



UNIVERSIDAD SIMÓN BOLÍVAR
Decanato de Estudios Profesionales
Ingeniería en Electrónica

ARQUITECTURA DE OPTIMIZACIÓN CONVEXA PARA
LOCOMOCIÓN DE ROBOTS MULTI-PATAS EN TERRENOS
IRREGULARES

Trabajo de Grado presentado a la Universidad Simón Bolívar por
Bernardo Aceituno Cabezas

Como requisito parcial para optar al grado de
Ingeniero en Electrónica

Realizado con la tutoría del Profesor
Gerardo Fernández-López

Septiembre, 2017



UNIVERSIDAD SIMÓN BOLÍVAR
Decanato de Estudios Profesionales
Ingeniería en Electrónica

ARQUITECTURA DE OPTIMIZACIÓN CONVEXA PARA
LOCOMOCIÓN DE ROBOTS MULTI-PATAS EN TERRENOS
IRREGULARES

Este Trabajo de Grado ha sido aprobado en nombre de la Universidad Simón Bolívar por el siguiente jurado examinador:

Presidente
(Nombre y firma)

Miembro Externo
(Nombre y firma)
Indicar Institución a la que pertenece

Miembro Principal-Tutor
(Nombre y firma)

Fecha:_____

Resumen

Este trabajo presenta una arquitectura de optimización para planificación de locomoción en robots multi-patas. Métodos tradicionales de planificación de caminado planifican una secuencia de pisadas a seguir y generan una trayectoria dinámicamente estable con un modo de caminado fijo, usando modelos simplificados como el péndulo invertido y el criterio del "Zero Moment Point" (ZMP). Mientras que esta combinación ha funcionado exitosamente en terreno plano, estas estrategias no se generalizan a terrenos irregulares, o aquellos que requieren movimientos agresivos y transiciones de modo de caminado. Para sobreponerse a estas dificultades, en este trabajo presentaremos un esquema de optimización el cual puede planificar simultáneamente las posiciones de contacto y el modo de caminado a seguir, usando método de optimización convexa entera y heurísticas de terreno. Adicionalmente, haremos uso del criterio de estabilidad del Contact Wrench Cone (CWC), el cual generaliza el criterio del ZMP para terreno irregular con restricciones de cono de fricción, y planificaremos el movimiento de caminado de un modelo dinámico centroidal mediante un programa de optimización convexa. Nuestro enfoque se valida exitosamente en un cuadrúpedo LittleDog de *BostonDynamics*© sobre distintos terrenos irregulares. Demostramos que en terreno plano nuestro esquema genera caminado periódicos, iguales a aquellos esperados de un generador central de patrones (CPG) y un planificador de ZMP; Mientras que en terreno irregular nuestro planificador puede exitosamente generar movimientos que adaptan su modo de caminado al terreno, con trayectorias de centro de masa que respetan las restricciones de estabilidad impuestas por los conos de fricción, las cuales son violadas por un planificador de ZMP. Nuestro enfoque presenta un rendimiento significativamente superior a métodos de estado-del-arte, en términos de eficiencia computacional y robustez ante terrenos irregulares.

Palabras Clave: Locomoción Multi-Patas, Optimización Convexa, Optimización de Enteros Mixtos, Cono de Llave de Contactos

Índice general

Índice de cuadros	v
Índice de figuras	vi
1. INTRODUCCIÓN	1
1.1. Contexto del Problema	1
1.2. Planteamiento del Problema	2
1.3. Objetivos del Trabajo	5
1.3.1. Objetivo general	5
1.3.2. Objetivos específicos	6
1.4. Contribuciones	6
1.5. Estructura del Libro	9
2. ANTECEDENTES	10
2.1. Optimización Numérica	10
2.1.1. Optimización Convexa	10
2.1.2. Optimización Convexa de Enteros Mixtos	12
2.1.3. Optimización Robusta	13
2.2. Dinámica de Contacto	14
2.2.1. Dinámica Centroidal	15
2.2.2. Zero-Moment Point	16
2.2.3. Cono de Llaves de Contactos	17
2.2.4. Margen de Cono de Llave de Contactos	18
2.3. Planificación de caminado	19
2.3.1. Planificación de contactos	22
2.3.2. Planificación de movimientos	23
3. PLANIFICACIÓN DE PISADAS MEDIANTE OPTIMIZACIÓN DE ENTEROS MIXTOS	25
3.1. Introducción	25
3.1.1. Desarrollos relevantes	26
3.1.2. Estructura del capítulo	27
3.2. Formulación Propuesta	28
3.2.1. Restricciones cinemáticas	28
3.2.2. Evasión de obstáculos	31

3.2.3.	Orientación de la base	32
3.2.4.	Duración del plan	33
3.2.5.	Objetivos	34
3.2.6.	Formulación completa	35
3.3.	Conclusiones	36
4.	PLANIFICACIÓN DE CAMINADO ROBUSTO MEDIANTE OPTI- MIZACIÓN CONVEXA	37
4.1.	Introducción	38
4.1.1.	Desarrollos Relevantes	39
4.1.2.	Contribución y estructura del capítulo	40
4.2.	Enfoque técnico	41
4.2.1.	Optimización de modo de caminado	41
4.2.2.	Planificación de movimientos robustos	46
4.3.	Conclusión	48
5.	SIMULACIONES Y RESULTADOS	49
5.1.	Plataformas y ambiente de simulaciones	49
5.2.	Planificador de Pisadas	49
5.2.1.	Hexápodos	50
5.2.2.	Cuadrúpedos	51
5.2.3.	Rendimiento	52
5.3.	Planificador de Movimiento y Caminado	53
5.3.1.	Terreno Plano	53
5.3.2.	Terreno irregular	54
6.	CONCLUSIONES	58
6.1.	Conclusiones	58
6.2.	Trabajo Futuro	59
6.2.1.	Planificación de pisadas	59
6.2.2.	Planificación de movimientos	59
	Bibliografía	61

Índice de cuadros

5.1. Rendimiento del MIQP sobre distintos escenarios	53
--	----

Índice de figuras

1.1. HyQ: un robot cuadrúpedo de actuación hidráulica.	2
1.2. Robot ATLAS del equipo MIT compitiendo en el DARPA Robotics Challenge	3
1.3. Izquierda: Modo de caminado de arrastre, donde solo se levanta una pata a la vez. Derecha: Modo de caminado de trote, donde dos patas transicionan en paralelo.	4
1.4. Esquema de optimización propuesto en este trabajo. El robot captura una serie de regiones convexas del ambiente, luego planifica modo de caminado y pisadas de forma simultanea y finalmente genera un movimiento óptimo y dinamicamente estable.	7
2.1. Arriba: Ejemplo de conjunto convexo y no-convexo en el plano 2D. Abajo: Ejemplo de función convexa y no-convexa.	11
2.2. Interpretación geométrica del margen de CWC	19
2.3. Dos esquemas de planificación de caminado desacoplado tradicionales, presentados en [1, 2]. En ambos casos, se fija un modo de caminado, luego se busca una serie de contactos y de allí se genera una trayectoria de CoM y ZMP del robot.	21
3.1. Planificación de pisadas basada en búsqueda de grafos, como se muestra en [3]	27
3.2. Restricción geométrica en f_i , \mathcal{R} y el centro de contactos p_i	30
3.3. Aproximación lineal y convexa del alcance de la pata	31
3.4. Aproximación lineal segmentada de las funciones \sin y \cos , como se muestra en [4]	33
4.1. LittleDog caminando sobre terreno irregular con contactos no co-planares y conos de fricción linealizados (azul)	38
4.2. Representación gráfica del <i>wrench</i> agregado (vectores f y τ) dentro del cono de llave de contactos proyectado en el plano y su relación con el ZMP. . . .	40
4.3. Diagramas de fase de dos modos de caminado cuadrúpedo, donde el eje horizontal representa el porcentaje de ciclo de transferencia ejecutado en cada fase, y el eje vertical los soporte frontales 'F' y traseros 'H' (LF, RF, LH, RH).s	42
4.4. Izquierda: Ejemplo de terreno segmentado, con conjuntos de terreno admisibles, dados como entrada al planificador de pisadas. Derecha: Ejemplo de un modo de caminado optimizado.	44

4.5. Conjunto elipsoidal de incertidumbre B_r (gris) de la posición del centro de masa r (rojo)	47
5.1. A <i>BH3R</i> hexápodo siguiendo un plan de pisadas generalizado sobre un camino de piedras.	50
5.2. <i>BH3R</i> un plan de pisadas que requiere rotación de 90°	51
5.3. <i>BH3R</i> siguiendo un plan de pisadas sobre terreno irregular	51
5.4. <i>LittleDog</i> siguiendo el plan de pisadas para un camino de piedras complejo.	52
5.5. <i>LittleDog</i> siguiendo un plan de pisadas sobre un terreno irregular	52
5.6. Diagrama de fase para un modo de caminado y trayectoria de CoM en terreno plano	54
5.7. Izquierda: Comparación de margen de CWC entre planificación de ZMP y el enfoque propuesto. Derecha: $ k_G _1$ para caminado en terreno plano	55
5.8. Arriba: <i>LittleDog</i> siguiendo un plan de caminado robusto obtenido con nuestro enfoque de planificación en terreno irregular con $\mu_s = 0,5$. Abajo: Modo de caminado generado para un terreno irregular, verde indica troto y azul indica caminado.	56
5.9. Trayectoria de CoM para terreno irregular con $\mu_s = 0,5$ usando ZMP (rojo) y margen de CWC (azul).	56
5.10. Izquierda: Comparación de margen de CWC usando planificación de ZMP y el enfoque propuesto. Derecha: $ k_G _1$ para terreno irregular con $\mu_s = 0,5$	57

INTRODUCCIÓN

Este capítulo introductorio desarrolla el planteamiento de los problemas de planificación de caminado tratados en esta tesis. En primer lugar se especifica el contexto de este trabajo, para luego pasar a la definición, antecedentes e importancia del problema. Finalmente se exponen los objetivos del trabajo.

1.1. Contexto del Problema

Una de las principales metas de la investigación en robots móviles es el diseño de robots capaces de acceder a ambientes complejos e inaccesibles para humanos, como es el caso de estructuras afectadas por desastres naturales [5]. Sin embargo, los robots móviles tradicionales, accionados por ruedas o hélices, se ven severamente limitados por las restricciones que imponen estos ambientes irregulares y no estructurados, muy distintos a los pasillos o autopistas para los que son diseñados. En este contexto, los robots caminantes resultan ser herramientas extremadamente poderosas a la hora de acceder a dichos ambientes y maniobrar de forma robusta sobre estos. Particularmente, el diseño de plataformas caminantes modernas [6, 7, 8] favorece de forma significativa la habilidad para adaptarse a terrenos irregulares y maniobrar sobre estos. Un ejemplo de ello es HyQ [9], un robot cuadrúpedo de actuación hidráulica diseñado para movimiento dinámico sobre terrenos irregulares, mostrado en la figura 1.1, desarrollado por el Instituto Italiano de Tecnología (IIT). Por lo tanto, la comunidad de robótica de los últimos años ha dado significativa atención a la tarea de generar trayectorias y movimientos de caminado para este tipo de robots frente a terrenos irregulares.

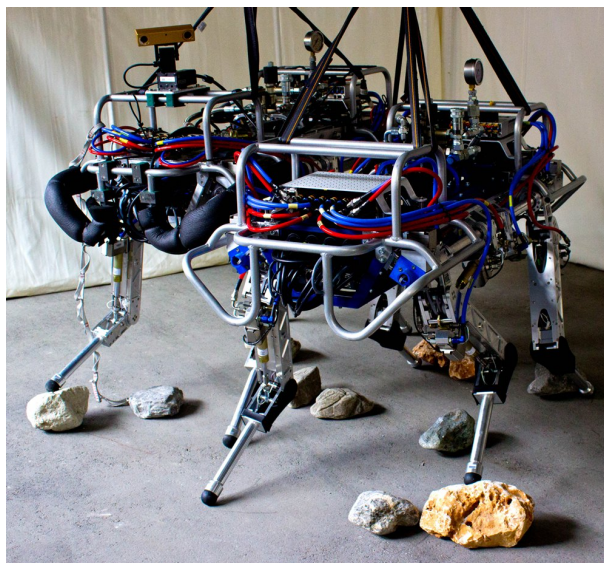


Figura 1.1: HyQ: un robot cuadrúpedo de actuación hidráulica.

Recientemente, en el año 2015, el *DARPA Robotics Challenge* (DRC) retó a una serie de robots caminantes de última generación compitiendo ante una serie de tareas sencillas, en un ambiente altamente restringido y de comunicación deteriorada. Entre las tareas a cumplir se encontraban: tripular un automóvil, abrir una puerta, taladrar una pared, caminar sobre un terreno irregular y subir por una escalera. Aunque de apariencia simple, la mayoría de los equipos competidores fueron incapaces de cumplir todas las tareas planteadas, a pesar de contar con potentes plataformas robóticas humanoides como la mostrada en la figura 1.2. Las principales causas de tal desempeño son las restricciones impuestas sobre la compleja dinámica de estos robots, lo que no permite representar de forma fiel las interacciones del robot con el ambiente. Por lo tanto, la mayoría de estos equipos vieron como sus robots fallaban en mantener un caminado o una postura estable debido a la presencia de una fuerza externa o un contacto no modelado.

1.2. Planteamiento del Problema

Desafortunadamente, las dinámicas de contacto a la que se ven sometidas estas plataformas resultan complejas de representar usando las matemáticas tradicionales para el modelado y control de sistemas dinámicos, al resultar en un sistema no-lineal con dinámica



Figura 1.2: Robot ATLAS del equipo MIT compitiendo en el DARPA Robotics Challenge

híbrida. Adicionalmente, se añade la dificultad de la sub-actuación de estas plataformas, debido a la falta de autoridad de control sobre los 6 grados de libertad correspondientes a la posición y orientación de la base del robot. Por lo tanto, la mayoría de investigadores en locomoción de robots caminantes se ven limitados a recurrir a modelos simples y a sobre restringir el problema de caminado para simplificar la dinámica del sistema [10].

Esta serie de simplificaciones y asunciones en las dinámicas de caminado son la principal causa de los movimientos conservadores a los que se ven limitados los robots caminantes, resultando en movimientos rígidos y altamente restringidos a terrenos simples y a interacciones limitadas con el ambiente.

Como es notado por Mastalli [11], es por estas limitaciones que por lo general se resuelve en desacoplar el problema de planificación en dos etapas: una que planifique una secuencia de pisadas a seguir y otra que genere el movimiento continuo de cada elemento terminal y el centro de masa, de forma que se pueda seguir tal secuencia de pisadas. Por lo tanto, la tarea de mover un robot con patas de una configuración inicial a una configuración final es definida por Deits [12], para un robot definido por su estado dinámico x , el cual incluye posiciones y velocidades del cuerpo del robot, como dos problemas distintos:

Definición 1.1 (Problema de Planificación de Pisadas): *Dado un estado inicial*

x_0 y un conjunto de estados objetivo X_f , encontrar una secuencia de posiciones de los elementos terminales f_1, f_2, \dots, f_{N_f} tales que siguiendo tal secuencia de pisadas se va a llevar al robot de forma segura de x_0 a $x_f \in X_f$.

Definición 1.2 (Problema de Planificación de Caminado): Dado x_0 , X_f y una secuencia de pisadas f_1, f_2, \dots, f_{N_f} , encontrar un trayectoria de estados que sea realizable y controlable a través de la secuencia de pisadas desde x_0 hasta $x_f \in X_f$.

Sin embargo, este planteamiento del problema ignora la decisión discreta del orden en que se seguirán las pisadas planificadas, lo cual puede ser crítico para cambios de terreno o para ciertas geometrías de robot. Por lo tanto, en el caso particular del caminado en robots multi-patas, emerge el problema de planificación del modo de caminado. El modo de caminado puede entenderse de forma general como el orden temporal en que se realizan las transiciones de una pisada a la siguiente. Por lo general, los modos de caminado son representados mediante diagramas de fase, los cuales especifican que elemento terminal se encuentra en apoyo en cada instante de tiempo [13], como se muestra en la figura 1.3.

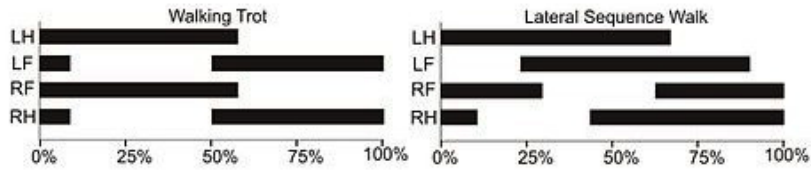


Figura 1.3: Izquierda: Modo de caminado de arrastre, donde solo se levanta una pata a la vez. Derecha: Modo de caminado de trote, donde dos patas transicionan en paralelo.

Por lo tanto, el problema de planificación de caminado puede ser entendido como la búsqueda del movimiento mediante el que se transfieren las patas de una configuración del robot a otra. Formalmente, definiremos este problema como:

Definición 1.3 (Problema de Planificación de Modo de Caminado): Dado un estado inicial x_0 y un conjunto de estados objetivo X_f , encontrar una secuencia de posiciones de los elementos terminales f_1, f_2, \dots, f_{N_f} y el orden temporal t_1, t_2, \dots, t_{N_f} en que estos

serán seguidos tal que siguiendo tal secuencia de pisadas en el orden obtenido se lleve al robot de forma segura y adaptativa de x_0 a $x_f \in X_f$.

En este caso, se mezclan tanto variables continuas, correspondientes a la dinámica del robot, junto a variables discretas, correspondientes al orden de transferencia de las patas. Por lo que la mayoría de los investigadores resuelven en fijar un modo de caminado para el robot, eliminando la componente discreta del problema. Otro método de aplicación común es planificar el modo de caminado de forma desacoplada, separando el orden de transferencia de la dinámica del robot [7, 8, 14]. Ambos métodos resultan en la desventaja de que el robot no puede razonar global-mente sobre la mejor dinámica de caminado, considerando hacia *donde y cuando* mover cada pata.

Finalmente, uno de los problemas fundamentales que emerge en proceso de planificación es la estabilidad en el caminado. Esto puede verse como la garantía de que la trayectoria de caminado podrá ejecutarse sin que el robot se deslice o caiga. En términos generales, se puede decir que un movimiento de caminado es robusto si todas las fuerzas de contacto con el ambiente son estáticas, por lo que el robot no puede deslizarse ni perder contacto durante una transición [15].

1.3. Objetivos del Trabajo

En esta sección presentaremos de forma detallada los objetivos generales y específicos de este trabajo de tesis.

1.3.1. Objetivo general

El objetivo de este trabajo es diseñar y desarrollar una arquitectura de planificación de caminado para robots mutli-patas, en la que se puedan considerar factores ignorados por arquitecturas tradicionales. Esto incluye el modo de caminado, el manejo de ambientes irregulares y la estabilidad generalizada del caminado. Para ello, vamos a manejar herramientas de optimización convexa y optimización de trayectorias, en particular herramientas de Programación Convexa de Enteros Mixtos (MICP por sus siglas en ingles)

y Optimización Robusta. Estas herramientas serán aprovechadas para desarrollar algoritmos pueda razonar de forma paralela sobre factores como el irregularidad del terreno, la fricción en los contactos y la factibilidad de la ejecución, a la par que presentar garantías de optimalidad y robustez.

1.3.2. Objetivos específicos

1. Realizar una revisión del estado del arte en el área de la planificación de movimiento para plataformas caminantes.
2. Realizar una revisión del estado del arte en optimización de trayectorias, para ambientes no convexos y dinámicas no-lineales.
3. Diseñar una arquitectura de software que permita verificar el funcionamiento del método mediante simulaciones dinámicas y visualizaciones.
4. Diseñar un algoritmo de planificación simultanea de pisadas y modo de caminado para ambientes continuos con garantías de optimalidad global, que considere factores de irregularidad en el terreno, rotación en el robot y evasión de obstáculos.
5. Diseñar un algoritmo de planificación de trayectoria de caminado optima, el cual pueda proveer garantías de robustez y estabilidad al momento de ejecutar el movimiento, sin limitaciones por el tipo de terreno ni restricciones en la dinámica del robot.
6. Implementar ambos algoritmos de planificación como parte de una arquitectura de planificación generalizada, adaptable a cualquier robot sin limitaciones por geometría o numero de patas.
7. Realizar una serie de experimentos sobre una plataforma caminante moderna en un ambiente simulado y evaluar los resultados obtenidos.

1.4. Contribuciones

En este trabajo, abordaremos el problema de planificación de caminado óptimo en robots multi-patas, empleando herramientas de optimización de trayectorias, optimi-

zación convexa, programación de enteros mixtos y optimización robusta. Abordaremos este problema proponiendo un nuevo método de planificación desacoplado que permitirá, a diferencia de métodos tradicionales, generar secuencias de pisadas óptimas, modos de caminado adaptados a la topología del terreno y trayectorias de centro de masa (CoM) que ofrezcan garantías formales de robustez en su ejecución. En particular, la primera etapa del planificador se encargará de optimizar una secuencia de pisadas en paralelo al modo de caminado, que aseguren que se siga el camino mas corto, se respeten las restricciones cinemáticas del robot y sea planteado como un problema convexo. Por otro lado, la segunda etapa del planificador se encargará de optimizar una trayectoria de centro de masa del robot, tal que maximice la robustez de la ejecución y la naturalidad del movimiento, basada en un criterio generalizado de estabilidad del caminado [16, 17] que permite definir garantías formales de robustez aun en terreno irregular con contactos no-coplanares, extendiendo la implementación para robots humanoides desarrollada por Dai [18], usando el margen Cono de Llave de Contactos (Contact Wrench Cone o CWC) como métrica de estabilidad generalizada.

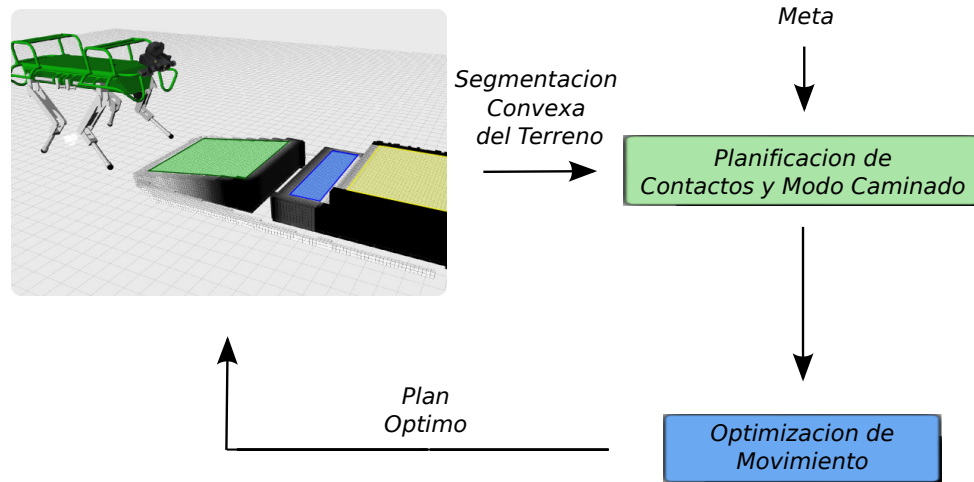


Figura 1.4: Esquema de optimización propuesto en este trabajo. El robot captura una serie de regiones convexas del ambiente, luego planifica modo de caminado y pisadas de forma simultanea y finalmente genera un movimiento óptimo y dinamicamente estable.

La contribución clave de este trabajo, es la habilidad de razonar de forma simultanea sobre *donde y cuando* ejecutar la secuencia de caminado, lo cual es comúnmente ignorado en arquitecturas tradicionales de planificación de caminado multi-patas. Esta considera-

ción permite que el caminado pueda ser menos conservador y que, por lo tanto, se pueda explotar la dinámica híbrida inherente del sistema. Tal planificación permite obtener planes de caminado mucho más complejos que arquitecturas tradicionales [19, 8, 7, 20, 3, 21, 22] con solo información básica de la segmentación del terreno en regiones convexas, como se muestra en la figura 1.4. Esta formulación del problema de planificación de pisadas, también permite considerar factores no lineales usualmente ignorados en métodos tradicionales, como la evasión de obstáculos en terrenos irregulares y la orientación del cuerpo del robot, usualmente fijada. A diferencia de estos métodos tradicionales, nuestra implementación es capaz de razonar cuando es conveniente evadir un obstáculo o caminar sobre este.

Otra de las ventajas más importantes de este método es la garantía de optimalidad global. Esto es posible ya que el problema de planificación será planteado como dos problemas de *Optimización Convexa* [23], al emplear herramientas de optimización de enteros mixtos [24] y optimización robusta [25] para representar las restricciones bi-lineales y discretas inherentes a la evasión de obstáculos, orientación del cuerpo, modo de caminado y la dinámica rotacional del sistema. Una descripción técnica de estas herramientas será presentada en el siguiente capítulo, a fin de familiarizar al lector con las herramientas de optimización convexa y sus ventajas.

Por otro lado, la elección de un criterio de estabilidad generalizado permite asegurar estabilidad en transiciones de modo de caminado, lo cual va más allá de la capacidad de métodos tradicionales basados en criterios de estabilidad basados en Zero-Moment Point (ZMP) [26], los cuales asumen que el caminado ocurre sobre terreno plano y que la altura del centro de masa se mantiene constante. Por lo tanto, este método permite generar movimientos más naturales que exploten la dinámica del sistema, respetando los límites impuestos por las interacciones de contacto con el ambiente.

Este método será demostrado en dos plataformas caminantes de geometrías y número de patas distintos, de forma que se pueda ilustrar la generalidad de su imple-

mentación. Los resultados obtenidos representan una mejora significativa ante métodos tradicionales basados en algoritmos de búsqueda discreta y programación dinámica (DP), en términos de garantías de optimalidad y rendimiento. Estos resultados abren una oportunidad para el desarrollo de futuras herramientas de planificación en tiempo real, los cuales puedan asegurar un resultado óptimo para un cierto horizonte de planificación.

1.5. Estructura del Libro

El contenido restante de este libro se encuentra organizado de la siguiente manera. El Capítulo 2 presentará los conceptos básicos de optimización convexa y las herramientas a utilizar en este trabajo, adicionalmente, familiarizará al lector con las arquitecturas tradicionales de planificación de caminado y los desarrollos más relevantes en esta área. El Capítulo 3 presentará nuestro método de planificación de pisadas y modos de caminado, basado en programación convexa de enteros mixtos. El Capítulo 4 presentará nuestro método de planificación de centro de masa basado en margen de cono de llave de contactos. El Capítulo 5 presentará una serie de experimentos en un ambiente simulado sobre dos plataformas caminantes multi-patas. Finalmente, el Capítulo 6 concluirá este libro con una discusión de los resultados obtenidos y los desafíos a abordar en trabajos futuros.

ANTECEDENTES

En este capítulo presentaremos los conceptos básicos usados como base para el desarrollo de este trabajo. Adicionalmente, realizaremos una revisión del estado del arte en planificación de movimiento para robots caminantes.

2.1. Optimización Numérica

Un problema de optimización numérica [23] puede definirse, de forma general, como:

$$\underset{x}{\text{minimizar}} \ J(x)$$

Tal que:

$$g(x) \leq 0$$

Donde x es una variable, ó un *conjunto de variables*, de decisión independiente, sujeta a las restricciones impuestas por $g(x)$, y $J(x)$ es una función de costo que se busca minimizar.

2.1.1. Optimización Convexa

En teoría de optimización, los problemas de optimización convexa representan un conjunto con propiedades deseables en términos de eficiencia computacional y convergencia. En teoría de conjuntos, un conjunto convexo se define como:

Definición 2.1 (Conjunto Convexo:) *Un conjunto convexo C es aquel tal que para cada par de elementos A y B cualquiera, pertenecientes al conjunto, y escalar $t \in [0, 1]$, la relación afín $A(1 - t) + Bt$ se encuentra contenida en el conjunto.*

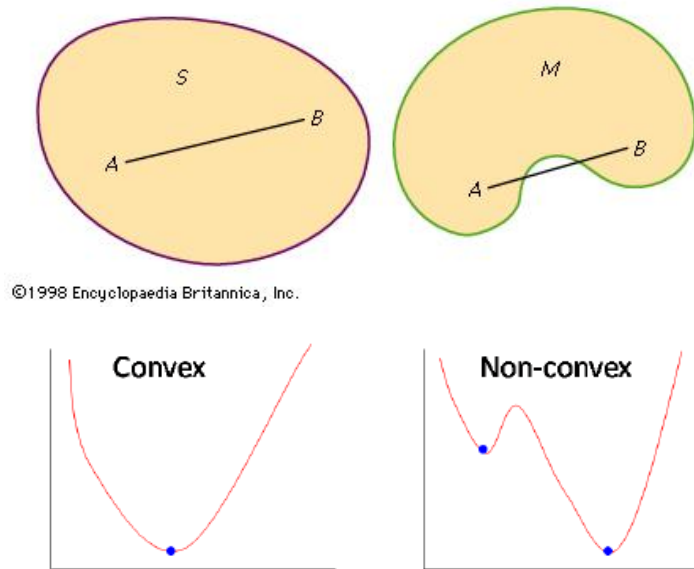


Figura 2.1: Arriba: Ejemplo de conjunto convexo y no-convexo en el plano 2D. Abajo: Ejemplo de función convexa y no-convexa.

Por otro lado, en teoría de funciones, una función convexa se define como:

Definición 2.2 (Función Convexa:) *Una función convexa es aquella cuyo epígrafo representa un conjunto convexo en el hiper-plano.*

Una representación simple de ambas definiciones y sus contrapartes en el plano bidimensional es presentada en la figura 2.1.

En términos de optimización, una función convexa cuenta con la propiedad de solo poseer un punto mínimo, siempre que la variable independiente se encuentra restringida por funciones convexas. Por lo tanto, a la hora encontrar el punto mínimo, basta con descender en dirección opuesta al gradiente hasta hallar el punto mínimo de la función. Por ello definiremos un problema de optimización convexa como:

Definición 2.3 (Optimización Convexa:) *Es aquel problema de optimización donde la función de costo representa una función convexa y las restricciones sobre las variables de decisión forman un conjunto convexo.*

Este tipo de problemas son fácilmente manejables por software de optimización y convergen de forma rápida a soluciones óptimas. Algunos casos especiales de optimización convexa incluyen:

1. Programación Lineal: es un problema de optimización donde la función de costos es lineal y las restricciones representan relaciones lineales entre las variables de decisión. Ejemplo: minimizar norma L_1 con restricciones politópicas.
2. Programación Cuadrática: es un problema de optimización donde la función de costos es cuadrática y las restricciones representan relaciones lineales entre las variables de decisión. Ejemplo: minimizar norma cuadrática con restricciones politópicas.
3. Programación de Conos de Segundo Orden: es un problema de optimización donde la función de costos es lineal y las restricciones representan relaciones cónicas de segundo-orden entre las variables de decisión. Ejemplo: mínima norma L_1 con restricciones elipsoidales.
4. Programación Semidefinida: es un problema de optimización donde las restricciones representan matrices semidefinidas definidas por las variables de decisión y la función de costos es la traza de una matriz semidefinida. Ejemplo: Ecuación de Riccati, Programación de suma de cuadrados (SOS).

Los avances recientes en software de optimización permiten manejar de forma muy eficiente este tipo de problemas, aún cuando el condicionamiento numérico no es favorable. En el caso particular de la programación lineal y la programación cuadrática, existen herramientas de gran madurez que permiten resolver estos problemas de forma casi instantánea, por debajo de los mili-segundos. Gracias a esto, las herramientas de planificación y búsqueda en espacios continuos de gran escala son fácilmente implementables cuando estas se pueden plantear como problemas convexos.

2.1.2. Optimización Convexa de Enteros Mixtos

Un caso particular de optimización numérica es el programa de enteros mixtos (MIP), este es tipo de programa es aquel donde un conjunto de las variables de decisión se

encuentran restringidas a ser números enteros [24]. El uso de este tipo de variables expande forma significativa el alcance de los problemas de optimización numérica. De forma general un problema de enteros mixtos puede escribirse como:

$$\underset{x,z}{\text{minimizar}} \ J(x, z)$$

Tal que:

$$g(x, z) \leq 0$$

$$z \in \mathbb{Z}$$

Un caso particular es donde la variable z se encuentra restringida a 0 o 1. Tales problemas permiten usar las variables binarias para modelar condicionales lógicos dentro de las restricciones del programa, usando formulación de M-grande [24]. Un ejemplo de este tipo de restricción es:

$$z_1 \Rightarrow x \leq a \equiv x + Mz_1 \leq a + M, \ M \gg a, \ z_1 \in \{0, 1\}$$

Sin embargo, las variables enteras hacen al problema de optimización no-convexo, y por lo tanto más difícil de resolver. Sin embargo, el caso particular en que el problema laxo es convexo, donde z es una variable continua en el conjunto $[0, 1]$, es manejable de forma eficiente usando métodos de optimización convexa y luego hacer uso de algoritmos de búsqueda eficientes para obtener el arreglo de variables binarias que minimicen la función de costo $J(x, z)$, conociendo el mínimo global dado por el problema laxo [27].

2.1.3. Optimización Robusta

Un problema de optimización robusta p^* es aquel donde los parámetros de las restricciones tienen incertidumbre [25]. De forma general, este tipo de problemas pueden escribirse como:

$$p^* = \min_x J(x) \quad \text{s.t.} \quad f(x, u) \leq 0 \quad \forall u \in U$$

donde x es la variables de decisión, f es la función de de restricciones, u es el parámetro con incertidumbre, y U es el conjunto admisible de tal parámetro. En el caso específico

en que el conjunto U es una elipsoide centrada en el punto u^* , este puede representarse como:

$$U = \{u \mid (u - u^*)^T Q_u (u - u^*) \leq 1\}$$

Por lo tanto, si la relación entre los parámetros y las variables de decisión es bi-lineal, el problema se convierte en:

$$\begin{aligned} p^* &= \min_x J(x) \\ s.t : (Ax)^T u &\leq b, (u - u^*)^T Q_u (u - u^*) \leq 1 \end{aligned}$$

En este caso, la relación bi-lineal hace a p^* un problema no-convexo. Sin embargo, para una variable acotada por un conjunto elipsoidal, se conoce la siguiente relación:

$$\max_{\substack{u \\ u \in U}} c^T u = c^T u^* + \sqrt{c^T Q_u^{-1} c} \quad (2.1)$$

Teniendo esto en mente, el problema p^* puede ser relajado al asumir que se busca minimiza sobre la frontera del conjunto de incertidumbre. Por lo tanto, solo optimizando sobre la cota superior de la restricción no convexa. Entonces, el problema se relaja al problema laxo p_1^* :

$$\begin{aligned} p_1^* &= \min_x J(x) \\ s.t : (Ax)^T u^* &+ \sqrt{x^T A^T Q_u^{-1} A x} \leq b \end{aligned}$$

El cual es un programa de conos de segundo orden [23], el cual es un problema especial de optimización convexa. Tal problema garantiza que las restricciones del problema se cumplan para cualquier valor de u y puede ser resuelto de forma eficiente a su óptimo global.

2.2. Dinámica de Contacto

En esta sección haremos un repaso breve al análisis dinámico de un robot caminante y presentaremos los principales conceptos de interés para este trabajo.

2.2.1. Dinámica Centroidal

Para un robot de base flotante, aquel cuyos elementos terminales no se encuentran unidos al suelo, la dinámica del sistema puede representarse de forma general como:

$$H(q)\ddot{q} + G(q, \dot{q}) = S^T \tau + J_e^T \lambda$$

Donde H es la matriz de inercia, G es la matriz que captura los efectos gravitacionales, centrífugos y de coriolis, S es una matriz de selección de las articulaciones actuadas, τ son los pares de cada articulación, J_e es el jacobiano a los puntos de contacto y λ es el conjunto de fuerzas de contacto. En este caso, para $n+6$ dimensiones, con n articulaciones, las ecuaciones del sistema pueden descomponerse en:

$$H_u(q)\ddot{q} + G_u(q, \dot{q}) = J_{e^u}^T \lambda \quad (2.2)$$

$$H_a(q)\ddot{q} + G_a(q, \dot{q}) = \tau + J_{e^a}^T \lambda \quad (2.3)$$

donde la ecuación 2.3 representa la dinámica de los grados de libertad actuados del sistema y la ecuación 2.2 representa la dinámica de los grados de libertad sub-actuados, correspondientes a la base. Por ello, la ecuación 2.2 puede interpretarse como la ecuación Newton-Euler del sistema, la cual representa los cambios en el momento del robot ante la presencia de fuerzas externa. Por otro lado, asumiendo suficiente autoridad de torque, la ecuación 2.3 puede entenderse como las condiciones necesarias para asegurar la factibilidad cinemática del sistema, mientras que la ecuación 2.2 representa las condiciones para que haya consistencia dinámica en el sistema [28]. Por lo tanto, si asumimos que las articulaciones no presentan fricción interna y que todas las fuerzas internas son canceladas por tercera ley de Newton, las ecuaciones dinámicas del sistema pueden ser reducidas a:

$$\dot{h}_G = \begin{pmatrix} m\ddot{r} \\ \dot{k}_G \end{pmatrix} = \begin{pmatrix} \sum_{j=1}^{n_{ee}} \lambda_j + m\mathbf{g} \\ \sum_{j=1}^{n_{ee}} (p_j - r) \times \lambda_j \end{pmatrix} \quad (2.4)$$

donde h_G es el momento del sistema, m es la masa del robot, \mathbf{g} el vector de gravedad, r es la posición del centro de masa, n_{ee} es el numero de elementos terminales, p es la posición de los elementos terminales y k_G es el momento angular centroidal. Este modelo presentado en la ecuación 2.4 es referido como el modelo de dinámica centroidal, pues captura los efectos de la dinámica del sistema sobre el centro de masa del robot [29].

2.2.2. Zero-Moment Point

En esta sección presentaremos una derivación sencilla del criterio de estabilidad del Zero-Moment Point (ZMP), como se presenta en [30]. Si asumimos que el movimiento de caminado se realiza sobre terreno plano, entonces tomaremos la componente en z del vector de fuerza $F_{i,z}$ como la fuerza de contacto normal en la i -ésima posición. Adicionalmente, si asumimos que el robot solo empuja el suelo sin halar con sus contactos, entonces se cumple la siguiente relación:

$$\forall i, F_{i,z} \geq 0 \Rightarrow \sum_i F_{i,z} \geq 0 \Rightarrow \ddot{z} \geq -g$$

Es decir, al no ser capaz de halar con las fuerzas de contacto, el centro de masa no puede caer con aceleración mayor a la gravedad. Luego, usando el modelo de fricción de Coulomb sobre terreno plano, con coeficiente de fricción μ , entonces:

$$\forall i, |F_{i,x}| \leq \mu F_{i,z} \Rightarrow \sum_i |F_{i,x}| \leq \mu \sum_i F_{i,z} \Rightarrow |\ddot{x}| \leq \mu(\ddot{r}_z + g).$$

Donde el vector r es la posición del centro de masa y g el vector de gravedad. Luego, definiremos el Centro de Presión (CoP) como el punto en el terreno donde:

$$z = \frac{\sum_i p_{i,x} F_{i,z}}{\sum_i F_{i,z}}$$

donde p_i es la posición del i -ésimo contacto. Entonces, ya que los $p_{i,z}$ son iguales en el terreno plano, z_z es la altura del terreno. Luego al ser el *terreno plano y los contactos coplanares*, el centro de presión es un "zero-moment point" (ZMP), donde el momento agregado es nulo. Por lo tanto, las ecuaciones de momento y balance permiten derivar la siguiente relación dinámica para el centro de presión:

$$(m\ddot{r}_z + mg)(z - r) = (z_z - r_z)m\ddot{r} - \dot{k}_G \quad (2.5)$$

En este caso, si ignoramos las dinámicas de colisión y asumimos que el momento angular es constante, lo que implica que $\ddot{r}_z = \ddot{k}_G = 0$, entonces la relación $z_z - r_z$ es una altura constante h . Por lo tanto, la ecuación 2.5 se convierte en la tradicional "ecuación del ZMP", dada por:

$$z = r - \frac{h}{g}\ddot{r}$$

Esta relación *afín* implica que todo cambio en la aceleración del sistema viene dado por un cambio en la posición y en el zero-moment point. Mas aún, esta relación implica que el caminado siempre será estable si el zero-moment point se encuentra contenido dentro del polígono definido por los soportes de contacto, siempre que $\ddot{r}_z = \dot{k}_G = 0$ y que el terreno sea plano. Por lo tanto, se define el margen de estabilidad del ZMP como la distancia mínima entre z y el borde del polígono de soporte. Es importante notar que este criterio de estabilidad solo tiene validez sobre las condiciones impuestas: contactos en terreno plano, aceleración vertical nula, y momento angular constante, por lo que no es generalizable a terreno irregular y, aún cuando se cumpla, puede hacer que el robot caiga.

2.2.3. Cono de Llaves de Contactos

En el modelo de fricción de Coulomb, las fuerzas de contacto deben mantenerse dentro de un cono para asegurar que no exista deslizamiento, como se muestra en la figura 4.1. Esta noción puede generalizarse al mantener la llave de contacto (concatenación de fuerza y torque de contacto) dentro de cono de llaves [15]. Una simplificación común es aproximar el cono como un poliedro de n_e esquinas en cada contacto [18, 31], donde cada esquina e_i^j en la posición de contacto p_i es mapeada al espacio hexadimensional de llaves como:

$$w_i^j = \begin{bmatrix} e_i^j \\ p_i \times e_i^j \end{bmatrix} \quad (2.6)$$

luego, el cono de llave de contactos (CWC) se define como el conjunto admisible de la llave de contacto total. Tal conjunto puede calcularse al realizar la suma de *Minkowski* de todos los poliedros de llave en cada soporte, *siempre resultando en un conjunto convexo*. Como resultado de esto, este conjunto puede calcularse como el cono convexo generado por todos los rayos extremos en los n_c contactos:

$$CWC = \text{ConvexCone}(w_i^j \mid i = 1, \dots, n_c \ j = 1, \dots, n_e) \quad (2.7)$$

Donde $\text{ConvexCone}(v_1, \dots, v_n) = \{\sum_i^n \lambda_i v_i \mid \lambda_i \geq 0\}$ es una función que retorna el poliedro convexo con rayos extremos v_1, \dots, v_n . Este computo puede ser realizado facilmente cuando las posiciones de los contacto son dadas, los cuales pueden ser eficientemente obtenidos de planificadores de pisadas [4, 32, 7]. Este poliedro también puede ser descrito mediante sus

caras, como:

$$CWC = \{w | a_k^T w \leq 0, k = 1, \dots, n_f\} \quad (2.8)$$

donde n_f es el numero de caras del poliedro y $a_k \in \mathbb{R}^6$ es el vector normal a cada cara. Finalmente, ya que el criterio de estabilidad del CWC requiere que la llave de contacto agregada deben encontrarse en el CWC. Usando la segunda ley de Newton sobre las dinámicas centroidales del sistema, ignorando fricción entre articulaciones del robot, se puede requerir:

$$\dot{h}_O - w_g = \begin{bmatrix} m\ddot{r} \\ \dot{k}_O \end{bmatrix} - \begin{bmatrix} m\mathbf{g} \\ r \times m\mathbf{g} \end{bmatrix} \in CWC \quad (2.9)$$

Donde h_O es el momento del robot en el marco global, w_g es la llave gravitacional, r es el centro de masa (CoM) del robot, k_O el momento angular con respecto al marco global, m la masa y \mathbf{g} el vector de gravedad.

2.2.4. Margen de Cono de Llave de Contactos

La naturaleza dinámica del CWC lo convierte en un espacio de gran utilidad a la hora de determinar la robustez de un movimiento de caminado, en términos de la estabilidad dinámica del robot [15]. Por lo tanto, la comunidad de locomoción de robots introdujo la idea de un *Margen de Cono de Llave de Contactos*, inspirado en métricas de robustez en manipulación [16], y el cual se define en [18] como:

Definición 4.1 *Margen de Cono de Llave de Contactos: es la menor magnitud de perturbaciones de llave aplicadas a un cierto punto, tal que el robot no puede mantener el cono de contacto en su conjunto admisible, dadas las posiciones de los contactos y las restricciones de conos de fricción.*

Esto es equivalente a la máxima magnitud de llave ϵ , tal que la llave de contacto superpuesta a la perturbación se mantienen en el CWC. Una representación de esta noción proyectada en el plano bi-dimensional es mostrada en la figura 2.2. Algebraicamente:

$$\epsilon = \max \bar{\epsilon}$$

Tal que:

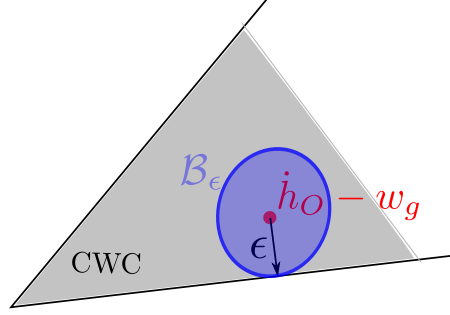


Figura 2.2: Interpretación geométrica del margen de CWC

$$B_{\bar{\epsilon}} = \left\{ \dot{h}_O - w_g + T(p_d)w_d \mid w_d^T Q_d w_d \leq \bar{\epsilon}^2 \right\} \subset CWC$$

donde w_d es una llave de perturbación aplicada en $p_d \in \mathbb{R}^3$, $T(P_d) \in \mathbb{R}^{6 \times 6}$ es la matriz de transformación que mapea la perturbación como una llave en el origen del marco de referencia global y $Q_d \in \mathbb{R}^{6 \times 6}$ es una matriz simétrica que mapea la norma en el espacio de llaves. Usando la descripción de facetas del CWC, esta representación del margen se puede replantear como:

$$\epsilon = \min_{i=1, \dots, n_f} -\bar{a}_i \begin{bmatrix} m\ddot{r} - m\mathbf{g} \\ \dot{k}_O - r \times m\mathbf{g} \end{bmatrix} \quad (2.10)$$

donde $\bar{a}_i = (a_i^T T(p_d) Q_d^{-1} T(p_d)^T a_i)^{-1/2} a_i^T$. En el capítulo 4 de este libro, vamos a formular un problema de optimización para generar trayectorias de centro de masa que maximicen este margen.

2.3. Planificación de caminado

Los enfoques tradicionales de planificación de caminado multi-patas suelen dividirse en dos vertientes principales:

1. **Planificación desacoplada:** estos enfoques [1, 2, 7, 33] separan el problema de planificación en dos o más etapas de planificación. Por lo general, estos esquemas trabajan bajo una arquitectura de: a) seleccionar modo de caminado, b) planificar secuencia de pisadas, c) planificar trayectoria de centro de masa. Estos métodos suelen ser los más comunes, pues son de rápida implementación y pueden ser implementados con algoritmos de alta eficiencia. Sin embargo, cuentan con la desventaja

de requerir modelos restringidos de la cinemática y dinámica del robot, ya que las etapas anteriores no son capaces de razonar sobre el modelo completo del sistema. Ejemplos de este tipo de arquitecturas son presentados en la figura 2.3.

2. **Planificación acoplada:** este tipo de enfoques [34, 11, 35, 36] tratan el problema de planificación en una sola etapa, donde se planifican contactos y movimientos de forma simultanea como parte de un solo algoritmo, por lo general con un modo de caminado pre-especificado. La principal ventaja de estos algoritmos es que permiten tomar ventaja de la estructura dinámica de estos sistemas, por lo que se pueden generar movimientos ágiles, agresivos y no-periódicos. La principal desventaja de este tipo de enfoques es que requieren algoritmos de alto costo computacional, debido a las relaciones no-lineales y no-convexas que implican. En la actualidad, aún no es posible resolverlos de forma interactiva y requieren minutos u horas de computo para generar movimientos complejos. Trabajos recientes [35] han logrado mejorar el desempeño de este tipo de planificadores, asumiendo un modo de caminado, recurriendo a modelos de dinámica reducida y restringiendo el movimiento a secuencias periódicas.

En este trabajo presentaremos un esquema de planificación desacoplado que tomará ventaja de las propiedades dinámicas del robot. Por lo tanto, el resto de esta sección presentará los métodos mas comunes para abordar cada etapa de planificación.

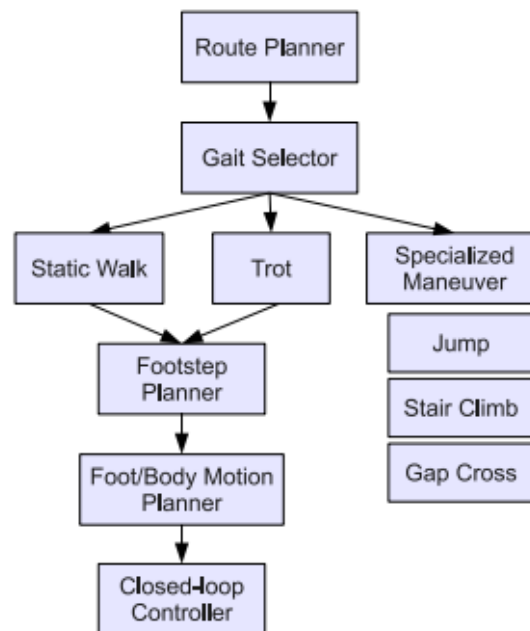
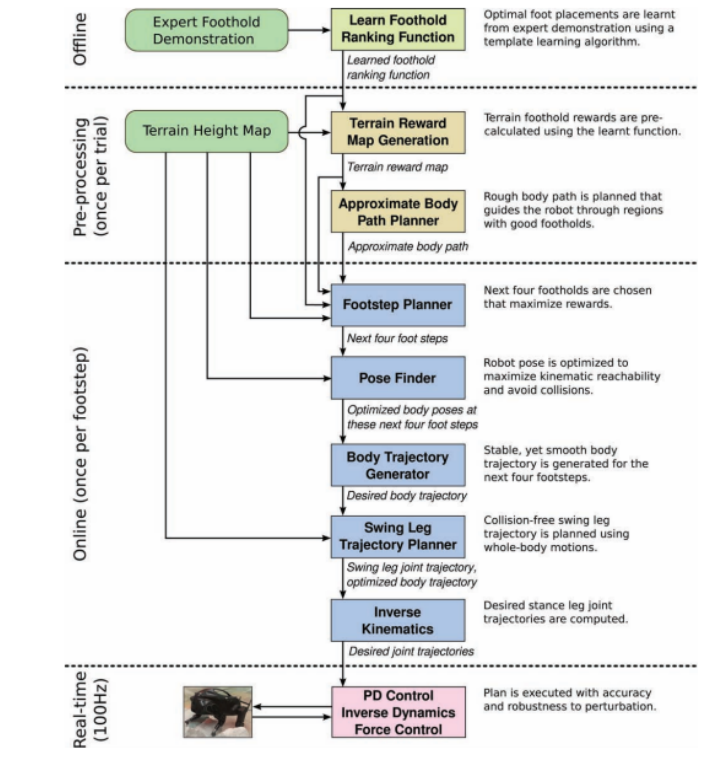


Figura 2.3: Dos esquemas de planificación de caminado desacoplado tradicionales, presentados en [1, 2]. En ambos casos, se fija un modo de caminado, luego se busca una serie de contactos y de allí se genera una trayectoria de CoM y ZMP del robot.

2.3.1. Planificación de contactos

Los métodos de planificación de contactos buscan resolver el problema de planificación de pisadas (Definición 1.1) para un número arbitrario de patas. Por lo general, como se nota en [4] estos métodos suelen dividirse en dos tipos: métodos basados en búsqueda discreta y métodos basados en optimización continua. Aquí presentaremos un breve repaso de ambos métodos.

1. **Búsquedas discretas:** Este tipo de métodos plantean el problema de planificación de contactos como una búsqueda informada sobre el espacio, donde se suele recurrir a un conjunto de acciones discretas pre-definidas y buscar la combinación óptima de estas. En estos casos, se discretiza el ambiente y las regiones de caminado donde luego se definen una serie de zonas de pisado posible, sobre las cuales se itera para obtener el camino óptimo. Estos métodos suelen basarse en algoritmos de búsqueda como A*, ARA*, RRT, o RRT*. La principal ventaja de estos métodos es que son fáciles de implementar y de resolver sin requerir herramientas adicionales. Sin embargo, estos métodos se restringen a un espacio de búsqueda limitado, ya sea por sus conjuntos de acción o por sus zonas de pisado, por lo que el resultado obtenido siempre es sub-óptimo. Por otro lado, de no restringir el espacio de búsqueda de esa forma, el problema de búsqueda se vuelve extremadamente complejo, pues sería de complejidad computacional exponencial $O(2^N)$. Adicionalmente, este tipo de problemas no suelen manejar espacios de búsqueda tridimensionales, no manejan rápidamente espacios cubiertos con obstáculos, y no permiten manejar rotaciones en el cuerpo del robot.
2. **Optimizaciones continuas:** Los métodos de optimización continua se han explorado recientemente como alternativa de complejidad polinomial a los métodos de búsqueda. En particular, Herdt [37] propone una formulación de programación de cuadrática (QP) para buscar contactos y trayectorias de ZMP de forma simultánea. Sin embargo, tal formulación no es capaz de orientar el cuerpo ni evitar obstáculos. En [5] se propone un esquema de programación no-lineal que permite modelar obstáculos y rotaciones de cuerpo, sin dar garantías de optimalidad y siendo altamen-

te susceptible a mínimos locales. Finalmente, [12] propone una formulación basada en programación convexa de enteros mixtos la cual permite incorporar obstáculos y manejar rotaciones. Sin embargo, esta formulación se encuentra limitada a solo robots bípedos y requiere recursos computacionales significativos, debido al uso de restricciones cuadráticas.

2.3.2. Planificación de movimientos

Los métodos de planificación de movimiento han sido mucho menos explorados que los de búsqueda de contacto. En general, estos métodos toman la información del plan de pisadas encontrado y generan una trayectoria de centro de masa que siga dichas pisadas. Para ello, estos métodos asumen un tiempo para cada transición entre pisadas y con ello generan trayectorias de pies. El principal reto en esta etapa, es asegurar que el movimiento generado sea estable, pues la estabilidad dinámica del sistema no es trivial. Métodos tradicionales donde los robots caminan lentamente han sido basados en el criterio de estabilidad estática [20, 38], donde la estabilidad se garantiza cuando el centro de masa se encuentra proyectado dentro del polígono de soporte. Este criterio es equivalente al criterio del ZMP cuando la aceleración es nula, sin embargo, requiere movimientos lentos y terreno plano para que sea válido.

Por otro lado, múltiples planificadores se han implementado para generar trayectorias que respeten el criterio de ZMP, basados en reguladores cuadráticos lineales (LQR) [7, 1, 33, 8, 2]. En estos casos, se busca generar la trayectoria que siga un ZMP deseado, ubicado en el centro del polígono de soporte, y han sido aplicados en múltiples escenarios de caminado dinámico. Sin embargo, estos métodos suelen fallar cuando el caminado se realiza sobre terreno irregular donde el criterio de ZMP pierde validez.

Finalmente, el criterio de estabilidad de CWC ha emergido como una solución general a la generación de caminado en los últimos años. El trabajo realizado en [18, 31], en particular, explora como métodos de optimización pueden emplearse para generar movimientos estables en base a este margen. En [18] se propone una formulación de progra-

mación cuadrática, que permite generar un movimiento robusto y natural, dado un plan de pisadas. En este caso, las no-linealidades correspondientes a la dinámica angular son relajadas al asumir que el centro de masa se encuentra limitado por una región admisible poliedral, sobre la cual se aplica optimización robusta para minimizar el momento angular respecto al centro de masa. Esta formulación es capaz de generar movimientos estable a pesar de irregularidades en el terreno y permite la integración de garantías formales de robustez, gracias a su formulación convexa.

PLANIFICACIÓN DE PISADAS MEDIANTE OPTIMIZACIÓN DE ENTEROS MIXTOS

Este capítulo fue originalmente publicado como el reporte técnico "A Generalized Mixed-Integer Convex Program for Multilegged Footstep Planning on Uneven Terrain" por Bernardo Aceituno, J. Cappelletto, J. C. Grieco y G. Fernández, y puede ser encontrado en arXiv bajo el identificador arXiv:1612.02109v2 [32].

En este capítulo introduciremos una estrategia generalizada de optimización continua para planificación de pisadas en robots multi-patas, la cual es adaptable a diferentes tipos de plataformas, independientemente del numero de patas o geometría de la plataforma. Este método emplea *Programación Convexa de Enteros Mixtos* para representar las restricciones no-convexas asociadas a la orientación de la base y la evasión de obstáculos. El problema de planificación es formulado como un problema de optimización matemática, el cual puede considerar la cinemática y la geometría del robot mediante restricciones lineales y puede ser resuelto de forma eficiente usando software comercial de optimización. Este trabajo representa, al conocimiento de los autores, *la primera implementación exitosa* de un algoritmo de planificación de pisadas para robots multi-patas basado en optimización continua.

3.1. Introducción

Como notamos en el capítulo anterior, la mayoría de las arquitecturas de planificación desacoplan el problema de planificación en dos etapas: una de planificación de pisadas y otra de planificación de trayectoria de caminado. Tal método requiere potentes

herramientas de planificación para obtener planes de pisadas óptimos y realizables en la plataforma. Como es notado por Deits y Tedrake [4], los métodos de planificación de pisadas pueden ser divididos en dos clases principales: *búsquedas discretas y optimizaciones continuas*. Mientras que las herramientas de planificación basadas en búsqueda discreta han sido estudiadas amplia-mente, todas las implementaciones de optimizaciones continuas se han limitado a plataformas bípedas. La principal restricción para adaptar estos métodos a plataformas multi-patas, es el que la representación de la geometría múltipeda y su espacio de trabajo como parte de un problema de optimización resulta en la introducción restricciones no-convexas. Resultando en *programa no-lineal*, el cual por lo general es muy difícil de resolver y no dar garantías en la optimalidad del plan.

3.1.1. Desarrollos relevantes

En el pasado, algoritmos de búsqueda como A* o RRT han sido exitosamente aplicados para el problema de planificación de pisadas, por ejemplo: Zucker et al. [3] usan ARA* con mapas de costos aprendidos del terreno, usado para planear un robot cuadrúpedo LittleDog caminando sobre terreno irregular, como se muestra en la figura 3.1. Por otro lado, Satzinger et al. [22] introdujo una versión modificada del algoritmo A* que permite generar planes de pisadas para un robot cuadrúpedo dado un conjunto de pisadas posibles pre-calculado, usada por el equipo RoboSimian en las finales del DARPA Robotics Challenge (DRC). Otro trabajo de interés es el realizado por Winkler, Mastalli et al. [21] quienes construyen un mapa de costos del terreno para asegurar caminado seguro, para luego planificar usando ARA* para obtener para un robot cuadrúpedo. Estos métodos resultan eficientes de implementar y han sido validados en distintas plataformas robóticas. Sin embargo, estos requieren *conjuntos de búsqueda pre-calculados* y mapas de costos basados en terreno, de forma que se pueda emplear una buena heurística en la búsqueda. Estos por lo general son difíciles de obtener sin un *conocimiento previo del camino* del robot y se ven limitados a una versión *discreta* del ambiente.

Por otro lado, enfoques de planificación con optimizaciones continuas se han restringido a plataformas humanoides, entre ellas: Herdt et al. [37] introducen una formulación

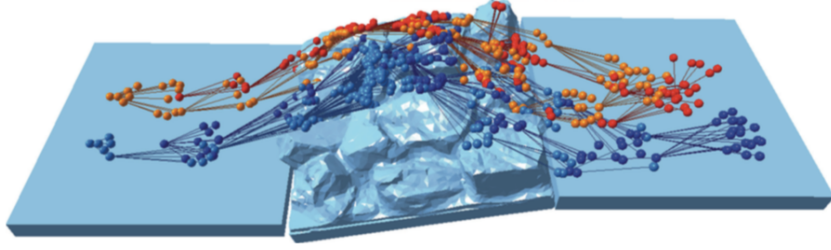


Figura 3.1: Planificación de pisadas basada en búsqueda de grafos, como se muestra en [3]

de Programación Cuadrática (QP) capaz de producir planes de pisadas óptimos y entradas de control para un robot caminante cuando la orientación del caminado está fija y los obstáculos en el ambiente son ignorados. Similar al espíritu de este trabajo, Fallon et al. [5] implementan un algoritmo de programación no-lineal el cual puede planificar pisadas considerando orientación y obstáculos, aunque sin la posibilidad de garantizar optimidad en el resultado y siendo vulnerable a mínimos locales, siendo incapaz de generar planes alrededor de obstáculos. Deits y Tedrake [4] generalizan la implementación anterior al formular el problema como un *Programa Cuadrático con Restricciones Cuadráticas* (MIQCQP) para las finales del DRC, restringiendo las pisadas del plan a estar ubicadas en regiones libres convexas, y usando una *aproximación lineal segmentada* las funciones *sin* y *cos* para planificar orientaciones. Tal método sigue una idea similar al trabajo de Richards et al. [39], donde trayectorias seguras de un vehículo aéreo no-tripulado (UAV) son obtenidas al usar variables binarias que restringen el plan a un conjunto de *regiones convexas seguras* representadas por la intersección de los semi-planos definidos por las caras de los obstáculos.

3.1.2. Estructura del capítulo

Este capítulo presenta un planificador de pisada generalizado basado en optimización convexa para robots multi-patas, el cual es formulado como un *Programa Cuadrático de Enteros Mixtos* (MIQP) que puede ser resuelto de forma eficiente usando software comercial de optimización. Este puede ser fácilmente adaptado a distintas plataformas multi-patas, de forma independiente al número de patas y geometría del robot. El resto del contenido de este capítulo está organizado de la siguiente manera: La sección 3.2 des-

cribe la formulación del problema de optimización para planificación de pisadas propuesto y la sección 3.3 discute brevemente y concluye sobre las contribuciones de este trabajo.

3.2. Formulación Propuesta

La formulación propuesta tiene como meta retornar un arreglo de $N_f + n_l$ pisadas para un robot de n_l patas, donde N_f es el número de pisadas a optimizar, representadas en el espacio cartesiano como:

$$f = (f_x \ f_y \ f_z \ \theta) \quad (3.1)$$

donde f_x , f_y , f_z y θ son las coordenadas xyz del pie y el ángulo yaw de la base del robot al ejecutarla. Entonces, el problema de planificación se formulara como un problema de optimización continua sobre el arreglo de pisadas, sujeto a las siguientes restricciones:

1. Toda pisada debe ser alcanzable desde el estado en que el robot se ubica en la pisada anterior.
2. Todas las pisadas deben ubicarse en una región convexa libre de obstáculos.

Donde asumiremos que la geometría del robot y la segmentación convexa del espacio libre son asumidos como variables de entrada al algoritmo (ilustrado en la figura 1.4).

3.2.1. Restricciones cinemáticas

Debido a la variedad de geometrías que poseen los robots multi-patas, lo que determina el espacio de trabajo de cada pata y la distribución de las patas en el robot, es necesario restringir cada pisadas para satisfacer tal geometría. Por ello, cada pisada es restringida alrededor de una posición nominal, centrada en el espacio de trabajo de la pata. Para simplificar este computo definiremos el Centro de Contactos como:

Definición 3.1 Centro de Contactos: Es el punto c_i ubicado en el centro geométrico de todas las patas en soporte al momento de mover una pata hasta la pisada

f_i .

Algebraicamente:

$$c_i = \frac{\sum_{k \in C(f_i)} f_k}{n_{legs}} \quad (3.2)$$

Donde $C(f_i)$ es el conjunto de patas en soporte al moverse hacia la pisada i . Entonces, dado c para una cierta pisada y la distancia L_{leg} entre ambos, defininemos la posición nominal de la pisada $r_{nom \ i}$ como:

$$r_{nom \ i} = \begin{pmatrix} c_{x \ i} + L_{leg} \cos(\phi_i) \\ c_{y \ i} + L_{leg} \sin(\phi_i) \end{pmatrix} \quad (3.3)$$

Donde ϕ_i entre la orientación de la base θ_i y la posición nominal de la pata. Luego, la pisada es restringida a ubicarse en una región admisible \mathfrak{R} centrada en la posición nominal. En este caso, se escoge una región cuadrada, lo que la hace invariante a rotaciones en la trayectoria (como es mostrado en la figura 3.2):

$$f_i \in \mathfrak{R} \Leftrightarrow f_i \in \begin{pmatrix} r_{nomx \ i} \pm l_{bnd} \\ r_{nomy \ i} \pm l_{bnd} \end{pmatrix} \quad (3.4)$$

Donde l_{bnd} es el lado del cuadrado que limita la región admisible.

La presencia de las funciones \sin y \cos en la ecuación 3.3 hace el problema de optimización no-convexo. Por lo tanto, la solución escogida es representar estas funciones usando aproximaciones lineales segmentadas, manteniendo el problema convexo.

Es importante notar que la distancia nominal L_{leg} debería variar con los ángulos roll y pitch de la base, particularmente cuando mucho mayores a 0. Una posible solución a este problema es representarlo como una relación en la altura de las n_l patas en soporte, resultando en una relación bilineal, la cual puede ser aproximada usando Envoltentes de McCormick, Técnicas de De-segregación Multiparamétrica o diferencia de relaciones cuadráticas. En el caso particular de este trabajo, L_{leg} será fijada como una constante, debido al comportamiento aproximadamente invariante en los ángulos roll y pitch de la

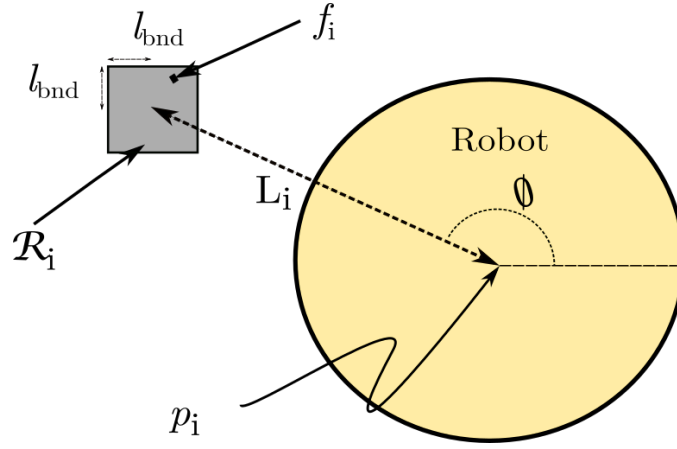


Figura 3.2: Restricción geométrica en f_i , \mathcal{R} y el centro de contactos p_i

base, haciendo todas las restricciones geométricas lineales.

Por otro lado, para que el plan sea factible, cada pisada debe ser alcanzable desde la configuración anterior de soportes. Ya que el espacio de trabajo de cada pata suele resultar en un conjunto no-convexo, es necesario de aproximar de forma que se pueda representar con una restricción convexa. Por ello, cada pisada es restringida a un polígono inscrito en el mayor círculo centrado en r_{nom} y contenido en el espacio de trabajo.

Esta aproximación nos permite representar el alcance de las patas mediante restricciones convexas, independientemente de la forma del robot, como se muestra en la figura 3.3. Por lo tanto, la siguiente restricción lineal es introducida:

$$f_i \in \begin{pmatrix} r_{nom(i-n_{legs})x} \pm d_{lim} \\ r_{nom(i-n_{legs})y} \pm d_{lim} \end{pmatrix} \quad (3.5)$$

donde d_{lim} representa los límites del conjunto convexo cuadrado, calculados como en Rojas et al. [20].

El centro del espacio de trabajo será ubicado en la posición nominal de la pata, lo que permite considerar cambios en el alcance debido a los movimientos de la base. adicionalmente, cada movimiento de ascenso y descenso también será restringido. Lo que se puede representar con la siguiente restricción lineal:

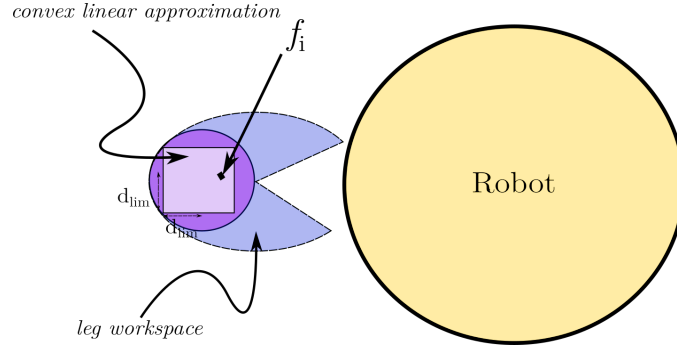


Figura 3.3: Aproximación lineal y convexa del alcance de la pata

$$|f_{iz} - f_{(i-n_{legs})z}| \leq \Delta z_{max} \quad (3.6)$$

Donde Δz_{max} es la máxima variación de altura posible.

Otras formulaciones del alcance de la pata también fueron consideradas. Por ejemplo, Deits y Tedrake. [12] representan el espacio de trabajo como un círculo centrado en la posición nominal de la pata, lo que se puede introducir como una restricción cuadrática en el programa. Este método entregó resultados muy similares al presentado anteriormente, con la ventaja de ser invariante a las rotaciones. Sin embargo, esta representación aumenta de forma significativa la complejidad del problema al añadir restricciones cuadráticas. En nuestro caso, la representación cuadrada también resulta en una aproximación invariante a las rotaciones bajo mucha menor complejidad computacional.

3.2.2. Evasión de obstáculos

Restringir el plan para evadir obstáculos, o considerar terreno irregular hace el problema no-convexo. Una solución a este problema, es restringir todos los puntos del plan a ubicarse en un conjunto de *regiones seguras y convexas*. Estas regiones pueden ser calculadas fácilmente empleando algoritmos de percepción o usando el algoritmo IRIS [40] el cual emplea programación semi-definida [23] para calcular las regiones convexas libres de obstáculos en el ambiente. Cada región R es entonces representada por siguiente convex hull:

$$R = \{x \in \mathbb{R}^3 \mid A_r x \leq b_r\}$$

La asignación de las pisadas del plan a cada región convexa es realizada empleando variables de decisión binarias. Para ello, una matriz binaria $\mathcal{H} \in \{0, 1\}^{N \times N_r}$ es definida, donde N_r es el numero de regiones convexas. Luego, cada pisada es asignada a una sola región segura, usando la siguiente restricción lineal de enteros-mixtos:

$$\sum_{r=1}^{N_r} \mathcal{H}_{ir} = 1 \quad (3.7)$$

Mientras que tal restricción no signifique que estas regiones no se pueden intersectar, esto reduce la complejidad del problema de optimización al solo asignar una de las regiones a cada pisada. Entonces, Todas las pisadas son restringidas a una región convexa al añadir la siguiente restricción lineal:

$$\mathcal{H}_{ir} \Rightarrow A_r f_i \leq b_r \quad (3.8)$$

Donde el operador \Rightarrow (implica) se puede modelar en Programación de Enteros-Mixtos como una restricción lineal usando formulación de M-grande [24]. Esta restricción asegura que la pisada i se ubica en la región r .

3.2.3. Orientación de la base

Como mencionamos anteriormente, nuestro método emplea *aproximaciones lineales segmentadas* de las funciones *sin* y *cos* para mantener la convexidad del problema, como se muestra en la figura 3.4. Como en [4], dos variables auxiliares s y c son definidas para representar las funciones $\sin(\theta)$ y $\cos(\theta)$ mediante dos matrices binarias \mathcal{S} y \mathcal{C} que asignan cada variable a un segmento lineal de la aproximación. Esto se puede representar mediante las siguientes restricciones de enteros-mixtos:

$$\mathcal{S}_{ik} \Rightarrow \begin{cases} \psi_{k-1} \leq \theta_i \leq \psi_{k+1} \\ s_i = m_k \theta_i + n_k \end{cases} \quad (3.9)$$

$$\mathcal{C}_{ik} \Rightarrow \begin{cases} \gamma_{k-1} \leq \theta_i \leq \gamma_{k+1} \\ c_i = m_k \theta_i + n_k \end{cases} \quad (3.10)$$

donde los ángulos ψ y γ representan las fronteras entre segmentos lineales, mientras que m y n representan su pendiente e intersección. Luego, se restringe que cada variable

auxiliar se ubiquen en un solo segmento de la aproximación. Por lo tanto para que pisada f_i :

$$\sum_{s=1}^{N_s} \mathcal{S}_{is} = 1 \quad (3.11)$$

$$\sum_{s=1}^{N_s} \mathcal{C}_{is} = 1 \quad (3.12)$$

donde N_s es el numero de segmentos de la aproximación lineal por partes.

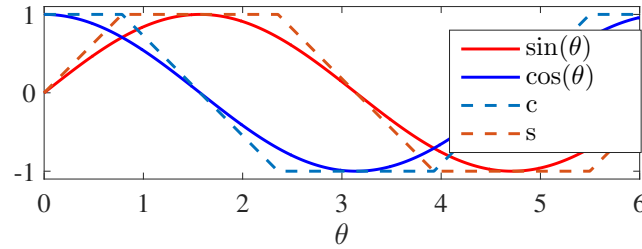


Figura 3.4: Aproximación lineal segmentada de las funciones \sin y \cos , como se muestra en [4]

Ya que la introducción de variables binarias hacen el problema de optimización NP-completo [41], y la presencia de multiples patas aumenta el numero de variables binarias significativamente, es deseable reducir el tamaño de las matrices de asignación \mathcal{C} y \mathcal{S} . Por ello, se resuelve restringir la orientación de la base en cada conjunto de n_i pisadas para a ser el mismo (θ), asumiendo que el torso del robot no rota significativamente al desplazarse de una pisada a la siguiente, y luego emplear relaciones lineales para obtener el ángulo ϕ requerido en el computo del centro de contactos de cada pata.

3.2.4. Duración del plan

Finalmente, ya que el numero de pisadas requerido para completar el plan no es conocido *a priori*, la solución elegida es restringir el tamaño del plan a un numero máximo de pisadas $N_f + 4$, y luego recortar las pisadas innecesarias en el plan resultante. Para esto,

una variable binaria de recorte t es introducida, con la finalidad de indicar si una pisadas debe ser recortada del plan, y luego agregar la siguiente restricción de enteros-mixtos:

$$t_i \Rightarrow f_i = g_{leg(i)} \quad (3.13)$$

donde $leg(f) : \mathbb{Z} \rightarrow \{1, 2, \dots, n_{legs}\}$ es una función que retorna el indice de la ultima pisada para la pata con pisada f .

3.2.5. Objetivos

Dadas la restricciones planteadas anteriormente, el problema se formula como un programa de minimización multi-objetiva, con los siguientes objetivos:

1. Distancia final a la meta
2. Numero de pisadas en el plan
3. Separación relativa entre pisadas

El primer objetivo introduce un costo cuadrático sobre la diferencia entre las n_l pisadas finales f_g y las pisada meta g , formulado como:

$$(f_g - g)^T Q_{goal} (f_g - g) \quad (3.14)$$

Donde Q_{goal} es una matriz de costo diagonal. Luego, para minimizar el numero de pisadas en el plan se escoge maximizar la cantidad de pisadas a recortar. Esto se puede lograr asignando un costo negativo a la suma total de las variables de recorte, lo se representa algebraicamente como:

$$\sum_{k=1}^N q_t t_k \quad (3.15)$$

donde q_t es un peso negativo sobre la variable de recorte t .

Luego, para minimizar la distancia relativa entre pisadas introducimos un costo cuadrático sobre la separación entre los centros de contacto de cada pisada, representado como la siguiente suma pesada:

$$\sum_{k=n_{legs}}^N (c_k - c_{k-n_{legs}})^T Q_r (c_k - c_{k-n_{legs}}) \quad (3.16)$$

donde Q_r es un peso positivo sobre el costo de desplazamientos relativos en el plan.

En nuestra implementación, el mayor peso se asigna a la distancia relativa con la meta, seguido por los pesos de recorte de pisadas y la separación relativa entre pisadas.

3.2.6. Formulación completa

El MIQP de planificación de pisadas completo, con todas las restricciones y objetivos, para n_l patas, se representa como:

$$\begin{aligned} \text{minimizar} \quad & (f_g - g)^T Q_g (f_g - g) + \\ & \sum_{k=1}^N q_t t_k + \sum_{k=n_l+1}^N (c_k - \\ & c_{k-n_l})^T Q_r (c_k - c_{k-n_l}) \end{aligned}$$

sujeto a:

- *Restricciones cinemáticas:*

$$c_i = \frac{\sum_{k \in C(f_i)} f_k}{n_l}$$

$$r_{nom} = \begin{pmatrix} c_x + L_{leg} c_\phi \\ c_y + L_{leg} s_\phi \end{pmatrix}$$

$$f_i \in \begin{pmatrix} r_{nomx} \pm l_{bnd} \\ r_{nomy} \pm l_{bnd} \end{pmatrix}$$

- *Alcance en XYZ:*

$$f_i \in \begin{pmatrix} r_{nom(i-n_l)x} \pm d_{lim} \\ r_{nom(i-n_l)y} \pm d_{lim} \end{pmatrix}$$

$$|f_{iz} - f_{(i-n_l)z}| \leq \Delta z_{max}$$

- *Evasión de obstáculos:*

$$\sum_{r=1}^{N_r} \mathcal{H}_{ir} = 1$$

$$\mathcal{H}_{ir} \Rightarrow A_r f_i \leq b_r$$

- *Aproximación lineal segmentada de sin-cos:*

$$\mathcal{S}_{ik} \Rightarrow \begin{cases} \phi_{k-1} \leq \theta_i \leq \phi_{k+1} \\ s_k = m_k \theta_i + n_k \end{cases}$$

$$\mathcal{C}_{ik} \Rightarrow \begin{cases} \gamma_{k-1} \leq \theta_i \leq \gamma_{k+1} \\ c_k = m_k \theta_i + n_k \end{cases}$$

- *Recorte de pisadas:*

$$t_i \Rightarrow f_i = g_{leg(i)}$$

donde $f_g = [f_N, f_{N-1}, \dots, f_{N-n_l+1}]$ son las n_l pisadas finales del plan y $leg(f) : \mathbb{Z} \rightarrow \{1, 2, \dots, n_l\}$ es una función que recibe un índice de pisada y retorna su número de pata correspondiente.

3.3. Conclusiones

En este capítulo hemos introducido un enfoque de optimización continua para el problema de planificación de pisadas. El método propuesto hace uso de programación de enteros mixtos para reemplazar las restricciones no-lineales asociadas al caminado multi-patas como restricciones convexas. Más adelante, este método será implementado en Drake [42], por lo que representa la **primera implementación exitosa** de un planificador de pisadas multi-patas generalizada basado en optimización convexa.

En el capítulo 5, mostraremos una serie de simulaciones de este método para manejar escenarios complejos de forma exitosa, esto incluye casos con rotaciones complejas y ambientes irregulares. De aún mayor interés, resulta el manejo eficiente de estos problemas, siendo resueltos por un computador de gama-baja en tiempo mucho menores a un segundo. Por lo tanto, futuras extensiones al algoritmo propuesto en este capítulo pueden hacer uso de esta capacidad de cómputo para la integración de planificación de pisadas en tiempo interactivo, al estilo de un Control Predictivo de Modelo (MPC) para planificación de caminado en terrenos complejos.

PLANIFICACIÓN DE CAMINADO ROBUSTO MEDIANTE OPTIMIZACIÓN CONVEXA

Este capítulo fue originalmente publicado como "A Mixed-Integer Convex Optimization Framework for Robust Multi-Legged Locomotion Planning over Challenging Terrain" por Bernardo Aceituno, H. Dai, J. Cappelletto, J. C. Grieco y G. Fernández, y puede ser encontrado en los proceedings de la IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2017). Vancouver, 2017 [43].

Este capítulo presenta un esquema basado en optimización convexa para la planificación de caminado robusto. La mayoría de los enfoques tradicionales se basan en el criterio de Zero Moment Point (ZMP) para garantizar estabilidad dinámica y fijan el modo de caminado como constante. Mientras que esta combinación funciona correctamente en terreno plano, no se generalizan a terreno irregular para el cual se requieran transiciones agresivas o cambios de modo de caminado. Para resolver este problema, se propondrá un esquema de optimización convexa complementario al presentado en el capítulo 3, el cual puede planificar pisadas junto a transiciones el modo de caminado. Adicionalmente, haremos uso del criterio de estabilidad de Cono de Llave de Contactos (CWC), el cual se generaliza a terreno irregular con restricciones de cono de fricción, para planificar trayectorias robustas de centro de masa, junto al momento angular a través de un programa de optimización convexa. Este método es capaz de generar movimientos robustos en terreno arbitrariamente complejos, incluyendo aquellos donde un planificador de ZMP fallaría.

4.1. Introducción

Las mas plataformas robóticas multi-patas representan herramientas poderosas para acceder a ambientes complejos e irregulares [6, 8, 7]. Sin embargo, hay dos retos que emergen para estas plataformas 1) El número de modos de caminado posibles crece de forma exponencial con el numero de patas; 2) no es trivial asegurar que el movimiento planeado sea realizable de forma dinámica, que el robot no caiga al caminar. Anteriormente, investigadores han usado Generadores de Patrones Centrales (CPG) para planificar el modo de caminado [8, 44, 45, 14], y se basan en el criterio de Zero Moment Point (ZMP) para asegurar la estabilidad dinámica del movimiento [26]. Mientras que los CPG y el criterio del ZMP funcionan bien para locomoción en terreno plano, ninguno de ellos se adapta naturalmente a terreno irregular. En terreno irregular, los robots deben ajustar su caminado de forma frecuente para adaptarse a cambios en el terreno, aunque donde los CPG se adaptan para caminados de transición. Por otro lado, aunque el Zero Moment Point se encuentra definido para terreno plano, este no es aplicable a terreno irregular con restricciones de conos de fricción. Por lo tanto generalizaremos enfoques anteriores basados en terrenos plano, planificando el modo de caminado en conjunto a la secuencia de pisada usando Programación Convexa de Enteros Mixtos, y garantizaremos la estabilidad dinámica del caminado empleando el criterio del Cono de Llave de Contactos (CWC o Contact Wrench Cone).



Figura 4.1: LittleDog caminando sobre terreno irregular con contactos no co-planares y conos de fricción linealizados (azul)

4.1.1. Desarrollos Relevantes

Como se mencionó en el capítulo 3, múltiples trabajos estudiado la aplicación de Programación de Enteros Mixtos para planificar caminado y secuencias de pisada para robots caminantes [4, 35]. En los trabajos [4, 32] y en el capítulo anterior, primero se obtiene un conjunto de regiones convexas libres de obstáculos y luego emplean variables binarias para asignar cada contacto a una de estas. En este capítulo extenderemos estos trabajos y método propuesto en capítulo anterior para manejar transiciones en los modos de caminado. Aunque, en comparación a robot bípedos, los cuales solo presentan un modo de caminado, el cual alterna entre pierna izquierda y pierna derecha; los robots multi-patas presentan la complejidad adicional de que el número de secuencias de caminado realizables incrementa de forma exponencial con el número de patas. En este capítulo extenderemos los trabajos anteriores para la planificación de pisadas usando programación de enteros mixtos para optimizar modos de caminado de forma eficiente.

Por otro lado, a la hora de garantizar estabilidad dinámica en el caminado, la mayoría de los enfoques tradicionales se basan en el criterio del Zero-Moment Point (ZMP), el cual requiere que el centro de presión se encuentre dentro del polígono de soporte del robot en todo momento. Sin embargo, la región de soporte y el centro de presión no se encuentran bien definidos cuando los contactos no son co-planares o el terreno es irregular; además de esto, el criterio de ZMP ignora las restricciones de cono de fricción, por lo que el pie puede resbalar en la ausencia de suficiente fricción en el terreno. Por estas razones, el criterio de estabilidad del *Contact Wrench Cone* (CWC) o Cono de Llave de Contactos [15, 17] emergió como una generalización del criterio del ZMP para cualquier tipo de terreno, incluso en el caso de contactos no co-planares y bajo coeficiente de fricción, al forzar que el *wrench* o llave, concatenación de fuerza y momento, agregado sea parte de conjunto admisible definido por los conos de fricción proyectados al espacio hexa-dimensional. Una representación gráfica de este criterio es presentada en [17] y se muestra en la figura 4.2.

Más recientemente, las herramientas de optimización convexa han sido empleadas de forma exitosa para maximizar la robustez de las trayectorias de caminado de forma

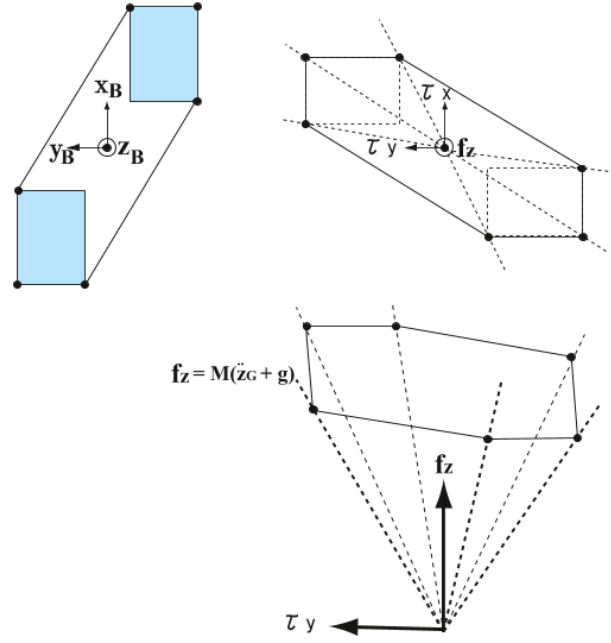


Figura 4.2: Representación gráfica del *wrench* agregado (vectores f y τ) dentro del cono de llave de contactos proyectado en el plano y su relación con el ZMP.

eficiente y universal empleando este criterio para el caso de robots humanoides atravesando terreno irregular [31, 18], logrando proveer *garantías formales* de robustez en terrenos donde el criterio de ZMP fallaría, haciendo al robot resbalar y caer. Concretamente, Dai y Tedrake [18] logran mejorar la robustez de trayectorias de caminado mientras se mantiene la naturalidad del movimiento de cuerpo, al optimizar la trayectoria del caminado una vez que las planes de pisadas son generados por una etapa anterior. Sin embargo, los métodos anteriormente se encuentran restringidos a plataformas bipedas de contactos planos, las cuales se encuentran restringidas un modo de caminado único.

4.1.2. Contribución y estructura del capítulo

La principal contribución de este capítulo es la introducción de un esquema de planificación de caminado robusto robots *robots multi-patas complejos*, ofreciendo garantías formales de robustez aún en movimientos de contacto mínimo sobre terrenos irregulares. En este capítulo formulamos el problema de planificación como una optimización convexa, la cual permite generar: secuencias de contactos con modos de caminado adaptativos al tipo de terreno y movimientos robustos, basados en el criterio de estabilidad del CWC.

El resto de este capítulo se encuentra estructurado de la siguiente manera. La sección II introduce los conceptos principales del criterio de CWC y su asociado margen de estabilidad. La sección III presenta la formulación del programa de optimización del esquema propuesto, Finalmente, la sección IV discute y concluye sobre las contribuciones del capítulo.

4.2. Enfoque técnico

En esta sección propondremos un esquema de planificación de movimiento de caminado robusto, desacoplado en planificación de modo de caminado y movimiento corporal. Por lo tanto, abordaremos el problema en dos etapas:

1. Modos de caminado que se adaptan a las irregularidades del terreno.
2. Trayectorias de centro de masa y momento angular que maximizan la robustez del movimiento.

De aquí en adelante, se asumiremos que el modo de caminado es planificado como parte del plan de pisadas, dentro de un solo problema de optimización, el cual estará ordenado de forma secuencial por cada pata del robot.

4.2.1. Optimización de modo de caminado

Para lograr optimizar modos de caminado que se adapten al tipo de terreno, propondremos una formulación Programación de Enteros Mixtos [24]. Tal formulación puede trabajar como parte de planificadores de pisadas basados en optimización convexa [4, 32] o con plan de pisadas dado. Nuestra meta es generar la secuencia de movimientos de cada pata, minimizar la duración del plan, y asignar soportes acorde a heurísticas de complejidad en el terreno.

1) *Formulación:* Las secuencias de caminado son generalmente representadas por diagramas de fase [13], los cuales indican los n_l soportes de contacto para una fase dada (eje horizontal), como se muestra en la figura 4.3. Inspirados en esta representación, buscaremos

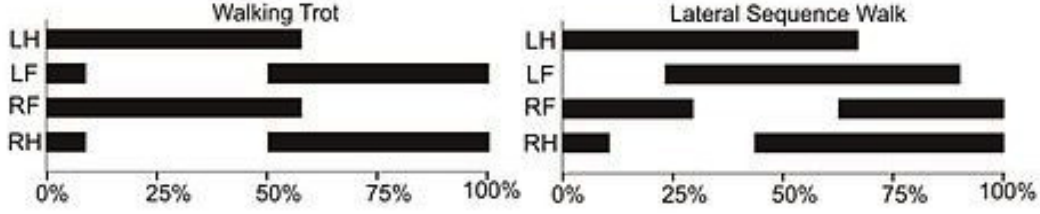


Figura 4.3: Diagramas de fase de dos modos de caminado cuadrúpedo, donde el eje horizontal representa el porcentaje de ciclo de transferencia ejecutado en cada fase, y el eje vertical los soporte frontales 'F' y traseros 'H' (LF, RF, LH, RH).s

describir la fase de soporte asignada a cada pata en un ciclo de transferencia del plan de pisada. Para ello, introduciremos una matriz binaria $T \in \{0,1\}^{N_f \times N_f}$, donde $T_{ij} = 1$ implica que la pisada i -ésima será alcanzada en fase de soporte j -ésima, donde N_f es la longitud del plan. Como cada pisada solo debe ser alcanzada una vez en el plan, introduciremos la siguiente restricción sobre la matriz T :

$$\sum_{j=1}^{N_f} T(i, j) = 1, \quad \forall i = 1, \dots, N_f \quad (4.1)$$

Al restringir el problema de esta forma, la matriz T equivale a un diagrama de fase, ya que el plan de pisadas se ordena de forma secuencial según el número de pata. Mas allá, requeriremos que cada ciclo de n_l pisadas sea alcanzado antes de que inicie el siguiente ciclo de transferencia. Por lo tanto, introduciremos un vector $t \in \mathbb{Z}^{N_f}$ que calcule la fase asignada a cada transferencia de pata, obtenido como:

$$t = T \begin{pmatrix} 1 \\ \vdots \\ N_f \end{pmatrix}$$

luego, ya que el modo de caminado sigue un plan de pisadas ordenado en secuencia, introduciremos la siguiente restricción:

$$t_j > t_{j-n_l}, \quad \forall j = n_l + 1, \dots, N_f \quad (4.2)$$

Esto asegura que el modo de caminado siempre finalice un ciclo de transferencia antes de comenzar el siguiente.

2) *Incorporación de heurísticas de terreno:* Asumiendo que cada pisada f es representada por su posición xyz , en la forma $(f_x \ f_y \ f_z)$, y que el centro de contactos p de cada n_l conjunto de contactos es calculado como:

$$p_j = \frac{\sum_{j \in C\{f_i\}} f_j}{n_l}$$

Donde $C\{f_i\}$ es el conjunto de n_l patas en soporte con la pisada i -ésima. Para lograr adaptar el modo de caminado al tipo de terreno, definiremos la desviación de altura de pisada como métrica de irregularidad, entre la i -ésima pisada y el ciclo de transferencia anterior, como:

$$\Delta z_i = |f_{zi} - p_{z,i-n_l}|, \quad \forall i = n_l + 1, \dots, N_f$$

En este caso, buscaremos penalizar los movimientos agresivos cuando Δz_i sobrepase un cierto umbral γ_z . Para ello, introduciremos dos variables binarias $\delta_{z,i}^+, \delta_{z,i}^-$, con la restricción:

$$\delta_{z,i}^+ = 1 \Rightarrow f_i - p_{i-n_l} \geq \gamma_z \quad (4.3)$$

$$\delta_{z,i}^- = 1 \Rightarrow f_i - p_{i-n_l} \leq -\gamma_z \quad (4.4)$$

$$\delta_{z,i} = (\delta_{z,i}^+ + \delta_{z,i}^-) \gamma_z \quad (4.5)$$

La variable $\delta_{z,i}$ solo puede tomar dos valores, 0 ó γ_z . Por lo tanto, vamos a imponer un costo en la variable $\delta_{z,i}$ para penalizar cambios bruscos en el movimiento.

Restricciones adicionales pueden ser agregadas dependiendo de la geometría del robot, ya que algunas configuraciones de soporte pueden ser imposible (por ejemplo, caminar sobre un soporte) o si se desea guiar el modo de caminado hacia una secuencia deseada.

3) *Objetivos:* Como explicamos anteriormente, esta formulación tiene dos objetivos a optimizar:

1. Minimizar duración del plan.
2. Maximizar numero de contactos (minimizar el numero de patas en transferencia) acorde al tipo de terreno.

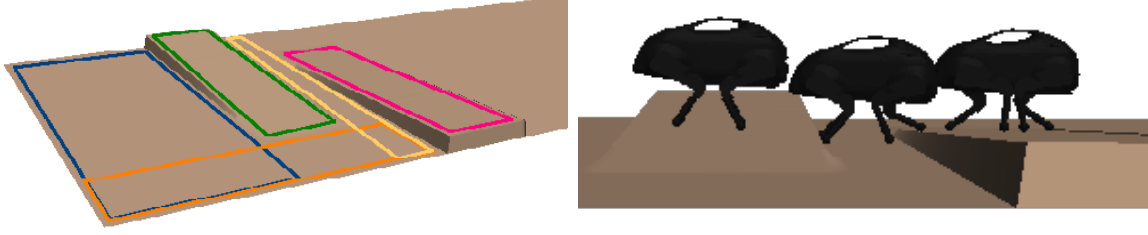


Figura 4.4: Izquierda: Ejemplo de terreno segmentado, con conjuntos de terreno admisibles, dados como entrada al planificador de pisadas. Derecha: Ejemplo de un modo de caminado optimizado.

Para el primer objetivo, se puede simplemente minimizar la suma de todas las fases asignadas, equivalente a minimizar el numero de fases utilizadas en el plan, algebraicamente:

$$\sum_{j=1}^N c_t t_j \quad (4.6)$$

donde c_t es un peso positivo. Por otro lado, para el segundo objetivo debemos relacionar la matriz de modo de caminado T con el vector de irregularidad de terreno δ_z . Esto se puede realizar fácilmente al realizar el producto entre ambas variables. Por lo tanto, definiremos un vector de costo u tal que:

$$u = \delta_z^T T$$

Este producto proporciona información sobre si una fase específica va a resultar en una configuración sobre terreno irregular durante la transición, y por lo tanto buscaremos minimizarla. Sin embargo, cada j -esimo elemento del vector u representa una relación bilineal entre δ_{zj} y T_{ji} , lo que hace al problema *no convexo*. Afortunadamente, que la matriz T es binaria, podemos fácilmente reemplazar el producto de matrices por una variable auxiliar β , tal que $u_j = \sum_{i=1}^{N_f} \beta_{ji}$, el cual puede ser calculado como:

$$\beta_{ji} = \delta_{zj} T_{ji} = \begin{cases} \delta_{zj} & \text{if } T_{ji} = 1 \\ 0 & \text{if } T_{ji} = 0 \end{cases}$$

Esta operación condicional se puede representar en software de optimización de problema enteros-mixtos usando formulación de M-grande [24]. Luego agregamos un coso cuadrático a cada elemento de u :

$$\sum_{j=1}^{N_f} u_j^T Q_u u_j \quad (4.7)$$

donde Q_u es un costo de matriz positiva semi-definida. Este costo asegurará que el planificador asigne mas soportes durante la transferencia cuando el terreno resulta muy irregular, mientras que busque el caminado más rápido cuando el terreno sea relativamente plano. Adicionalmente, si este costo es muy alto, el planificador de pisadas se verá parcializado hacia caminados conservativos sobre regiones planas.

3) *Formulación de planificación de pisadas extendida:* El programa de optimización de enteros mixtos para planificación de pisadas extendido, sobre $N_f + n_l$ pasadas, basado en [4, 32] y el presentado en el capítulo anterior, es formulado como:

$$\begin{aligned} \text{minimizar } f, T, H, S, C \quad & (f_g - g)^T Q_g (f_g - g) + \\ & \sum_{j=1}^{N_f} u_j^T Q_u u_j + \sum_{k=n_l+1}^{N_f+4} (p_k - \\ & p_{k-n_l})^T Q_r (p_k - p_{k-n_l}) \\ & + \sum_{j=1}^{N_f} c_t t_j \end{aligned}$$

sujeto a:

1. *Restricciones cinemáticas.*
2. *Asignación de regiones seguro.*
3. *Aproximación lineal por partes de sin-cos.*
4. *Restricciones de caminado.*
5. *Restricciones de terreno.*

Donde f es el plan de pisadas, f_g son las n_l pisadas finales del plan, g es la posición meta, p_j es el centro de contacto en para la pisada j -esima, y Q_g , Q_r , Q_u , q_t y c_t son costos positivos. La matriz H es usada para asignar pisadas a regiones convexas libres de obstáculos, como se muestran en la figura 4.4, las cuales pueden ser obtenidas usando el algoritmo IRIS [40], mientras que las matrices S y C asignan las funciones trigonométricas de la orientación de la base a aproximaciones lineales.

Aquí, la programación convexa de enteros mixtos es usada para representar las restricciones asociada a evasión de obstáculos y orientación corporal como *restricciones convexas*.

4.2.2. Planificación de movimientos robustos

Esta sección presenta una formulación de optimización convexa para generar planes de centro de masa y momento angular, propuesta inicialmente en [18]. Aquí resumiremos la formulación de [18], la cual maximiza el margen de CWC y minimiza momento angular centroidal angular. En esta sección, extenderemos esta formulación para robots multi-patas, mientras que [18] se limita a robots bípedos.

1) *Margen de Cono de Llave de Contactos:* Para poder maximizar el margen de CWC, debemos representar su computo como una restricción convexa. Usando la representación presentada en la ecuación (2.10) podemos calcular el margen en cada instante temporal usando la siguiente restricción:

$$\begin{aligned} & \underset{\epsilon}{\text{mín}} - \epsilon \\ \text{s.t. } & \epsilon_L \leq \epsilon \leq -\bar{a}_i \begin{bmatrix} m\ddot{r} - m\mathbf{g} \\ \dot{k}_O - r \times m\mathbf{g} \end{bmatrix} \end{aligned} \quad (4.8)$$

donde $\epsilon_L \geq 0$ es la cota inferior de margen requerido. Usando esta representación podemos efectivamente maximizar el margen de CWC en cada instante de tiempo.

2) *Momento Angular Centroidal:* Generalmente, los movimientos de caminado natural tienden a regular el momento angular centroidal k_G (momento angular respecto al CoM). Por lo tanto, ya que deseamos generar movimientos naturales usando restricciones convexas, buscaremos minimizar la norma L_1 de esta variable $|k_G|_1$, la cual se puede calcular en cada instante como:

$$k_G = k_O - mr \times \dot{r} = k_O + m[\dot{r}]_{\times} r \quad (4.9)$$

y $|k_G|_1 = \max_j \omega_j k_G$, donde ω_j es un vector de peso para obtener el valor absoluto, tal que: $\omega_j = [\pm 1, \pm 1, \pm 1]$. Desafortunadamente, el producto cruz necesario para calcular (4.9) es una relación bilineal entre r y \dot{r} , lo convierten en una restricción no-convexa. Para evitar esto, vamos a tratar la variable r como una variable *incierto* dentro de un conjunto admisible, definido por las pisadas, y trataremos el problema usando teoría de

optimización robusta [25]. En este caso, asumiremos que el conjunto admisible del CoM es una elipsoide B_r , con matriz radial Q_r y centrada en el punto r^* , como se muestra en la figura 4.5.

$$|k_G|_1 \leq \omega_j k_O + m\omega_j [\dot{r}]_{\times} r^* + m\sqrt{\omega_j [\dot{r}]_{\times} Q_r^{-1} [\dot{r}]_{\times}^T \omega_j^T} \leq s$$

Al añadir esta restricción podemos minimizar la cota superior s de la representación no-convexa de (4.9). En este caso, las relaciones bilineales son relajadas y el producto cruz es representado como una restricción de *conos de segundo orden*. Por lo tanto, el problema se hace convexo y puede ser resuelto de forma eficiente software de optimización comercial.

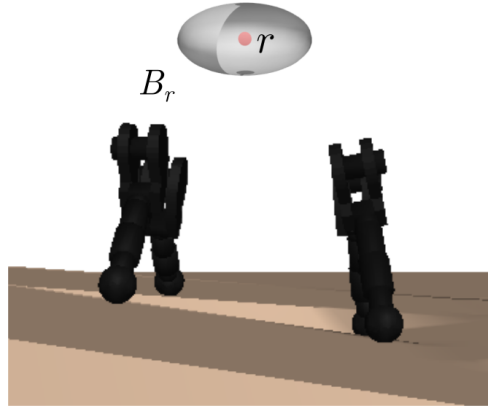


Figura 4.5: Conjunto elipsoidal de incertidumbre B_r (gris) de la posición del centro de masa r (rojo)

3) *Objetivos*: Siguiendo el ejemplo de [18], esta formulación introduce tres objetivos a minimizar en cada instante de tiempo:

- Minimizar la cota superior de $|k_G|_1$.
- Minimizar la aceleración de centro de masa.
- Maximizar el margen de CWC

Por lo tanto, la función de costo del planificador será:

$$\min_{r, k_O} J = \sum_{i=1}^N c_k s^T s + c_a \ddot{r}^T \ddot{r} - c_\epsilon \epsilon \quad (4.10)$$

Donde c_k , c_a y c_ϵ son pesos positivos sobre cada costo. Esta formulación, junto a las restricciones presentadas anteriormente, hacen del problema un *Programa de Conos de Segundo Orden* [23], el cual es un problema de optimización convexa especial.

4.3. Conclusión

En esta capítulo hemos presentado un esquema de planificación de locomoción de caminado robusto, capaz de producir planes de caminado que se adaptan a la irregularidad del terreno y trayectorias de centro de masa que maximizan el margen de CWC y minimizan el momento angular centroidal del movimiento. En el siguiente capítulo vamos a verificar la implementación de este esquema en un cuadrúpedo LittleDog; donde buscaremos generar modos de caminado de transición y movimientos robustos en situaciones donde planificadores clásicos basados en el criterio de **Zero-Moment Point** fallarían.

SIMULACIONES Y RESULTADOS

En este capítulo, presentaremos los resultados del esquema propuesto en una serie de simulaciones dinámicas sobre plataformas robóticas multi-patas trayendo terrenos complejos.

5.1. Plataformas y ambiente de simulaciones

La eficiencia de este enfoque basado en optimización y sus ventajas sobre métodos tradicionales serán validados de mediante simulaciones sobre dos plataformas robóticas multi-patas: un cuadrúpedo LittleDog de *BostonDynamics* © quadruped y un hexápodo BH3-R de *Linxmotion* ©. Para emplear nuestro método, realizaremos simulaciones dinámicas en MATLAB 2015b, usando Drake: Toolbox para planificación y control desarrollado por el MIT [42]. Para las secciones de optimización usaremos dos paquetes de software de optimización convexa con licencia académica: Gurobi [46] para manejar los problemas de enteros-mixtos y Mosek 7 [47] para resolver problemas cónicos de segundo orden. Todas las pruebas son realizadas sobre un computador de procesador de *gama-baja Intel Core 2 Quad* configurado con reloj de 2.4 Ghz, con poder de procesamiento similar a aquellos usados para procesamiento *on-board*.

5.2. Planificador de Pisadas

Ya que la complejidad del problema de optimización aumenta significativamente con el numero de variables enteras, las pruebas de mayor complejidad se realizaran calculando conjuntos de 4 n_{legs} pisadas y luego serán concatenados para obtener el plan final.

5.2.1. Hexápodos

Todas las pruebas iniciales son realizadas sobre un hexápodo circular BH3R, con una cota superior de 72 pisadas por plan. La primera prueba se realiza sobre un camino de piedras, donde el planificador debe buscar la ruta optima y la secuencia de pisadas, con 13 regiones convexas y una meta localizadas 2 metros delante de la posición inicial del robot con una rotación de 45° respecto a la base. El problema es resuelto en cuestión de pocos segundos, retornando el plan presentado en la figura 5.1.

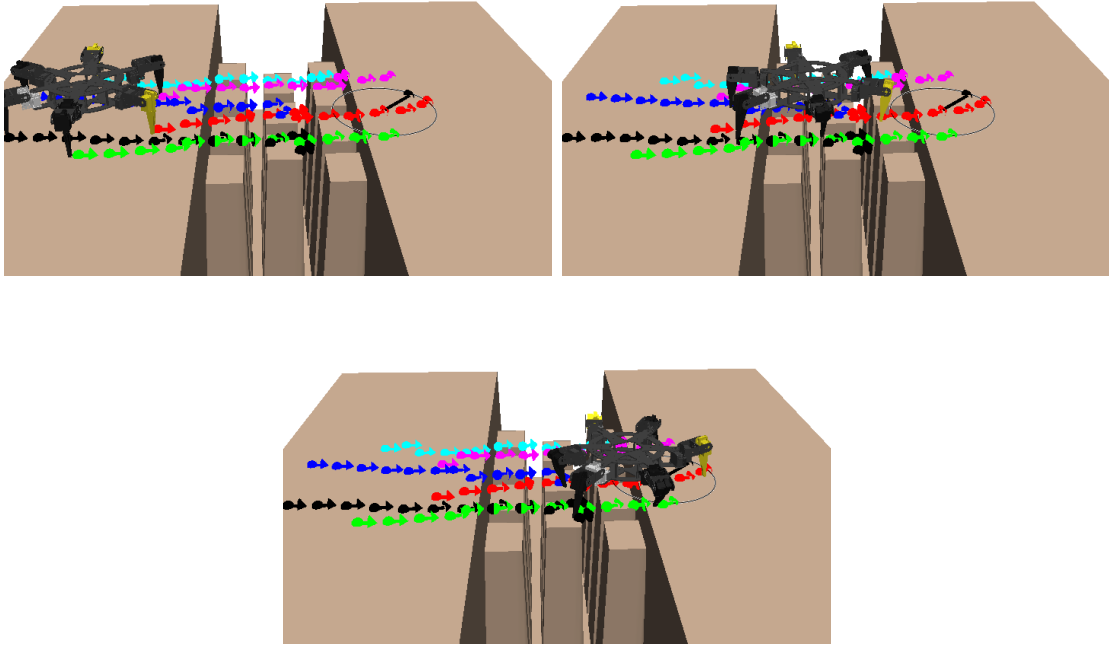


Figura 5.1: A *BH3R* hexápodo siguiendo un plan de pisadas generalizado sobre un camino de piedras.

Luego, verificaremos la capacidad del planificador de manejar rotaciones complejas en el plan, mientras que se respetan las restricciones cinemáticas del robot. Para ello, buscaremos ubicaremos la meta en frente origen y con una rotación de 90° respecto al robot, lo que resulta en el plan mostrado en la figura 5.2.

Finalmente, verificaremos la capacidad del planificador para manejar terreno irregular, algo difícilmente realizable empleando métodos tradicionales. Por lo tanto, se buscara

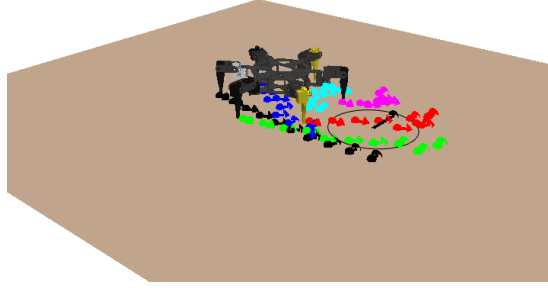


Figura 5.2: *BH3R* un plan de pisadas que requiere rotación de 90°

que el robot cruce un camino con dos bloques inclinados con una meta ubicada 2 metro delante del origen. Esta simulación es configurada, con 4 regiones convexas, es resuelta de forma eficiente en menos de un segundo y resulta en el plan mostrado en la figura 5.3.

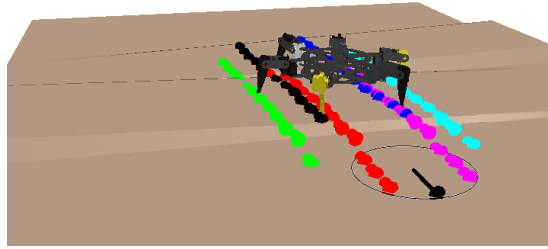


Figura 5.3: *BH3R* siguiendo un plan de pisadas sobre terreno irregular

5.2.2. Cuadrúpedos

Para demostrar la generalidad de nuestro enfoque, el planificador será adaptado a un cuadrúpedo LittleDog de *BostonDynamics* ©, sobre el cual realizaremos pruebas sobre los mismos ambientes probados en hexápodos, con un límite de 36 pisadas por plan. El primer escenario a estudiar, será el de camino de piedras con 13 regiones convexas. En este caso, el problema es resuelto rápidamente en pocos segundos, resultando en el plan presentado en la figura 5.4.

Luego, el planificador es verificado sobre el camino de terreno irregular, con una

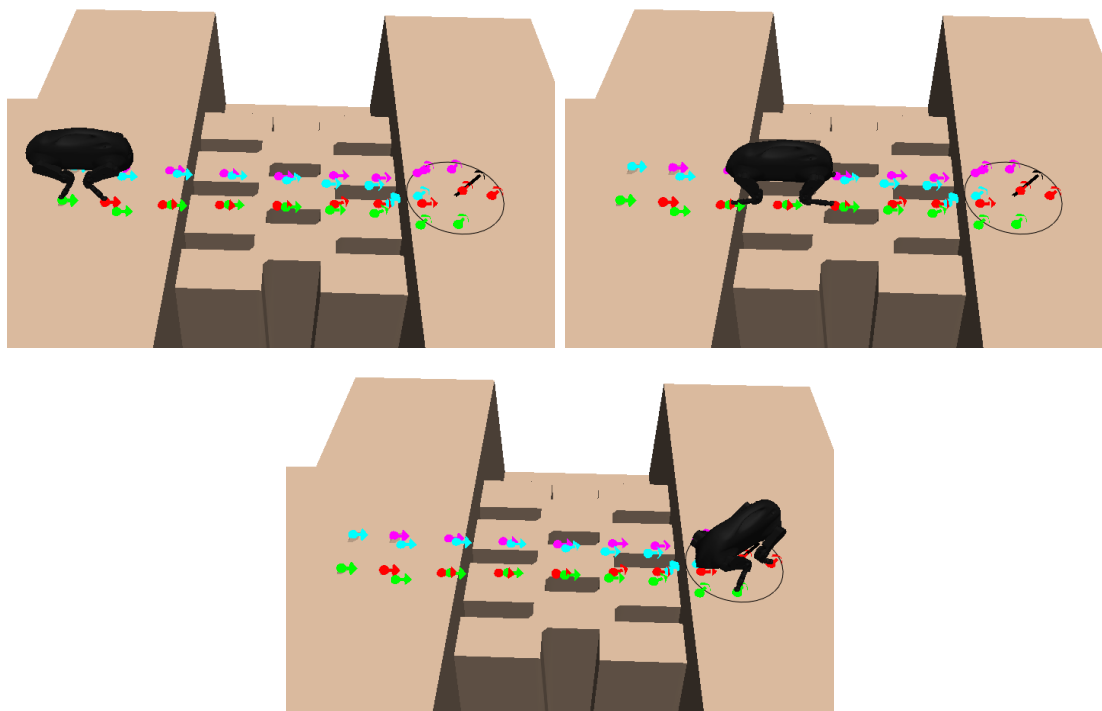


Figura 5.4: *LittleDog* siguiendo el plan de pisadas para un camino de piedras complejo.

meta ubicada 2 metros frente al robot y 4 regiones convexas. Un plan es generado eficientemente en pocos segundos y mostrado en la figura 5.4.

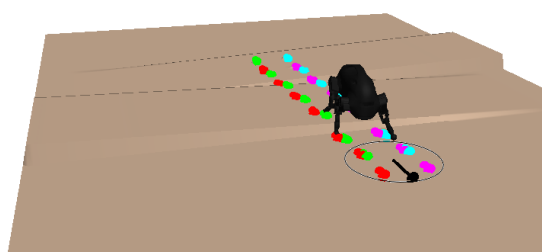


Figura 5.5: *LittleDog* siguiendo un plan de pisadas sobre un terreno irregular

5.2.3. Rendimiento

Como se menciona anteriormente, todas las pruebas son realizadas en un computador de gama baja, con rendimiento inferior al de *procesadores onboard* convencionales. Por esto, estudiaremos la evolución de la complejidad del problema con respecto al número

de pisadas y número de variables enteras-mixtas, a fin de estudiar las principales limitaciones en rendimiento de planificador. Un contraste de distintos rendimientos será realizado en la tabla 5.1, comparando distintos escenarios comunes en el planificador actual, los cuales pueden manejarse en *varios minutos* usando planificadores tradicionales.

Cuadro 5.1: Rendimiento del MIQP sobre distinto escenarios

Pisadas	Variable enteras	Tiempo de solución (s)
36	828	3.68
24	552	0.44
12	312	0.08
8	154	0.02

Es importante notar como el numero de variables enteras-mixtas generan variaciones significativas en la eficiencia del planificador. Por lo tanto, mejoras del rendimiento pueden realizarse al considerar solo regiones convexas cercanas al robot y rangos menores de rotación, reduciendo el numero de variables enteras.

5.3. Planificador de Movimiento y Caminado

Para verificar nuestra implementación de planificación de modo de caminado y movimiento robusto, extenderemos el planificador de pisadas presentado en el capítulo 3 y en [32] para planificar modo de caminado. En esta sección, demostraremos las capacidades de este enfoque a la hora de generar caminado adaptable al tipo de terreno y sus ventajas contra métodos tradicionales basados en Zero-Moment Point (ZMP). Todos las variable $x(t)$, $y(t)$ y $z(t)$ se encuentran en metros. Para los experimentos presentados cada "knot-point" de la optimización representa aproximadamente 100 ms. El margen de estabilidad ϵ no posee unidades pues se mapea a unidades de llave (newton y newton por metro o N/Nm) durante la optimización. La norma L_1 de momento angular centroidal $|k_G|_1$ tiene unidades de Nms ,

5.3.1. Terreno Plano

Para probar un escenario básico, presentamos un comparación entre una trayectoria de caminado rector sobre el eje x con un planificador de ZMP tradicional y con la

formulación propuesta sobre un terreno plano con un coeficiente de fricción de $\mu_s = 1,0$. El planificador de caminado robusto es configurado con un conjunto elipsoidal admisible de dimensiones $1,0cm \times 1,0cm \times 0,5cm$, y el planificador de pisada con 16 pisadas. La secuencia de modo de caminado y el movimiento del cuerpo son mostrados en la figura 5.6.

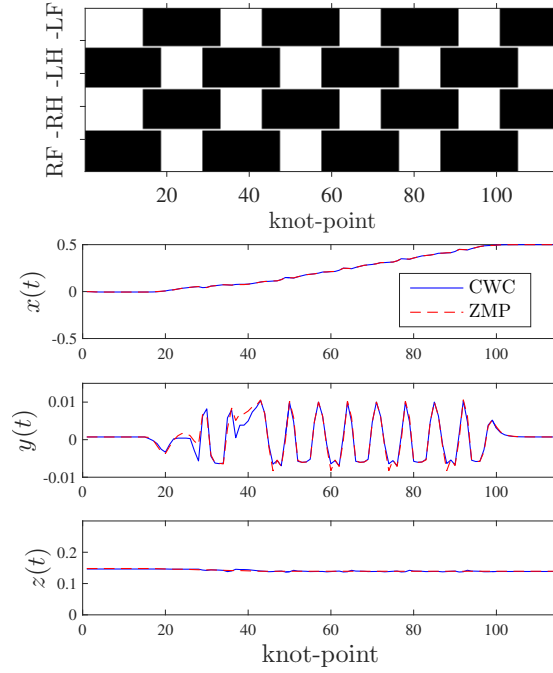


Figura 5.6: Diagrama de fase para un modo de caminado y trayectoria de CoM en terreno plano

Por otro lado, las métricas de robustez para este movimiento son presentadas en la figura 5.7. Como es de esperarse, el movimiento obtenido con nuestro esquema converge a la misma trayectoria que el planificador de ZMP [8]. Esto ocurre ya que el coeficiente de fricción es alto y el modo de caminado se mantiene constante, ya que el terreno no recibe cambios. Nuestra implementación del planificador de pisadas es ejecutada en 0.3 segundo y el movimiento robusto es optimizado en 2 segundos, usando hardware de gama baja.

5.3.2. Terreno irregular

La siguiente prueba es realizada en un ambiente de terreno irregular con un coeficiente de fricción de $\mu_s = 0,5$. El planificador de caminado robusto es configurado con los mismos parámetros y el planificador de pisadas se configura para concatenar cuatro

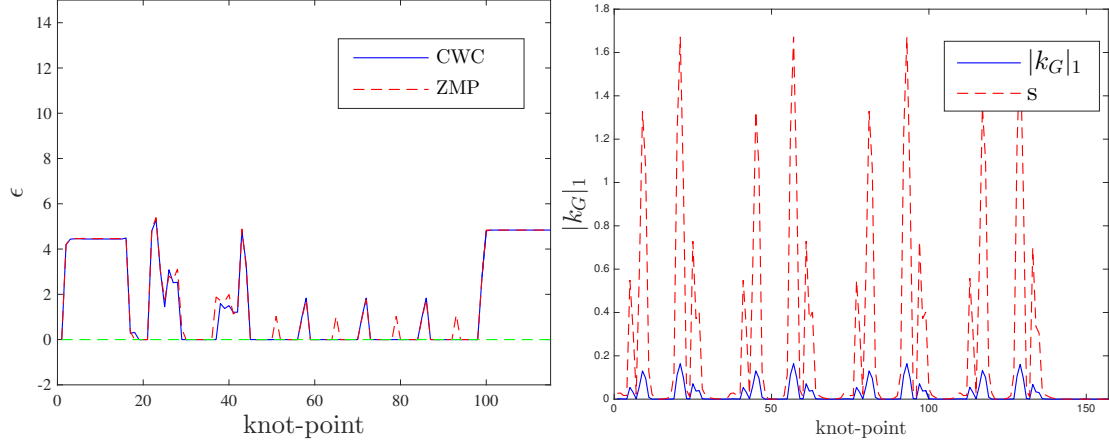


Figura 5.7: Izquierda: Comparación de margen de CWC entre planificación de ZMP y el enfoque propuesto. Derecha: $|k_G|_1$ para caminado en terreno plano

planes sucesivos de 16 pisadas. En este caso, el plan de caminado de robusto y el modo de caminado son mostrados en las figuras 5.9 y 5.8. Es importante resaltar como los modos de caminado de transición emergen al trasladarse entre plataformas irregulares y como son manejados de forma robusta posteriormente, específicamente en los intervalos de tiempo (150, 200), (240, 280) y (330, 375). En este caso, a diferencia de un CPG, el problema de optimización no converge a un patrón (como un trote o un arrastre); sino que generará una transición entre modos de caminado, favoreciendo la estabilidad entre plataformas de caminado.

Adicionalmente, las métricas de estabilidad en el caminado optimizado son mostradas en la figura 5.10. En este caso, el movimiento generado por el planificador Zero-Moment Point tiene segmentos con margen de CWC negativo, especialmente al hacer cambios entre modos de caminado, causando que el robot caiga. Por otro lado, el enfoque propuesto resulta en un plan robusto a través de todo el terreno (con un margen de estabilidad positivo en todo momento), por lo que el robot puede mantenerse estable aun durante las transiciones de modo de caminado.

El planificador de pisadas es resultado en 7.3 segundos y el movimiento robusto es optimizado en menos de 3.7 segundos, usando la misma configuración de hardware de los experimentos anteriores.

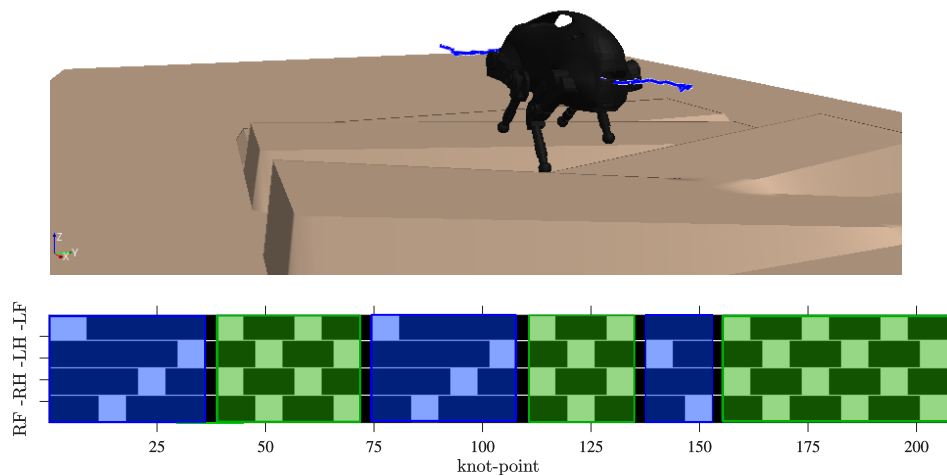


Figura 5.8: Arriba: LittleDog siguiendo un plan de caminado robusto obtenido con nuestro enfoque de planificación en terreno irregular con $\mu_s = 0,5$. Abajo: Modo de caminado generado para un terreno irregular, verde indica trote y azul indica caminado.

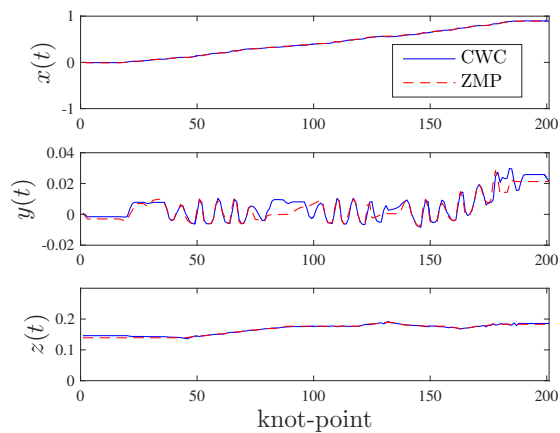


Figura 5.9: Trayectoria de CoM para terreno irregular con $\mu_s = 0,5$ usando ZMP (rojo) y margen de CWC (azul).

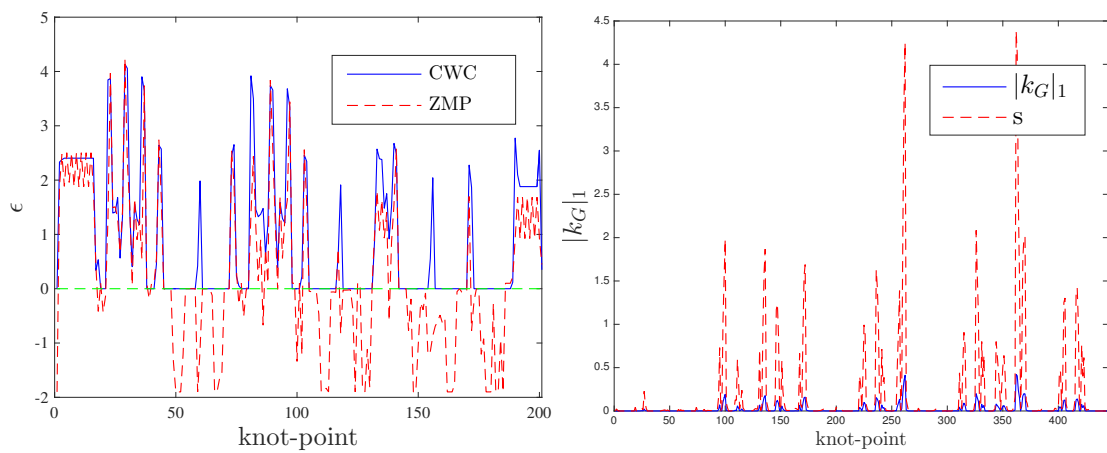


Figura 5.10: Izquierda: Comparación de margen de CWC usando planificación de ZMP y el enfoque propuesto. Derecha: $|k_G|_1$ para terreno irregular con $\mu_s = 0,5$

CONCLUSIONES

En este capítulo final recapitularemos el trabajo realizado para planificación de pisadas, modo de caminado y movimiento robusto en robots multi-patas mediante optimización convexa, indicando las conclusiones obtenidas y los aportes realizados en esta línea de investigación. Finalmente, presentaremos una serie posibles extensiones y trabajos futuros.

6.1. Conclusiones

En este trabajo hemos presentado un esquema de planificación de movimiento en robots caminantes basado en métodos de optimización convexa. A diferencia de métodos actuales, basados en algoritmos de búsqueda y ZMP, nuestro enfoque elimina la necesidad de asumir un terreno plano y fijar un modo de caminado para la ejecución. Por lo tanto, este método permite a robots multi-patas caminar sobre terrenos irregulares, adaptando su trayectoria en función de las condiciones de terreno. En esta trabajo también incorporamos garantías formales de robustez como parte de la planificación, adoptando en margen de cono de llave de contactos como métrica de estabilidad y optimizando movimientos en función de este. Nuestro algoritmo de optimización puede ser manejado de forma eficiente por software comercial "off-the-shelf" y presenta un rendimiento superior a algoritmos de búsqueda anteriores. Nuestro esquema ha sido verificado en simulaciones sobre un cuadrúpedo LittleDog, donde generamos movimientos robusto sobre terrenos irregulares de baja fricción donde un planificador tradicional de ZMP resultaría en una ejecución fallida. A nivel teórico, nuestro método sobrepasa significativamente a enfoques tradicionales basados en algoritmos de búsqueda, planificación de ZMP, o generación central de patrones (CPG).

6.2. Trabajo Futuro

En esta sección presentaremos una serie de recomendaciones y futuras direcciones de trabajo, basadas en las experiencias obtenidas de este trabajo y en áreas de potencial para futuros trabajos de investigación, así como extensiones directas a este trabajo.

6.2.1. Planificación de pisadas

El rendimiento del planificador implementado representa una oportunidad para el desarrollo de planificadores de pisadas *online*. Esta implementación puede aprovechar la estructura de nuestra optimización continua al planificar con secuencias de pisadas cortas y recalculer cambios en el ambiente de forma interactiva, similar al trabajo basado en búsqueda mostrado en [48]. Esto requiere un conjunto de metas locales para evitar mínimos locales, como un muro entre el robot y la meta, posiblemente corriendo una búsqueda discreta sobre el espacio muestreado de las regiones convexas seguras usando RRT* o ARA*.

Adicionalmente, aún hay extensiones que pueden mejorar la calidad del planificador. Un caso de esto es la representación de la cinemática del robot para movimientos dinámicos y agresivos, ya que estos requieren cambios constantes en el espacio de trabajo. Añadir esta consideración requeriría extender la formulación para representar rotaciones en tres dimensiones y la dinámica aproximada del robot, similar al trabajo presentado en [49].

6.2.2. Planificación de movimientos

En el futuro, nos interesa proveer validación experimental de este método de planificación robusta en plataformas robóticas multipatas modernas [6, 7]. Adicionalmente, ya que esta formulación no se encuentra restringida a cuadrúpedos o geometrías en particular, es posible realizar pruebas de este enfoque en robots con distintas geometrías y modos de caminado.

En este proyecto, hemos presentado un planificador movimiento dividido en dos etapas de planificación, planificando pisadas y movimientos por separado. Una extensión

de interés es considerar el método propuesto en [35, 50] y extenderlo con nuestro método, lo que permitiría implementar todo el planificador en un solo problema de optimización convexa de enteros mixtos. Esta extensión permitiría acoplar dos problemas intrínsecamente relacionados y permitiría hacer considerar la evolución dinámica del sistema, generando modos de caminado mas ricos.

En el caso particular de la planificación de modos de caminado, es importante considerar métricas adicionales para representar el tipo de terreno, como la orientación de las regiones convexas en que se ubican los contactos o el área de soporte descrita por las pisadas. Finalmente, este enfoque de planificación no considera los límites en las articulaciones del robot, al solo considerar la dinámica centroidal del sistema. Por lo tanto, es posible que algunos movimientos no sean realizables en el robot sin el uso de motores potentes, como en el caso de los robots humanoides. Añadir esta consideración incrementaría de forma significativa la complejidad de la formulación, ya que requeriría proyectar los límites de torque en el cono de llaves de contacto, Por esta razón, estamos interesados en simplificar nuestra formulación del planificador basado en CWC, lo que permitiría integrar restricciones adicionales sin comprometer la eficiencia del programa.

Bibliografía

- [1] Jonas Buchli, Jerry Pratt, Nicholas Roy, Mrinal Kalakrishnan, Jonas Buchli, Peter Pastor, Michael Mistry, and Stefan Schaal. Learning, planning, and control for quadruped locomotion over challenging terrain. *The International Journal of Robotics Research*, 30(2):236–258, 2011.
- [2] J Zico Kolter, Pieter Abbeel, and Andrew Y Ng. Hierarchical apprenticeship learning with application to quadruped locomotion. In *Advances in Neural Information Processing Systems*, pages 769–776, 2008.
- [3] Matt Zucker, J Andrew Bagnell, Christopher G Atkeson, and James Kuffner. An optimization approach to rough terrain locomotion. In *Robotics and Automation (ICRA), 2010 IEEE International Conference on*, pages 3589–3595. IEEE, 2010.
- [4] Robin Deits and Russ Tedrake. Footstep planning on uneven terrain with mixed-integer convex optimization. In *2014 IEEE-RAS International Conference on Humanoid Robots*, pages 279–286. IEEE, 2014.
- [5] Maurice Fallon, Scott Kuindersma, Sisir Karumanchi, Matthew Antone, Toby Schneider, Hongkai Dai, Claudia Pérez DÁrpino, Robin Deits, Matt DiCicco, Dehann Fourie, et al. An architecture for online affordance-based perception and whole-body planning. *Journal of Field Robotics*, 32(2):229–254, 2015.
- [6] Marco Hutter, Christian Gehring, Dominic Jud, Andreas Lauber, C Dario Bellicoso, Vassilios Tsounis, Jemin Hwangbo, Karen Bodie, Peter Fankhauser, Michael Bloesch, et al. Anymal-a highly mobile and dynamic quadrupedal robot. In *Intelligent Robots and Systems (IROS), 2016 IEEE/RSJ International Conference on*, pages 38–44. IEEE, 2016.
- [7] Carlos Mastalli, Ioannis Havoutis, Alexander W Winkler, Darwin G Caldwell, and Claudio Semini. On-line and on-board planning and perception for quadrupedal locomotion. In *2015 IEEE International Conference on Technologies for Practical Robot Applications (TePRA)*. IEEE, 2015.
- [8] Katie Byl. *Metastable legged-robot locomotion*. PhD thesis, Massachusetts Institute of Technology, 2008.

- [9] Claudio Semini, Nikos G Tsagarakis, Emanuele Guglielmino, Michele Focchi, Ferdinando Cannella, and Darwin G Caldwell. Design of hyq—a hydraulically and electrically actuated quadruped robot. *Proceedings of the Institution of Mechanical Engineers, Part I: Journal of Systems and Control Engineering*, 225(6):831–849, 2011.
- [10] Pierre-Brice Wieber, Russ Tedrake, and Scott Kuindersma. Modeling and control of legged robots. In *Springer Handbook of Robotics*, pages 1203–1234. Springer International Publishing, 2016.
- [11] Carlos Mastalli. *Planning and Execution of Dynamic Whole-Body Locomotion on Challenging Terrain*. PhD thesis, Istituto Italiano di Tecnologia (IIT), 2017.
- [12] Robin Lloyd Henderson Deits. *Convex segmentation and mixed-integer footstep planning for a walking robot*. PhD thesis, Massachusetts Institute of Technology, 2014.
- [13] Milton Hildebrand. The quadrupedal gaits of vertebrates. 1989.
- [14] D Wettergreen and C Thorpe. Gait generation for legged robots. In *Intelligent Robots and Systems, 1992., Proceedings of the 1992 IEEE/RSJ International Conference on*. IEEE, 1992.
- [15] Pierre-Brice Wieber. On the stability of walking systems. In *Proceedings of the international workshop on humanoid and human friendly robotics*, 2002.
- [16] Sébastien Barthélemy and Philippe Bidaud. Stability measure of postural dynamic equilibrium based on residual radius. In *Advances in Robot Kinematics: Analysis and Design*, pages 399–407. Springer, 2008.
- [17] Hirohisa Hirukawa, Shizuko Hattori, Kensuke Harada, Shuuji Kajita, Kenji Kaneko, Fumio Kanehiro, Kiyoshi Fujiwara, and Mitsuharu Morisawa. A universal stability criterion of the foot contact of legged robots—adios zmp. In *Proceedings 2006 IEEE International Conference on Robotics and Automation, 2006. ICRA 2006*. IEEE, 2006.
- [18] Hongkai Dai and Russ Tedrake. Planning robust walking motion on uneven terrain via convex optimization. In *Humanoid Robots (Humanoids), 2016 IEEE-RAS 16th International Conference on*, pages 579–586. IEEE, 2016.
- [19] N. Perrin, C. Ott, J. Engelsberger, O. Stasse, F. Lamiriaux, and D. G. Caldwell. Continuous legged locomotion planning. *IEEE Transactions on Robotics*, PP(99):1–6, 2016.
- [20] Maureen Rojas, Novel Certad, José Cappelletto, and Juan Grieco. Foothold planning and gait generation for a hexapod robot traversing terrains with forbidden zones. In *12th Latin American Robotics Symposium LARS2015 and 2015 Third Brazilian Symposium on Robotics*, 2015.
- [21] Alexander W Winkler, Carlos Mastalli, Ioannis Havoutis, Michele Focchi, Darwin G Caldwell, and Claudio Semini. Planning and execution of dynamic whole-body locomotion for a hydraulic quadruped on challenging terrain. In *2015 IEEE International Conference on Robotics and Automation (ICRA)*, pages 5148–5154. IEEE, 2015.

- [22] Brian W Satzinger, Chelsea Lau, Marten Byl, and Katie Byl. Tractable locomotion planning for robosimian. *The International Journal of Robotics Research*, page 0278364915584947, 2015.
- [23] S. Boyd and L. Vandenberghe. *Convex Optimization*. Cambridge University Press, 2004.
- [24] Arthur Richards and Jonathan How. Mixed-integer programming for control. In *Proceedings of the 2005, American Control Conference, 2005.*, pages 2676–2683. IEEE, 2005.
- [25] Dimitris Bertsimas, David B Brown, and Constantine Caramanis. Theory and applications of robust optimization. *SIAM review*, 53(3):464–501, 2011.
- [26] M. Vukobratovic and D. Juricic. Contribution to the synthesis of biped gait. *IEEE Transactions on Biomedical Engineering*, BME-16(1):1–6, Jan 1969.
- [27] George L Nemhauser and Laurence A Wolsey. Integer programming and combinatorial optimization. *Wiley, Chichester. GL Nemhauser, MWP Savelsbergh, GS Sigismondi (1992). Constraint Classification for Mixed Integer Programming Formulations. COAL Bulletin*, 20:8–12, 1988.
- [28] Alexander Herzog, Nicholas Rotella, Stefan Schaal, and Ludovic Righetti. Trajectory generation for multi-contact momentum control. In *Humanoid Robots (Humanoids), 2015 IEEE-RAS 15th International Conference on*, pages 874–880. IEEE, 2015.
- [29] David E Orin, Ambarish Goswami, and Sung-Hee Lee. Centroidal dynamics of a humanoid robot. *Autonomous Robots*, 35(2-3):161–176, 2013.
- [30] Russ Tedrake. Underactuated robotics: Learning, planning, and control for efficient and agile machines: Course notes for mit 6.832. *Working draft edition*, page 3, 2009.
- [31] Stéphane Caron, Quang-Cuong Pham, and Yoshihiko Nakamura. Leveraging cone double description for multi-contact stability of humanoids with applications to statics and dynamics. In *Robotics: Science and System*, July 2015.
- [32] Bernardo Aceituno-Cabezas, Jose Cappelletto, Juan C Grieco, and Gerardo Fernandez-Lopez. A generalized mixed-integer convex program for multilegged footstep planning on uneven terrain. *arXiv preprint arXiv:1612.02109*, 2017.
- [33] Scott Kuindersma, Robin Deits, Maurice F Fallon, Andrés Valenzuela, Hongkai Dai, Frank Permenter, Twan Koolen, Pat Marion, and Russ Tedrake. Optimization-based locomotion planning, estimation, and control design for the atlas humanoid robot. *Auton. Robots*, 40(3):429–455, 2016.
- [34] Hongkai Dai, Andrés Valenzuela, and Rus Tedrake. Whole-body motion planning with centroidal dynamics and full kinematics. In *IEEE-RAS International Conference on Humanoid Robots*, 2014.

- [35] Brahayam Ponton, Alexander Herzog, Stefan Schaal, and Ludovic Righetti. A convex model of humanoid momentum dynamics for multi-contact motion generation. In *Humanoid Robots (Humanoids), 2016 IEEE-RAS 16th International Conference on*, pages 842–849. IEEE, 2016.
- [36] Michael Posa, Cecilia Cantu, and Russ Tedrake. A direct method for trajectory optimization of rigid bodies through contact. *International Journal of Robotics Research*, 33(1):69–81, 2014.
- [37] Andrei Herdt, Holger Diedam, Pierre-Brice Wieber, Dimitar Dimitrov, Katja Mombaur, and Moritz Diehl. Online walking motion generation with automatic footstep placement. *Advanced Robotics*, 24(5-6):719–737, 2010.
- [38] Elena Garcia, Joaquin Estremera, and Pablo Gonzalez De Santos. A comparative study of stability margins for walking machines. *Robotica*, 20(6):595–606, 2002.
- [39] Arthur Richards, John Bellingham, Michael Tillerson, and Jonathan How. Coordination and control of multiple uavs. In *AIAA guidance, navigation, and control conference, Monterey, CA*, 2002.
- [40] Robin Deits and Russ Tedrake. Computing large convex regions of obstacle-free space through semidefinite programming. In *Algorithmic Foundations of Robotics XI*, pages 109–124. Springer, 2015.
- [41] Richard M Karp. Reducibility among combinatorial problems. In *Complexity of computer computations*, pages 85–103. Springer, 1972.
- [42] Russ Tedrake. Drake: A planning, control, and analysis toolbox for nonlinear dynamical systems, 2014.
- [43] Bernardo Aceituno-Cabezas, Hongkai Dai, Jose Cappelletto, Juan C Grieco, and Gerardo Fernandez-Lopez. A mixed-integer convex optimization framework for robust multilegged robot locomotion planning over challenging terrain. In *IROS*. IEEE, 2017.
- [44] V. Barasuol, J. Buchli, C. Semini, M. Frigerio, E. R. De Pieri, and D. G. Caldwell. A reactive controller framework for quadrupedal locomotion on challenging terrain. In *2013 IEEE International Conference on Robotics and Automation*, May 2013.
- [45] M. Oliveira, C. P. Santos, L. Costa, V. Matos, and M. Ferreira. Multi-objective parameter cpg optimization for gait generation of a quadruped robot considering behavioral diversity. In *2011 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2011.
- [46] Inc. Gurobi Optimization. Gurobi optimizer reference manual, 2015.
- [47] MOSEK ApS. *The MOSEK optimization toolbox for MATLAB manual. Version 7.1 (Revision 28)*., 2015.

- [48] Philipp Karkowski and Maren Bennewitz. Real-time footstep planning using a geometric approach. In *2016 IEEE International Conference on Robotics and Automation (ICRA)*, pages 1782–1787. IEEE, 2016.
- [49] Andrés Klee Valenzuela. *Mixed-integer convex optimization for planning aggressive motions of legged robots over rough terrain*. PhD thesis, Massachusetts Institute of Technology, 2016.
- [50] Bernardo Aceituno-Cabezas, Carlos Mastalli, Hongkai Dai, Michele Focchi, Andreea Radulescu, Darwin G. Caldwell, Jose Cappelletto, Juan C Grieco, Gerardo Fernandez-Lopez, and Caludio Semini. Simultaneous contact and motion planning for robust multi-legged locomotion via mixed-integer convex optimization. 2018.

Apéndice A

En este apéndice se presenta el resumen de tres artículos de investigación escritos durante el presente trabajo de tesis.

- Aceituno-Cabezas, Bernardo; Dai, Hongkai; Cappelletto, José; Grieco, Juan C. and Fernández-López, Gerardo. **A Mixed-Integer Convex Optimization Framework for Robust Multi-Legged Locomotion Planning over Challenging Terrain**. 2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2017). Vancouver, 2017.

Este trabajo se presentará de forma oral en la conferencia IEEE/RSJ International Conference on Intelligent Robots and Systems 2017, cuyos *proceedings* serán publicados en el índice *IEEE Xplore*. Esta conferencia es uno de los principales eventos de la Sociedad IEEE de Robótica y Automatización (*IEEE Robotics and Automation Society*) y goza del mayor prestigio en esta área de investigación.

El motivo de este trabajo es la presentación del esquema de optimización convexa entera para locomoción multi-patas propuesto en este trabajo de tesis.

- Aceituno-Cabezas, Bernardo; Mastalli, Carlos; Dai, Hongkai; Focchi, M.; Radulescu, Andreea; Caldwell, D. G.; Cappelletto, José; Grieco, Juan; Fernández, Gerardo; Semini, Claudio. **Simultaneous Contact and Motion Planning for Robust Multi-Legged Locomotion via Mixed-Integer Convex Optimization**.

Este artículo fue enviado a la revista IEEE Robotics and Automation Letters para su arbitraje.

El tema de este artículo versa sobre el desarrollo de un algoritmo de planificación simultánea de contactos y movimiento basado en optimización convexa entera, usando los principios teóricos desarrollados en esta tesis. Este algoritmo es validado en robot cuadrúpedo hidráulico (HyQ) con experimentos sobre distintos terrenos irregulares.

- Aceituno-Cabezas, Bernardo; Cappelletto, José; Grieco, Juan C. and Fernández-López, Gerardo. **A Generalized Mixed-Integer Convex Program for Multi-legged Footstep Planning on Uneven Terrain**. arXiv:1612.02109v2.

Este reporte técnico se encuentra publicado de forma abierta en el repositorio arXiv, manejado por la biblioteca de Cornell.

Este reporte presenta los detalles de desarrollo del algoritmo de planificación de pisadas basado en programación de enteros mixtos presentado en este trabajo.

En las siguientes páginas se presentan los resúmenes y primeras páginas de los trabajos mencionados.

Adicionalmente, se presentan dos artículos cortos de investigación escritos durante el presente trabajo de tesis.

- Aceituno-Cabezas, Bernardo; Dai, Hongkai; and Fernández-López, Gerardo. **Mixed-Integer Convex Optimization of Non-Gaited Multi-Legged Walking Sequences**. Robotics Science and Systems (RSS) 2017 Workshop on Challenges in Dynamic Legged Locomotion.

Este trabajo se presentó de forma oral en el taller Challenges on Dynamic Legged Locomotion de la conferencia Robotics Science and Systems (RSS) 2017, Cambridge, Massachusetts.

El motivo de este trabajo es la presentación de la formulación de programación entera para planificación de modo de caminado multi-patas propuesto en este trabajo de tesis.

- Aceituno-Cabezas, Bernardo; Dai, Hongkai; Mastalli, Carlos; Focchi, M.; Radulescu, Andreea; Caldwell, D. G.; Cappelletto, José; Grieco, Juan; Fernández, Gerardo; Semini, Claudio. **A Mixed-Integer Convex Formulation for Simultaneous Contact, Gait and Motion Optimization on Multi-Legged Robots**. IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS) 2017 Workshop on Frontiers in Contact-rich Robotic Interaction: Modeling, Optimization and Control Synthesis.

Este trabajo se presentará de forma oral en el taller Frontiers in Contact-rich Robotic Interaction: Modeling, Optimization and Control Synthesis de la conferencia IEEE/RSJ International Conference on Intelligent Robots and Systems 2017, Vancouver, CA.

Este artículo describe un algoritmo de planificación simultánea de contactos y movimiento basado en optimización convexa entera, usando los principios teóricos desarrollados en esta tesis.

En las siguientes páginas se presentan los trabajos cortos mencionados.

A Mixed-Integer Convex Optimization Framework for Robust Multilegged Robot Locomotion Planning over Challenging Terrain

Bernardo Aceituno-Cabezas¹, Student Member, IEEE, Hongkai Dai², José Cappelletto¹,
Juan C. Grieco¹, and Gerardo Fernández-López¹, Member, IEEE

Abstract—This paper introduces an optimization-based framework for robust multilegged walking motion planning. Previous approaches use fixed gait sequences, and rely on Zero Moment Point (ZMP) to guarantee dynamic stability. While this combination works well on flat ground, it does not generalize to uneven terrain requiring aggressive gait or gait transition. To overcome such difficulties, in this paper, we present an optimization framework, that can plan both the contact location and gait sequence simultaneously in a mixed-integer convex optimization program. Moreover, we rely on the Contact Wrench Cone (CWC) stability criterion, which generalizes the ZMP criterion to uneven terrain with friction cone constraints, and we plan the walking motion together with the angular momentum through a convex optimization program. Our approach is successfully tested on a LittleDog quadruped over simulated scenarios. We show that on the flat ground, our planner generates a periodic gait, same as Central Pattern Generator + ZMP planner; while on uneven terrain, our planner can successfully generate a motion containing different gaits, with a center-of-mass motion that respects the friction cone constraints, which are violated by ZMP planners. This improvement clearly demonstrates the advantage of our approach over traditional planning strategies.

Index Terms—Motion and Path Planning, Multilegged Robots, Robust Optimization.

I. INTRODUCTION

Multilegged robots are powerful platforms when traversing rough terrains [1], [2], [3]. There are two major challenges for this motion planning task: 1) The number of possible gait combinations grows exponentially with the number of footsteps; and 2) It is non-trivial to guarantee that the planned motion is dynamically feasible, so that the robot will not fall over. Previously, several researchers have used Central Pattern Generators (CPG) as an improved model to generate gait sequences [2], [4], [5], [6], and relied on Zero Moment Point (ZMP) for dynamic stability [7]. While CPG and ZMP work well locomotion over flat ground, neither of them extend naturally to uneven terrain. On challenging terrain, the robot needs to adjust its gait frequently to adapt to the terrain change [8], and CPG does not easily account for the transition gait [9]. Moreover, while well defined on flat ground, the Zero Moment Point is not applicable to uneven ground with friction cone constraints. To this end, instead of using CPG, we will plan the gait sequence simultaneously to the contact plan by using a Mixed-Integer Convex Optimization, and guarantee the dynamical stability



Fig. 1. LittleDog walking over rough terrain with non-coplanar contacts and linear friction cones (blue)

of the motion by explicitly optimizing the Contact Wrench Cone (CWC) stability margin [10], [11], rather than ZMP, by extending the work of [12] for multilegged robots.

There has been a series of work on planning footstep locations for bipedal robot, using Mixed-Integer Convex Programming (MICP) [13], [14]. This approach allows the planner to efficiently handle cluttered environments, rotations, and uneven terrain. In [13], the authors first find a set of collision free contact regions, and then use integer variables to assign each footstep to one region. We extend this approach to multileg robot. Compared to bipedal robot which has a unique footstep sequence, where it only alternates between left and right leg; the multileg robot has extra complexity that the footstep sequence has exponentially number of combinations. Furthermore, such formulations do not consider heuristics in the optimization, leading to a purely geometric planning problem. In this paper we will use MICP to efficiently find the optimal contact and gait sequences, along with the incorporation of terrain heuristics. This allows the planner to reason simultaneously about *where* and *when* to make contact.

The main contribution of this work is a robust walking motion planning framework for multilegged robots, which can negotiate different gaits and contact paths, depending on the conditions of the terrain, and provide *formal robustness guarantees*, based on the CWC stability margin. We decouple the planning problem in two numerical optimization problems, leveraging MICP and Convex Optimization. Both stages are solved efficiently to their global optimum, using off-the-shelf optimization solvers.

The remainder of this paper is organized as follows: Section II introduces the main concepts referred throughout the paper, Section III presents the formulation of our proposed

The authors are with (1) the Mechatronics Research Group, Simón Bolívar University {12-10764, cappelletto, jcgrico, gfernandez}@usb.ve, and (2) is with the Toyota Research Institute hongkai.dai@tri.global

Simultaneous Contact and Motion Planning for Robust Multi-Legged Locomotion via Mixed-Integer Convex Optimization

Bernardo Aceituno-Cabezas¹, Carlos Mastalli², Hongkai Dai³, Michele Focchi², Andreea Radulescu², Darwin G. Caldwell², José Cappelletto¹, Juan C. Grieco¹, Gerardo Fernández-López¹, and Claudio Semini²

Abstract—Traditional multi-legged motion planning approaches divide the problem in several stages such as contact search and trajectory generation. However, reasoning about contacts and motions simultaneously is crucial to the generation of complex whole-body motions. Currently, coupling these problems has required either to assume fixed gait and flat terrain condition, or non-convex optimization with intractable computation time. In this paper, we propose a mixed-integer convex formulation to plan simultaneously contact locations, gait transitions and motion, in a computationally efficient fashion. In contrast to previous works, our approach is not limited to flat terrain nor to a pre-specified gait sequence. Instead, we incorporate the friction cone stability margin, approximate the robot's torque limits, and plan the gait using mixed-integer convex constraints. We validate our approach on the HyQ robot by traversing different challenging terrains, where non-convexity and flat terrain assumptions might lead to sub-optimal or unstable plans. Our method increases the motion generality while keeping a low computation time.

I. INTRODUCTION

Planning motions for multi-legged robots can be a challenging task, as it involves both the discrete choice on gait sequence, and continuous decisions on foot location and robot dynamics. Because of this, most traditional approaches [1], [2], [3], [4] simplify the planning problem by decoupling it in two stages: 1) find a sequence of contact locations; and 2) generate body trajectories that respect a stability metric, like the Zero Moment Point (ZMP) [5] or the generalized Contact Wrench Cone (CWC) [6], [7] stability criterion. Ignoring the dynamics in the first one might restrict the feasible motion on the second one, when considering dynamics. Hence, in this paper we combine these two stages into one single problem, in order to efficiently generate complex behaviors.

Trajectory Optimization (TO) has emerged as a solution to the coupled motion planning problem [8], [9], [10], [11], [12]. Unfortunately, even state of the art approaches are often restricted to simple environments, a fixed gait, or require intractable computation time. Furthermore, since these methods are posed as non-convex optimizations, none of them can guarantee global optimality, and can easily fall into local minima. On the other hand, Mixed-Integer Convex Programming (MICP) proved to be an efficient tool to solve multi-contact motion planning problems. In [1], [3], the authors first find a set of safe surfaces, and

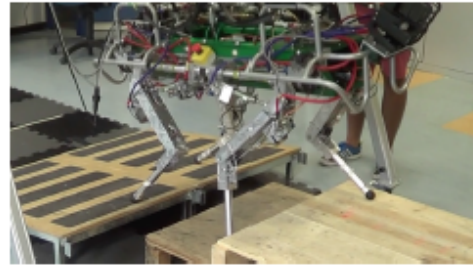


Fig. 1: The proposed approach generates robust plans in challenging terrain, simultaneously reasoning about contacts and motions. This figure shows HyQ executing an optimized motion with non-coplanar contacts.

then use integer variables to assign each contact to one surface. Further works incorporate a convex or mixed-integer convex dynamic model in the formulation [13], [14], which can efficiently optimize motions and contacts. Unfortunately, these approaches do not incorporate torque constraints nor maximize any generalized stability margin. Moreover, they do not reason about the sequence of contacts, being limited to a fixed gait sequence.

In this work, we introduce a novel MICP formulation for non-gaited multi-legged locomotion on rough terrain. Here, we formulate a single mixed-integer convex optimization problem that simultaneously plans contacts and motions. In contrast to other approaches, our method guarantees dynamic feasibility in the execution and is able to handle complex environments. For this, we incorporate convex representations of friction cone, torque limits, and gait planning constraints. For the friction cone constraints, we exploit the segmentation of the terrain and pose the stability margin computation as a convex constraint. This optimization problem can then be solved efficiently to its *global optimum*, at least two orders of magnitude faster than similar non-convex approaches, using off-the-shelf optimization solvers. We validate our approach by traversing different challenging terrains with the Hydraulically actuated Quadruped (HyQ) robot, shown in Fig. 1. Our approach generates robust motions, even during gait transitions and non-coplanar contact scenarios.

The rest of this paper is organized as follows: Section II presents our simultaneous contact and motion planning approach. Section III presents our whole-body controller. Section IV presents a set of experiments on HyQ traversing challenging terrains, and Section V discusses and concludes on the contributions of this work.

The authors are with (1) the Mechatronics Research Group, Simón Bolívar University {12-10764, cappelletto, jcgrieco, gfernandez}@usb.ve, (2) are with the Department of Advanced Robotics, Istituto Italiano di Tecnologia {firstname.surname}@iit.com, and (3) is with the Toyota Research Institute hongkai.dai@tri.global

A Generalized Mixed-Integer Convex Program for Multilegged Footstep Planning on Uneven Terrain

Bernardo Aceituno-Cabezas, *Student Member, IEEE*, José Cappelletto, Juan C. Grieco, and Gerardo Fernández-López, *Member, IEEE*

Abstract—Robot footstep planning strategies can be divided in two main approaches: discrete searches and continuous optimizations. While discrete searches have been broadly applied, continuous optimizations approaches have been restricted for humanoid platforms. This article introduces a generalized continuous-optimization strategy for multilegged footstep planning which can be adapted to different platforms, regardless the number and geometry of legs. This approach leverages *Mixed-Integer Convex Programming* to account for the non-convex constraints that represent footstep rotation and obstacle avoidance. The planning problem is formulated as an optimization problem, which considers robot geometry and reachability with linear constraints and can be efficiently solved using optimization software. To demonstrate the functionality and adaptability of the planner, a set of tests are performed on a BH3R hexapod and a LittleDog quadruped on scenarios which can't be easily handled with discrete searches, such tests are solved efficiently in fractions of a second. This work represents, to the knowledge of the authors, the first successful implementation of a continuous optimization-based multilegged footstep planner.

Index Terms—Legged Robots, Motion and Path Planning, Optimization and Optimal Control.

I. INTRODUCTION

Motion planning for walking robots has become an active area of research on the recent years. One particular and well-established method is based in the separation of the contact sequences and body motion plans as different problems. This consists in specifying a sequence of contacts in the form of footsteps for the robot to follow, and then solve for body motions separately. Such method requires powerful planning tools to obtain optimal and feasible footstep plans for the platform. As noted by Deits and Tedrake [1], footstep planning approaches can be broadly divided in two areas: *discrete searches and continuous optimizations*. While discrete searches have been thoroughly studied for multilegged robots, the entirety of continuous optimization approaches have been restricted to bipedal platforms. The main issue when adapting such approach to multilegged platforms, is that representing the geometry of a multilegged robot and its workspace in a continuous optimization footstep planner results in the introduction of non-convex constraints. This results in a nonlinear program which is difficult to solve and can't guarantee an optimal result.

This research was partially funded by the Universidad Simón Bolívar Research and Development Deanship

B. Aceituno-Cabezas, J. Cappelletto, J. C. Grieco and G. Fernández-López are with the Mechatronics Research Group, Simón Bolívar University, 1080 Sartenejas, Venezuela {12-10764, cappelletto, jcgrieco, gfernandez}@usb.ve

In the past, several graph-search algorithms such as A* or RRT have been successfully applied to multilegged footstep planning: Zucker et al. [2] use ARA* with learned cost maps to obtain plans for a LittleDog robot walking on challenging terrain, Satzinger et al. [3] introduced a planner based on a modified version of A* to obtain footstep plans for a quadruped platform given a pre-computed foothold set to follow during the DARPA Robotics Challenge (DRC) finals, Winkler et al. [4] build cost maps based on the complexity of terrain, and then apply ARA* to obtain footsteps for a Quadruped. However, these methods require *pre-computed foothold sets* and terrain cost maps in order to provide a good heuristic for the search. These are difficult to compute without a prior knowledge of the desired path and are limited to a *discrete* version of the environment.

On the other hand, continuous optimization approaches have been restricted for humanoid platforms. Herdt et al. [5] introduced a Quadratic Programming (QP) formulation that can produce optimal footstep placements and controls for a walking robot model when footstep orientations are fixed and obstacles in the environment are ignored. Fallon et al. [6] implemented a nonlinear programming algorithm that can plan footsteps while considering orientation, but couldn't guarantee optimality, nor generate plans around obstacles. Deits and Tedrake [1] generalized the previous approach by formulating the problem as a *Mixed-Integer Quadratically Constrained Quadratic Program* (MIQCQP) for the DRC Finals, constraining every step to lie within a safe convex region, and using a *piecewise linear approximation* of the *sin* and *cos* functions to plan orientations. Such approach follows a similar idea as the work of Richards et al. [7], where safe UAV trajectories are obtained by using binary variables to restrict the plan within a set of *safe convex regions* represented as the intersection of the half-planes of the obstacles faces.

This paper introduces a generalized continuous optimization footstep planner for multilegged platforms, which can be formulated as a *Mixed-Integer Quadratic Program* (MIQP), solved efficiently using commercial convex optimization software, and can be easily adapted to multilegged robots with different geometries. The remaining of this paper is organized as follows: Section II describes the formulation of the program and its constraints, Section III presents the results obtained from a implementation of the planner in two different multilegged platforms (a LittleDog quadruped and a BH3R circular hexapod) and Section IV discusses and concludes on the contributions of this work.

Mixed-Integer Convex Optimization of Non-Gaited Multi-Legged Walking Sequences

Bernardo Aceituno-C.*, Hongkai Dai† and Gerardo Fernández-López*

*Universidad Simón Bolívar, Caracas, Venezuela

Email: {12-10764, gfernandez}@usb.ve

†Toyota Research Institute, Los Altos, CA, USA

Email: hongkai.dai@global.ti

I. MOTIVATION

There has been a series of work on planning optimal footstep plans for bipedal robots, using Mixed-Integer Convex Programming (MICP). In Deits' work [1], the authors planned the footsteps for bipedal robots. We extend this approach to multi-leg robots. Unlike the bipedal robot case whose leg sequence is merely alternating the left and right leg, the multi-leg robot has extra complexity that the leg sequence has exponentially number of combinations.

Previously researchers have used Central Pattern Generators (CPG) to plan the gait sequence. While CPG works well on flat terrain, it doesn't extend naturally to uneven terrain. Therefore, most contact and walking motion planning approaches can't simultaneously reason about *where* and *when* to place the foot. Here, we will propose a MICP formulation to efficiently search for the foot locations and gait sequences simultaneously.

II. PROPOSED APPROACH

In order to optimize gait sequences that adapt to the terrain, we propose a MICP formulation that can work as a part of optimization-based footstep planners, such as proposed by Deits [1] or Aceituno [2]. Thus, this approach will optimize over the following decision variables:

- 1) Contact locations: an array of N_f vectors of the form $f = (x, y, z, \theta)$ representing the position of each contact the yaw orientation of the trunk when transitioning to the contact. This array is ordered by leg number, such that for n_l legs f_i represents a single leg position and f_{i+n_l} represents its next contact position.
- 2) Safe-region assignment: done with a binary matrix H , such that $H_{r,j} \Rightarrow A_r f_j \leq b_r$, assigns contact location j to a convex region r , where $\sum_{r=1}^{N_r} H_{r,j} = 1$.
- 3) Gait sequence: a binary matrix that assigns each transition of the leg from a contact to the next to a certain timestep. This also defines the workspace of the robot at each contact phase, ensuring dynamic reachability.

Gait sequences are often represented by phase diagrams, which indicate the n_l legged supports for a given support phase. Inspired by this representation, we will describe the support phase assigned to each leg transfer cycle within the contact plan. To do this, we introduce a binary matrix $T \in \{0, 1\}^{N_f \times N_f}$, where $T_{ij} = 1$ means that the i_{th} foot will

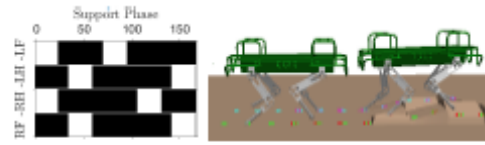


Fig. 1. Left: Phase diagram of an optimized gait sequence on rough terrain. Right: Execution of the optimized motion, HyQ trots when the terrain is flat and starts crawling when the terrain is rough enough.

transfer from foot location $i - n_l$ to i at the j_{th} support phase. Since each foot location can only be reached once in the plan, we constrain this as:

$$\sum_{j=1}^{N_f} T(i, j) = 1, \quad \forall i = 1, \dots, N_f \quad (1)$$

This representation is equivalent to indicating the supports at each phase, since we assume that the contact vector is ordered by leg number. Furthermore, we enforce that each cycle of n_l contacts must be reached before the next transfer cycle starts.

Finally, we aim to minimize a convex objective function, which includes the following costs:

- Aggressive movements with roughness between adjacent contact locations.
- Distance between final foot locations and the goal.
- Relative displacement between contacts.
- Plan execution time.

This cost function can ensure that the planner assigns more support to a transfer phase over rough terrain, while seeking the fastest gait when the terrain is relatively flat.

REFERENCES

- [1] R. Deits and R. Tedrake, "Footstep planning on uneven terrain with mixed-integer convex optimization," in *Humanoid Robots (Humanoids), 2014 14th IEEE-RAS International Conference on*. IEEE, 2014, pp. 279–286.
- [2] B. Aceituno-Cabezas, J. Cappelletto, J. C. Grieco, and G. Fernandez-Lopez, "A Generalized Mixed-Integer Convex Program for Multilegged Footstep Planning on Uneven Terrain," *arXiv preprint arXiv:1612.02109*, 2017.

A Mixed-Integer Convex Formulation for Simultaneous Contact, Gait and Motion Optimization on Multi-Legged Robots

Bernardo Aceituno-Cabezas¹, Hongkai Dai², Carlos Mastalli³, Michele Focchi³, Andreea Radulescu³, Darwin G. Caldwell³, José Cappelletto¹, Juan C. Grieco¹, Gerardo Fernández-López¹, and Claudio Semini³

I. MOTIVATION

Planning motions for multi-legged robots is a complex task, as it involves both the discrete choice on gait sequence, and continuous decisions on foot location and robot dynamics. For this reason, most approaches have decoupled the problem in two stages: searching contacts and planning motions. However, by ignoring the dynamics in the first stage, it might restrict the feasible motions on the second stage when considering the robot dynamics. Hence in this work we combine the two stages into one single problem using mixed-integer optimization.

Previous research has addressed this problem with non-convex optimization, which requires significant computation time and cannot guarantee optimality. Others have used Mixed-Integer Convex Programs (MICP) [1], [2], [3], using a fixed gait. Here, we propose a MICP formulation to simultaneously plan the gaits and dynamically feasible motions on rough terrain, by efficiently solving a MICP.

II. PROPOSED FORMULATION

In order to plan the robot's motion, we assume that it is divided in N_t time-slots over which we generate leg swings. Here, we rely on a centroidal dynamics model. Hence, our decision variables are the center of mass \mathbf{r} , end-effectors \mathbf{p}_e , and contact forces λ_e trajectories.

1) *Contact Locations*: we optimize over N_c contacts, which are represented by their position and the trunk orientation at their swing time.

2) *Gait Sequence*: we optimize the gait sequence by assigning the transition to each contact, given a determined time-slot N_t , in a gait transition matrix $\mathbf{T} \in \{0, 1\}^{N_c \times N_t}$.

3) *Safe-regions assignment*: As in previous work [2], we avoid obstacles by assigning contacts to one of N_r convex safe regions, done with a binary matrix $\mathbf{H} \in \{0, 1\}^{N_c \times N_r}$.

4) *Kinematic reachability*: we approximate the kinematic reachability of each end-effector by adding bounding box constraints on the contact locations. For this, we use the CoM location as a reference for the workspace of each leg.

5) *Contact Force Constraints*: we ensure stability by constraining each contact force to lie in its corresponding

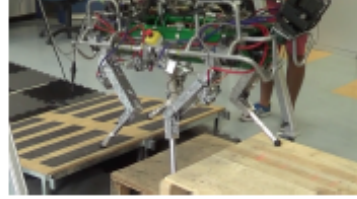


Fig. 1. Our approach generates robust plans on challenging terrains, simultaneously reasoning about contact locations and body motions. We validate on the HyQ robot by transversing different challenging terrains in which considering non-coplanar contacts dynamics and friction-cone constraints are important for the success of the task.

friction cone. We use a) the binary matrix \mathbf{H} to determine each cone, and b) the gait transition matrix \mathbf{T} to activate the contact forces.

6) *Approximate Torque Limits*: To improve feasibility in the execution, we rely on the quasi-static assumption on each leg $\mathbf{J}_e^T \lambda_e \leq \tau_{max}$, constraining leg torques over the approximate bounds of the operational space Jacobian \mathbf{J}_e .

7) *Angular dynamics*: We exploit the relation presented in [1], in order to represent bilinear terms of the angular momentum rate as a difference of convex quadratic terms.

Given these constraints, we formulate a MICP and add costs on the acceleration, angular momentum rate, execution time, and negative friction cone stability margin.

III. EXPERIMENTAL VALIDATION

We validate our approach on the HyQ robot traversing challenging terrain tracks (see Fig. 1), which would not be easily handled by non-convex trajectory optimization techniques or without friction cone constraints. Our approach generates robust trajectories in all scenarios, where each locomotion cycle (4 time-slots or one stride) is optimized in about 0.5 secs.

REFERENCES

- [1] B. Ponton, A. Herzog, S. Schaal, and L. Righetti, "A convex model of humanoid momentum dynamics for multi-contact motion generation," in *Humanoid Robots (Humanoids), 2016 IEEE-RAS 16th International Conference on*. IEEE, 2016, pp. 842–849.
- [2] A. K. Valenzuela, "Mixed-integer convex optimization for planning aggressive motions of legged robots over rough terrain," Ph.D. dissertation, Massachusetts Institute of Technology, 2016.
- [3] B. Aceituno-Cabezas, H. Dai, J. Cappelletto, J. C. Grieco, and G. Fernandez-Lopez, "A mixed-integer convex optimization framework for robust multilegged robot locomotion planning over challenging terrain," in *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2017.

The authors are with (1) the Mechatronics Research Group, Simón Bolívar University {12-10764, cappelletto, jcgrico, gfernandez}@usb.ve, (2) is with the Toyota Research Institute hongkai.dai@tri.global, and (3) are with the Department of Advanced Robotics, Istituto Italiano di Tecnologia {firstname.surname}@iit.org