotics*, 25(2), 399-413.
19. **Kostov, V.P.** & **Degtiariova-Kostova, E.V. (1995).** The planar motion with bounded derivative of the curvature and its suboptimal paths. *Acta Mathematica Universitatis Comeianae*, 64(2), 185-226.
20. **Lamiraux, F., Bonnafous, D.** & **Lefebvre, O. (2004).** Reactive path deformation for nonholonomic mobile robots. *IEEE Transactions on Robotics*, 20(6), 967-977.
21. **Latombe J.C. (1991).** *Robot Motion Planning*. Boston: Kluwer Academic Publishers.
22. **Laumond, J.P., Sekhavat, S.** & **Lamiraux, F. (1998).** Guidelines in Nonholonomic Motion Planning for Mobile Robots. *Robot Motion Planning and Control*. *Lecture Notes in Control and Information Sciences, 229, 1-53*.
23. **LaValle, S.M. (2006).** *Planning Algorithms*, Cambridge; New York: Cambridge University Press.
24. **Li, Z.** & **Canny, J. F. (1993).** *Nonholonomic Motion Planning*. Boston: Kluwer Academic.
25. **Mombaur, K., Laumond, J.P.** & **Yoshida, E. (2008).** An optimal control model unifying holonomic and nonholonomic walking. 8th *IEEE-RAS International Conference on Humanoid Robots*, Daejeon, Korea (South), 646-653.
26. **Murray, R.M, Li, Z.** & **Sastry S.S. (1994).** *A Mathematical Introduction to Robotic Manipulation*. Boca Raton: CRC Press.
27. **Nocedal, J.** & **Wright, S. J. (1999).** *Numerical Optimization*. New York: Springer.
28. **Ostrowski, J.P., Desai, J.P.** & **Kumar, V. (2000).** Optimal Gait Selection for Nonholonomic Locomotion Systems. *International Journal of Robotics Research*, 19(3), 225-237.
29. **Pecsvaradi, T. (1972).** Optimal Horizontal Guidance Law for Aircraft in the Terminal Area. *IEEE Transactions on Automatic Control*, 17(6), 763-772.
30. **Pontryagin, L.S., Boltyanskii, V.G., Gamkrelidze, R.V.** & **Mishchenko E.F. (1964).** *The Mathematical Theory of Optimal Processes*. Oxford, New York: Pergamon Press.
31. **Reeds, J.A.** & **Shepp, L.A. (1990).** Optimal paths for a car that goes both forward and backwards. *Pacific Journal of Mathematics*, 145(2), 367-393.
32. **Salaris, P., Fontanelli, D., Pallottino, L.** & **Bicchi, A. (2010).** Shortest Paths for a Robot With Nonholonomic

and Field-of-View Constraints. *IEEE Transactions on Robotics,* 26(2), 269-281.

33. **Sastry, S.S.** & **Montgomery, R. (1992).** The structure of optimal controls for a steering problem. *IFAC Conference on Nonlinear Control Systems Design*. Bordeaux, France, 135-140.
34. **Sontag, E.D. (1995).** Control of Systems Without Drift via Generic Loops. *IEEE Transactions on Automatic Control*, 40(7), 1210-1219.
35. **Souères, P.** & **Laumond, J.P. (1996).** Shortest path synthesis for a car-like robot. *IEEE Transactions on Automatic Control*, 41(5), 672-688.
36. **Sussmann, H.J. (1990).** *Nonlinear Controllability and Optimal Control*, New York: M. Dekker.
37. **Sussmann, H.J.** & **Tang, W. (1991).** *Shortest paths for Reeds-Shepp car: a worked out example of the use of geometric techniques in nonlinear optimal control* (Report SYCON-91-10). New Brunswick Rutgers University.

Gustavo Arechavaleta Servín

Obtuvo la Maestría y el Doctorado en Robótica por el Instituto Nacional de Ciencias Aplicadas de Toulouse, Francia en 2004 y 2007 respectivamente. Su investigación durante el periodo 2003-2007 la realizó en el Grupo de Robótica Humanoide del LAAS-CNRS en Toulouse, Francia. Desde agosto de 2008 es Investigador CINVESTAV adscrito al Grupo de Robótica y Manufactura Avanzada en la Unidad Saltillo. Ha desarrollado algoritmos de planificación y control de movimientos para robots antropomorfos. Sus intereses actuales en investigación se encuentrán en robótica humanoide.