UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO
PROGRAMA DE MAESTRÍA Y DOCTORADO EN CIENCIAS MATEMÁTICAS Y DE LA ESPECIALIZACIÓN EN ESTADÍSTICA APLICADA.

Análisis crítico del algoritmo de Megiddo para dimensiones bajas y su aplicación al problema de sujeción robótica

TESIS
QUE PARA OPTAR POR EL GRADO DE:
MAESTRA EN CIENCIAS

PRESENTA:
ANDREA BUSTILLOS GOROSAVE

DIRECTOR DE LA TESIS:
DR. GILBERTO CALVILLO VIVES
INSTITUTO DE MATEMÁTICAS, UNIDAD CUERNAVACA

MÉXICO, D. F., Mayo 2016

Universidad Nacional Autónoma de México
UNAM
Biblioteca Central
Dirección General de Bibliotecas de la UNAM

UNAM – Dirección General de Bibliotecas
Tesis Digitales
Restricciones de uso

DERECHOS RESERVADOS ©
PROHIBIDA SU REPRODUCCIÓN TOTAL O PARCIAL

Todo el material contenido en esta tesis esta protegido por la Ley Federal del Derecho de Autor (LFDA) de los Estados Unidos Mexicanos (México).

El uso de imágenes, fragmentos de videos, y demás material que sea objeto de protección de los derechos de autor, será exclusivamente para fines educativos e informativos y deberá citar la fuente donde la obtuvo mencionando el autor o autores. Cualquier uso distinto como el lucro, reproducción, edición o modificación, será perseguido y sancionado por el respectivo titular de los Derechos de Autor.

Índice general

Introducción **4**

Notación **6**

1. Teoría de poliedros **7**
1.1. Conjuntos convexos 7
1.2. Conjuntos afines 8
1.3. Polaridad y función de soporte 12

2. Programación lineal en dimensiones bajas **20**
2.1. Conceptos básicos 20
2.2. Algoritmo de Megiddo 23
2.2.1. Algoritmo de Megiddo en $\mathbb{R}^2$ 25
2.2.2. Algoritmo de Megiddo en $\mathbb{R}^3$ 36
2.2.3. Complejidad computacional y generalización de algoritmos de Megiddo 50

3. El problema de sujeción robótica **53**
3.1. Preliminares 55
3.2. Los problemas de sujeción *force-closure* 58

4. Modeación de la sujeción robótica **60**
4.1. Modelación matemática 60
4.1.1. Fuerzas de contacto 60
4.1.2. Modelo de contacto 63
4.1.3. Discretización del cono de fricción 65
4.2. Prueba cualitativa para sujeción force-closure 66
4.2.1. Antecedentes de Liu y Wang 66
4.2.2. Propuesta de Zheng y Qian 68

5. Resultados e Implementación **74**
5.1. Resultados 74
5.2. Implementación 78

6. Conclusiones **80**

A. Topología de Espacios Afines **82**

B. Algoritmo mediana de medianas **84**

Intoducción

Este trabajo surge del estudio teórico del algoritmo en tiempo lineal para resolver problemas de programación lineal de dimensiones fijas de Megiddo y la curiosidad de explorar su eficiencia práctica. Debido a lo anterior se consideró para su aplicación el problema de sujeción robótica, el cual se puede modelar como un problema de programación lineal de dimensión fija y relativamente baja.

Lo atractivo del enfoque de Megiddo, es que en cada subrutina realiza una cantidad lineal de trabajo y va reduciendo por una fracción constante el número de restricciones en cada iteración; para problemas de dimensión menor o igual a tres lo anterior resulta en un algoritmo de complejidad lineal sobre el número de restricciones [30]. Este enfoque para resolver problemas de programación lineal fue estudiado de manera paralela por Dyer en [14]. En [31] Megiddo expuso la generalización de este algoritmo para dimensiones mayores fijas; posteriormente, Dyer en [15] y Clarkson en [9] presentaron mejoras en la eficiencia computacional del algoritmo. Sin embargo, se decidió implementar el algoritmo original de Megiddo debido a su relativa sencillez con respecto a las versiones mejoradas. Cabe destacar a pesar de la búsqueda realizada no se encontraron implementaciones del algoritmo disponibles públicamente, por lo cual es imposible comparar los resultados obtenidos en este trabajo con otra implementación del mismo algoritmo.

Se realizó un estudio minucioso del artículo [30] donde Megiddo dio a conocer su algoritmo en tiempo lineal para problemas de programación lineal de dimensiones dos y tres. Para verificar la eficiencia del método, como parte de esta tesis se realizaron implementaciones del algoritmo en Python tanto para problemas de programación lineal en el plano como en el espacio. Cabe mencionar que éste fue el proceso más laborioso en este proyecto y el que requirió de más tiempo y atención debido a la vaguedad y falta de detalles en ciertos puntos en el artículo del Megiddo. Es importante la programación correcta de este algoritmo ya que una interpretación errónea podría aumentar la complejidad del algoritmo.

Por otra parte, el estudio y análisis de sujeciones robóticas es un área que se ha desarrollado en las últimas tres décadas, ya que la sustitución de la mano de obra por manos robóticas ha ido dominando los procesos de producción en masa, y encontrando aplicaciones en distintos campos como en la medicina, la

industria o en el sector militar. En general, la meta del estudio de sujeción robótica es lograr la inmobilización de un objeto a través de un conjunto adecuado de contactos (dedos), independientemente de si el objeto interactúa con fuerzas externas o no. Son muchos los factores que se involucran en la sujeción de un objeto, y cada uno de ellos da pie a problemas físicos concretos. El problema de interés de este trabajo es el de determinar si una sujeción presenta la propiedad de *force-closure*, que a grosso modo indica la "estabilidad " de la sujeción frente a perturbaciones sobre el objeto; esto quiere decir que si un objeto es sujeto por una mano robótica, ésta es capaz de aplicar fuerzas a través de los dedos para balancear cualquier perturbación razonable ejercida sobre el objeto en forma de fuerza o torque externos.

Desde el punto de vista matemático, el problema de la sujeción robótica es interesante porque su formulación como problema de programación lineal no es trivial; requiere un tratamiento cuidadoso de la teoría de poliedros así como nociones básicas de física. Para resolver el problema de identificar si una sujeción tiene la propiedad de *force-closure* se recurrió al método desarrollado por Zheng y Qian, quienes en [60] introdujeron un algoritmo de carácter cualitativo, el cual hace uso de la programación lineal. Para estudiar este planteamiento se revisa el trabajo realizado por Liu y Wang en [29], quienes inspiran el trabajo de los primeros. El trabajo de Zheng y Qian se presenta como una version mejorada y revisada de la propuesta de Liu y Wang, por lo tanto es importante saber como nace este enfoque.

La estructura de la tesis es la siguiente: en el capítulo 1 se describe la teoría de poliedros y polaridad de conjuntos necesaria tanto para la construcción del algoritmo de Megiddo como el desarrollo de la propuesta de Zheng y Qian. A continuación, se dedica el capítulo 2 al estudio de programación lineal en dimensiones bajas: primero se exponen los conceptos básicos de programación lineal; posteriormente, se realiza un profundo análisis de cada una de las partes que componen el algoritmo de Megiddo para problemas de dimensión dos y tres.

En el capítulo 3 se da una introducción al problema de la sujeción robótica. Se inicia con un marco histórico y se presentan los conceptos de robótica necesarios para la descripción de un modelo de sujeción; posteriormente, se dan a conocer los principales problemas relacionados con sujeción robótica y como se ha atacado cada uno de ellos, concluyendo así el marco de trabajo.

En el capítulo 4 se hace una explicación formal de la modelación del problema de interés: la propiedad de *force-closure* de una sujeción robótica, así como su interpretación como problema de programación lineal. Finalmente, los capítulos 5 y 6 presentan los resultados y conclusiones de la implementacion del algoritmo de Megiddo para determinar la propiedad de *force-closure* de distintas sujeciones de objetos varios en el plano.

Notación

La notación presentada a continuación se seguirá a lo largo de este trabajo, a menos se especifique lo contrario.

- Todas las letras mayúsculas negritas representan matrices.
- Todas las letras minúsculas negritas representan vectores.
- Todos los vectores serán vectores columna.
- Las letras griegas simbolizan escalares.

$\mathbf{A}$	-	Coeficientes de la matriz de un problema de programación lineal .
$\mathbf{A}_i$	-	i-ésimo renglón de la matriz $\mathbf{A}$.
$aff(S)$	-	Envoltura afín del conjunto S.
$b(S)$	-	Frontera del conjunto S.
$B(\boldsymbol{x}, r)$	-	Bola centrada en el punto $\mathbf{x}$ y con radio $\mathbf{r}$.
$\mathbf{c}$	-	coeficientes de la función objetivo.
$cl(S)$	-	Cerradura del conjunto S.
$conv(S)$	-	Envoltura convexa del conjunto S.
$dim(S)$	-	Dimensión del conjunto S.
H_-	-	Semiplano inferior.
H_+	-	Semiplano superior.
$int(S)$	-	Interior del conjunto S.
$[m]$	-	Conjunto con todos los números naturales hasta el m.
$\lfloor m \rfloor$	-	La parte entera de m.
$\mathbb{R}^n$	-	Espacio real de dimensión n.
$\mathbb{R}^+$	-	Espacio real positivo.
$ri(S)$	-	Interior relativo del conjunto S.
$rb(S)$	-	Frontera relativa del conjunto S.
$\hat{S}$	-	Conjunto polar de S restringido al afín de éste.
S^*	-	Conjunto polar del conjunto S.
$\mathbf{x}^T$	-	Transpuesta del vector $\mathbf{x}$.
x_i	-	i-ésima coordenada del vector $\mathbf{x}$.
$\|\|\mathbf{x}\|\|$	-	Norma euclidiana del vector $\mathbf{x}$.

Capítulo 1

Teoría de poliedros

En este capítulo se presentan las definiciones y resultados de poliedros y análisis convexo necesarios para estudiar el problema de sujeción robótica y el algoritmo de Megiddo. Son de particular interés un conjunto de resultados de análisis convexo relacionados con el interior relativo y la función de soporte de un conjunto, ya que éstos son cruciales para la validación del algoritmo propuesto por Zheng y Qian, y se utilizarán en el capítulo 4. Para repasar conceptos necesarios de topología refiérase al apéndice A. El contenido de esta sección está basado en los textos de Rockafellar [49] y Dahl [10].

1.1. Conjuntos convexos

Supóngase que S es un subconjunto de $\mathbb{R}^n$. Se dice que S es un *conjunto convexo* si dados $\mathbf{x}_1, \mathbf{x}_2 \in S$

$$(1-\lambda)\mathbf{x}_1 + \lambda\mathbf{x}_2 \in S, \tag{1.1}$$

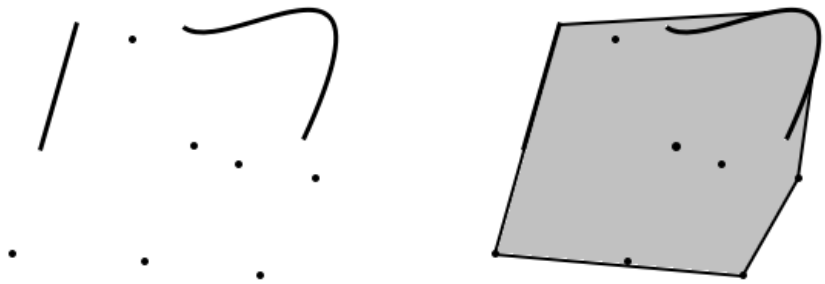
para todo $0 \leq \lambda \leq 1$. Geométricamente, un conjunto es convexo si para todo par de puntos en él, el segmento de recta que los une está contenido en S. Estos conjuntos tienen características interesantes, ya que la propiedad de convexidad se preserva bajo distintas operaciones como suma de conjuntos, intersección arbitraria de conjuntos y transformaciones afines [10].

Dado un conjunto de vectores $\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_t \in \mathbb{R}^n$, a un vector de la forma

$$\mathbf{x} = \sum_{j=1}^{t} \lambda_j \mathbf{x}_j, \tag{1.2}$$

$$\sum_{j=1}^{t} \lambda_j = 1, \tag{1.3}$$

$$\lambda_1, \lambda_2, ..., \lambda_t \in \mathbb{R} \text{ tal que } \lambda_j \geq 0 \tag{1.4}$$



(a) Conjunto S **(b)** *conv(S)*

Figura 1.1: Envoltura convexa de un conjunto de puntos.

se le llama *combinación convexa* de $\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_t$. En general, si $S \subset \mathbb{R}^n$ se dice que $\mathbf{x}$ es una combinación convexa de elementos de S si se puede expresar como una combinación convexa de un número finito de elementos de S. Al conjunto $conv(S)$ de todas las combinaciones convexas de elementos de S se le llama la *envoltura convexa* de S. En la figura 1.1 se muestra un conjunto S y su envoltura convexa. Nótese que $conv(S)$ es un conjunto convexo, independientemente de si S lo es. Para un conjunto S cualquiera, $conv(S)$ se puede expresar como la intersección de todos los conjuntos convexos que contienen a S.

Dado un conjunto de vectores $\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_t \in \mathbb{R}^n$ y una expresión de la forma (1.2) donde los coeficientes λ_j cumplen la condición (1.4), entonces se dice que $\mathbf{x}$ es una *combinación no negativa* de los vectores $\mathbf{x}_j$ para $j \in [t]$. Si para dicha expresión los coeficientes son de la forma $\lambda_j > 0$ para todo índice, entonces se dice que $\mathbf{x}$ es un *combinación positiva* de $\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_t$.

1.2. Conjuntos afines

Definición 1.2.1. *Se le da el nombre de* conjunto *o* espacio afín *a aquellos conjuntos $S \subseteq \mathbb{R}^n$ tal que para toda pareja de elementos $\boldsymbol{x}_1, \boldsymbol{x}_2 \in S$ se satisface la condición (1.1) con $\lambda \in \mathbb{R}$.*

Esto implica que toda la recta que pasa por estos dos puntos pertenezca a S. Obsérvese que en particular todo conjunto afín es convexo. Dado un conjunto de elementos $\{\mathbf{x}_1, ..., \mathbf{x}_t\} \in \mathbb{R}^n$, a los vectores de la forma

$$\mathbf{x} = \sum_{j=1}^{t} \lambda_j \mathbf{x}_j, \quad \text{donde} \quad \sum_{j=1}^{t} \lambda_j = 1, \quad \lambda_j \in \mathbb{R} \quad \forall j \in [t]$$

se llaman *combinaciones afines* de $\mathbf{x}_1, ..., \mathbf{x}_t$. Así se define la *envoltura afín, aff*(S), de un conjunto S como el conjunto de todas las combinaciones afines de sus elementos (véase la figura 1.2). De manera análoga al concepto de envoltura con-

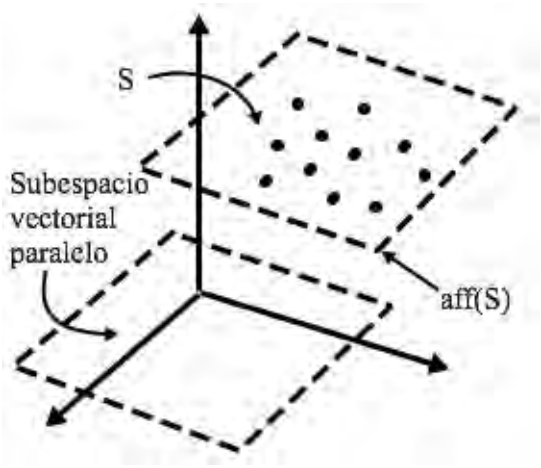


Figura 1.2: Envoltura afín de un conjunto de puntos y el subespacio vectorial paralelo a ésta.

vexa, se puede definir $aff(S)$ como la intersección de todos los conjuntos afines que contienen a S. Si se tienen $\mathbf{x}_1, ..., \mathbf{x}_t$ en $\mathbb{R}^n$ distintos, se dice que estos vectores son *afinmente independientes* si

$$\sum_{j=1}^{t} \lambda_j \mathbf{x}_j = 0 \text{ y } \sum_{j=1}^{t} \lambda_j = 0 \tag{1.5}$$

implican que $\lambda_i = 0$ para todo índice. Si $\mathbf{x}_1, ..., \mathbf{x}_t$ en $\mathbb{R}^n$ es un conjunto afinmente independiente maximal entonces es una *base afín*. Se observa que esta concepto de independencia afín resulta muy similar a la definición de independencia lineal. A continuación, se presenta un resultado que deja claro cómo se relacionan estos dos conceptos.

Proposición 1.2.2. *Los vectores $\mathbf{x}_1, ..., \mathbf{x}_t \in \mathbb{R}^n$ son afinmente independientes si y solo si los vectores $\mathbf{x}_2 - \mathbf{x}_1, ..., \mathbf{x}_t - \mathbf{x}_1$ son linealmente independientes.*

Demostración. Un conjunto de vectores $\{\mathbf{x}_1, ..., \mathbf{x}_t\} \subseteq \mathbb{R}^n$ es afinmente independiente si el cumplimiento de la condición (1.5) implica que $\lambda_j = 0$ para todo $j \in [t]$. Considerando entonces que $\sum_{j=1}^{t} \lambda_j \mathbf{x}_1 = \mathbf{0}$ se tiene

$$\sum_{j=1}^{t} \lambda_j \mathbf{x}_j = \sum_{j=1}^{t} \lambda_j \mathbf{x}_j - \sum_{j=1}^{t} \lambda_j \mathbf{x}_1 = \sum_{j=2}^{t} \lambda_j (\mathbf{x}_j - \mathbf{x}_1) = \mathbf{0},$$

siempre y cuando $\lambda_j = 0$, $\forall j \in [t]$. Esto implica que $\{\mathbf{x}_2 - \mathbf{x}_1, ...\mathbf{x}_t - \mathbf{x}_1\}$ es un conjunto linealmente independiente en $\mathbb{R}^n$. La suficiencia resulta naturalmente de forma inversa. □

Un corolario del resultado anterior expresa la relación que existe entre conjuntos afines y subespacios vectoriales de $\mathbb{R}^n$, y se describe a continuación.

Corolario 1.2.3. *Los subespacios lineales de $\mathbb{R}^n$ son conjuntos afines que contienen al origen.*

Demostración. Este resultado se obtiene al considerar $\mathbf{x}_1 = \mathbf{0}$ en la proposición anterior.

□

También existe una relación entre conjuntos afines y ecuaciones lineales:

Definición 1.2.4. *El conjunto* H *de soluciones de una ecuación de la forma $a\boldsymbol{x}^T = b$ con $a \in \mathbb{R}^n$, $b \in \mathbb{R}$ y $a \neq 0$ es llamado* hiperplano.

Es fácil ver por la linealidad de la ecuación que H es un conjunto afín. Otros conjuntos de importancia para este trabajo son los semiespacios cerrados:

Definición 1.2.5. *Un* semiespacio cerrado *o* medio espacio cerrado $H_-(H_+)$ *es el conjunto de soluciones de una desigualdad de la forma $\boldsymbol{c}^T\boldsymbol{x} \leq b$ ($\boldsymbol{c}^T\boldsymbol{x} \geq b$), con $\boldsymbol{c} \neq \boldsymbol{0}$.*

Si la desigualdad es estricta el semiespacio es llamado *abierto*. A H_+ se le llama *semiespacio superior* y a H_- *semiespacio inferior*. En general en este trabajo se trabajará con semiespacios cerrados, por lo que se les hará referencia simplemente como semiespacios, a menos de que se especifique de otra manera. Íntimamente relacionado con programación lineal, se tiene el concepto de poliedro:

Definición 1.2.6. *Un* poliedro convexo *o simplemente* poliedro *es la intersección finita de un conjunto de medios-espacios. Un* politopo *es un poliedro acotado [42].*

Por otra parte, existe una relación entre dos conjuntos determinada por la traslación en el espacio de uno de ellos. Dado un conjunto $S \subseteq \mathbb{R}^n$ y $a \in \mathbb{R}^n$ se define la *traslación* de S por a, $S + a$, como

$$S + a = \{x + a \,:\, x \in S\}.$$

Un par de conjuntos afines S,L son *paralelos* si $S = L + a$ con $a \in S$. En particular todo conjunto afín es paralelo a un subespacio vectorial único (figura 1.2) como lo determina la proposición siguiente.

Proposición 1.2.7. *Si $S \subseteq \mathbb{R}^n$ es un conjunto afín no vacío, entonces S es paralelo a un único subespacio vectorial L. Más aún, S puede ser expresado como*

$$S = L + \boldsymbol{x}_0 = \{\boldsymbol{x} + \boldsymbol{x}_0 \,:\, \boldsymbol{x} \in L\},$$

donde L es un subespacio vectorial de $\mathbb{R}^n$ y $\boldsymbol{x}_0 \in S$.

Demostración. Sea $\mathbf{x}_0 \in S$, constrúyase el conjunto

$$L = \{\mathbf{y} \in \mathbb{R}^n \,:\, \mathbf{y} = \mathbf{x} - \mathbf{x}_0, \quad \mathbf{x} \in S\}.$$

Nótese que L es un espacio afín que además contiene al origen, así por el corolario 1.2.3 se tiene que L es un espacio vectorial. Ahora, supóngase que S es paralelo a dos subespacios vectoriales L_1, L_2. Esto implica que

$$S = L_1 + \mathbf{x}_0 = L_2 + \mathbf{x}_0,$$

para algunos $\mathbf{x}_0 \in \mathbb{R}^n$. Entonces ambos subespacios son paralelos y por consiguiente se pueden expresar de la forma $L_1 = L_2 + \mathbf{x}_0 - \mathbf{x}_0$. Por lo tanto el espacio vectorial es único.

□

Esta traslación define un isomorfismo entre L y S donde el nuevo origen está dado por $\mathbf{x}_0$. Ahora, gracias a la proposición anterior se puede dar las siguientes definiciones.

Definición 1.2.8. *La* dimensión de un conjunto afín *es la dimensión de su subespacio paralelo.*

Definición 1.2.9. *La dimensión de un conjunto $S \subseteq \mathbb{R}^n$ se define como la dimensión de aff(S).*

La dimensión de un conjunto se denota por $dim(S)$. Si para $S \subseteq \mathbb{R}^n$ se tiene que $dim(S) = n$ entonces se dice que S es de *dimensión completa.* El siguiente teorema es uno de los principales resultados de análisis convexo y se relaciona con la dimensión de un conjunto.

Teorema 1.2.10. *(Carathéodory) Sea $S = \{\mathbf{x}_1, ..., \mathbf{x}_t\} \subseteq \mathbb{R}^n$. Entonces todo elemento $\mathbf{x} \in conv(S)$ se puede expresar como una combinación convexa de a lo más $n+1$ puntos (afinmente independientes) en S.*

Demostración. Si $\mathbf{x} \in conv(S)$ se puede expresar como

$$\mathbf{x} = \sum_{j=1}^{t} \lambda_j \mathbf{x}_j, \; \sum_{j=1}^{t} \lambda_j = 1, \quad \lambda_j > 0. \tag{1.6}$$

Claramente si $\mathbf{x}_1, ...\mathbf{x}_t$ son afinmente independientes el resultado es inmediato, por lo tanto supóngase lo contrario. Esto implica que $t > n+1$. Definiendo $\alpha_1 = -\sum_{j=2}^{t} \alpha_j$, se tiene que

$$\sum_{j=1}^{t} \alpha_j \mathbf{x}_j = \mathbf{0}, \quad \sum_{j=1}^{t} \alpha_j = 0. \tag{1.7}$$

De las ecuaciones (1.6) y (1.7) se puede expresar a $\mathbf{x}$ como

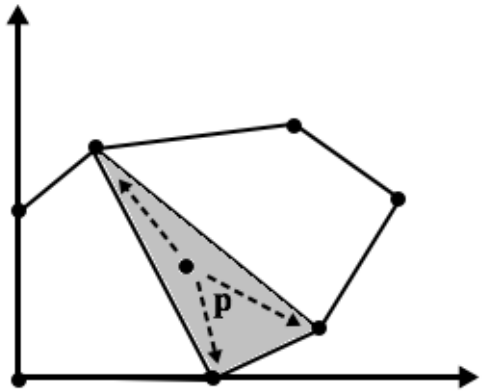


Figura 1.3: Ilustración del teorema de Carathéodory: para definir el punto **p** en el plano, se necesitan a lo sumo los tres puntos señalados.

$$\mathbf{x} = \sum_{j=1}^{t} \lambda_j \mathbf{x}_j - \beta \mathbf{0} = \sum_{j=1}^{t} (\lambda_j - \beta\alpha_j)\mathbf{x}_j, \tag{1.8}$$

para cualquier $\beta \in \mathbb{R}$. En particular, si $\beta = \text{mín}_{j\in[t]}\{\frac{\lambda_j}{\alpha_j} \ : \ \alpha_j > 0\}$ entonces todos los coeficientes en (1.8) son no negativos y $\lambda_k - \beta\alpha_k = 0$ para alguna k. Esto implica que $\mathbf{x}$ se puede expresar como una combinación convexa de $t-1$ puntos de S. Si $t-1 \leq n$ entonces $\mathbf{x}$ es la combinación convexa de un conjunto de vectores linealmente independientes y el resultado se sigue de inmediato. En caso contrario, se repite este proceso, hasta poder describir a $\mathbf{x}$ como una combinación convexa de a lo más $n+1$ elementos. □

Uniendo los resultados del teorema anterior y la proposición 1.2.2 se tiene que para un conjunto finito S se puede expresar $dim(conv(S))$ como el número máximo de puntos afinmente independientes en S menos 1. Intuitivamente la dimensión de un conjunto convexo S expresa los grados de libertad para moverse dentro del conjunto sin salirse de él.

1.3. Polaridad y función de soporte

A continuación, se presentan los resultados de polaridad necesarios para las construcciones y demostraciones en los siguientes capítulos. Para los conceptos y resultados necesarios de topología se hace referencia al apéndice A.

A lo largo de esta sección, el conjunto S va a denotar un conjunto no vacío convexo en $\mathbb{R}^n$, $int(S)$ y $ri(S)$ indican el interior y el interior relativo de S respectivamente. En el siguiente teorema se describe la relación entre los elementos de un conjunto convexo y el interior relativo del mismo.

Teorema 1.3.1. *Sea* $S = conv(\{\mathbf{x}_1, ..., \mathbf{x}_t\})$ *con* $\mathbf{x}_i \in \mathbb{R}^n$ *para todo* $i \in [t]$. *Entonces toda combinación convexa positiva de* $\{\mathbf{x}_1, ..., \mathbf{x}_t\}$ *está en* $ri(S)$.

Demostración. Supóngase sin pérdida de generalidad $\{\mathbf{x}_1, ..., \mathbf{x}_e\}$ una base afín de $\{\mathbf{x}_1, ..., \mathbf{x}_t\}$ y $T = conv(\{\mathbf{x}_1, ..., \mathbf{x}_e\})$. Se puede definir la frontera relativa de T $rb(T)$, como la unión de subconjuntos llamados T_i (ver figura 1.4) donde

$$T_i = \{\mathbf{y} \in T : \mathbf{y} = \sum_{j=1}^{e} \lambda_j \mathbf{x}_j,\ \lambda_i = 0,\ \sum_{j=1}^{e} \lambda_j = 1\}.$$

Así, $rb(T) = \cup_{i=1}^{e} T_i$. Si $\mathbf{x} = \sum_{j=1}^{e} \lambda_i \mathbf{x}_j$ es una combinación convexa positiva de elementos de la base, entonces $\mathbf{x} \in T \subseteq S$. Sin embargo, por ser todos los coeficientes λ_j de $\mathbf{x}$ positivos, resulta que $\mathbf{x} \notin T_i$ para todo $i \in [e]$ y consecuentemente $\mathbf{x} \notin rb(T)$. Así, $\mathbf{x}$ es un punto interior relativo de T y por lo tanto de S.

En la figura 1.4 se muestra un ejemplo de la demostración anterior. En este caso los puntos $\mathbf{x}_1, \mathbf{x}_2$ y $\mathbf{x}_3$ definen una base afín del conjunto S y su envoltura convexa T se presenta sombreada; claramente el hecho de que $\mathbf{x}$ sea combinación convexa positiva de la base afín indica que es un punto interior de T y por lo tanto de S. □

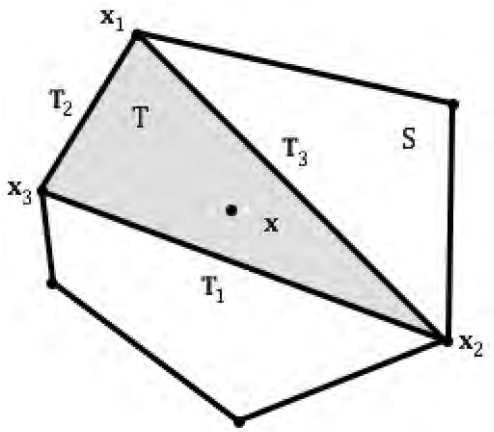


Figura 1.4: Ilustración de la demostración del teorema 1.3.1.

Definición 1.3.2. *Dado un conjunto* $S \subseteq \mathbb{R}^n$ *se define el* conjunto polar S^* *de* S, *como*

$$S^* = \{\mathbf{x} \in \mathbb{R}^n : \mathbf{y}^T\mathbf{x} \leq 1,\ \forall\, \mathbf{y} \in S\}.$$

En la figura 1.5 se presenta un conjunto convexo S (poliedro gris), su polar S^* (poliedro negro) y el círculo unitario se muestra punteado; de aquí se puede

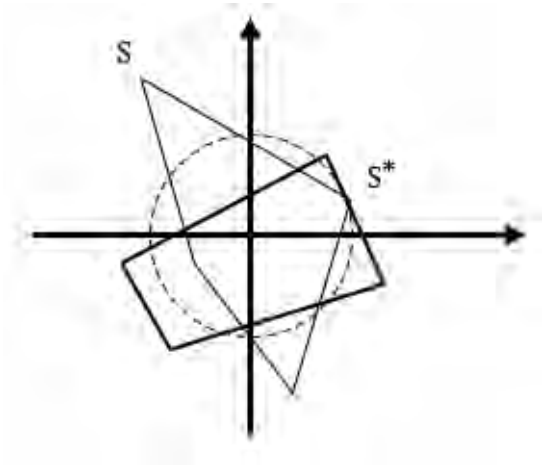


Figura 1.5: Ilustración de un conjunto S (poliedro delgado) y su conjunto polar S^* (poliedro grueso) en relación al círculo unitario (círculo punteado).

observar que los vértices de S más alejados del círculo unitario generan hiperplanos cercanos al origen y viceversa. La polaridad crea una dualidad entre los elementos de S y su polar de la siguiente manera: a un punto $\mathbf{x}$ de S le corresponde un semiespacio polar $H_{\mathbf{x}}$ conformado por todos los $\mathbf{y} \in \mathbb{R}^n$ tal que $\mathbf{x}^T\mathbf{y} \leq 1$, y de manera semejante a cada semiespacio en S le corresponde un punto $\mathbf{x}_H$.

Se tienen tres observaciones importantes sobre conjuntos polares: la primera es que si S es acotado convexo y contiene al origen, $(S^*)^* = S$ [52]; la segunda es que el conjunto polar de cualquier conjunto es siempre convexo por ser la intersección de semiespacios y finalmente, el conjunto polar del origen es todo el espacio. El conjunto polar es de particular interés ya que por sus propiedades, ayudan a simplificar el problema de sujeción robótica planteado por Zheng y Qian. A continuación, se demuestran ciertas propiedades del conjunto polar de un conjunto y se muestra su relación con los puntos interiores del conjunto original.

Lema 1.3.3. *Si $r > 0$ finito, entonces $B(\boldsymbol{0}, \frac{1}{r}) \subseteq B(\boldsymbol{0}, r)^*$.*

Demostración. Si $\mathbf{y} \in B(\mathbf{0}, \frac{1}{r})$ se puede expresar de la forma $\mathbf{y} = \frac{\mathbf{x}}{(r+\epsilon)||\mathbf{x}||}$ para cualquier $\mathbf{x} \in \mathbb{R}^n$ y $\epsilon > 0$. En particular, supóngase que $\mathbf{x} \in B(\mathbf{0}, r)$. Entonces,

$$\mathbf{y}^T\mathbf{x} = \frac{\mathbf{x}^T}{(r+\epsilon)||\mathbf{x}||}\mathbf{x} = \frac{||\mathbf{x}||}{(r+\epsilon)}.$$

Puesto que $||x|| < r$, de la ecuación anterior resulta que $\mathbf{y}^T\mathbf{x} < 1$ por lo que $\mathbf{y} \in B(\mathbf{0}, r)^*$. □

Lema 1.3.4. *Si $S \subseteq \mathbb{R}^n$ y $T \subseteq S$ entonces $S^* \subseteq T^*$.*

Demostración. Si $\mathbf{x} \in S^*$ entonces por la definición 1.3.2 $\mathbf{y}^T\mathbf{x} \leq 1$ para todo $y \in S$. En particular, $T \subseteq S$ lo que implica que $\mathbf{y}^T\mathbf{x} \leq 1$ para todo $y \in T$ y así $\mathbf{x} \in T^*$. Luego $S^* \subseteq T^*$. □

Estas propiedades sintetizan el comportamiento de los conjuntos polares de bolas y conjuntos contenidos entre sí de una manera intuitiva. Además dado un conjunto S, se tiene un simple resultado para determinar la presencia del origen en el conjunto polar S^*.

Teorema 1.3.5. *Si $S \subseteq \mathbb{R}^n$ es acotado y $\mathbf{0} \in ri(S)$, entonces $\mathbf{0} \in int(S^*)$.*

Demostración. Como S es acotado y contiene al origen, entonces existe un $r > 0$ finito tal que $S \subseteq B(\mathbf{0}, r)$. Por el lema 1.3.4 se tiene que $B(\mathbf{0}, r)^* \subseteq S^*$ y por el lema 1.3.3 $B(\mathbf{0}, \frac{1}{r}) \subseteq B(\mathbf{0}, r)^* \subseteq S^*$. Luego $\mathbf{0} \in int(S^*)$. □

De ahora en adelante se va a restringir el estudio de S^* al espacio afín donde vive S, por lo que resulta pertinente presentar la siguiente definición.

Definición 1.3.6. *Dado un conjunto $S \subseteq \mathbb{R}^n$ se define el conjunto $\hat{S}$ como el conjunto polar S^* restringido al espacio afín aff(S), es decir $\hat{S} = S^* \cap aff(S)$.*

A continuación, se muestra que $\hat{S}$ es compacto y tiene una interesante relación con S^* si el origen pertenece a S.

Teorema 1.3.7. *Sea $S \subseteq \mathbb{R}^n$ compacto tal que $0 \in ri(S)$. Entonces $\hat{S}$ es un conjunto compacto convexo.*

Demostración. Como $\mathbf{0} \in ri(S)$, existe $r > 0$ tal que $B(\mathbf{0}, r) \cap aff(S) \subset S$. Si $\mathbf{x} \in \hat{S}$ y $\epsilon > 0$, entonces $\mathbf{y} = (r - \epsilon)\frac{\mathbf{x}}{||\mathbf{x}||} \in B(\mathbf{0}, r) \cap aff(S)$, por lo que $\mathbf{y}$ es un elemento de S. En particular $\mathbf{x} \in S^*$ y por consiguiente resulta que

$$\mathbf{y}^T\mathbf{x} = (r - \epsilon)\frac{\mathbf{x}}{||\mathbf{x}||}^T\mathbf{x} = (r - \epsilon)||\mathbf{x}|| \leq 1.$$

Esto implica que $||\mathbf{x}|| \leq \frac{1}{r-\epsilon}$ o bien que la norma de $\mathbf{x}$ está acotada. Luego $\hat{S}$ es acotado. Ahora como S es compacto, para todo $\mathbf{y} \in S$ considérese el medio espacio cerrado $H_{\mathbf{y}} = \{\mathbf{x} \in aff(S) \ : \ \mathbf{y}^T\mathbf{x} \leq 1\}$. Como la intersección arbitraria de conjuntos cerrados es cerrado resulta que

$$\bigcap_{\mathbf{y} \in S} H_{\mathbf{y}} = \{\mathbf{x} \in aff(S) \ : \ \mathbf{y}^T\mathbf{x} \leq 1, \ \forall \mathbf{y} \in S\} = \hat{S},$$

por lo que $\hat{S}$ es cerrado. Luego $\hat{S}$ es compacto. Finalmente como S^* y *aff(S)* son conjuntos convexos y la intersección de conjuntos convexos es convexa, se sigue que $\hat{S}$ es convexo. □

Teorema 1.3.8. *Si $S \subseteq \mathbb{R}^n$ es convexo tal que $\mathbf{0} \in S$, entonces $S^* = \hat{S} + aff(S)^{\perp}$.*

Demostración. Por el corolario 1.2.3 el hecho de que $\mathbf{0} \in S$ implica que *aff(S)* es un subespacio vectorial de $\mathbb{R}^n$ y por lo tanto $aff(S)^{\perp}$ es simplemente el complemento ortogonal de *aff(S)*.

Supóngase que $\mathbf{x} \in \hat{S} + aff(S)^{\perp}$, entonces se puede expresar $\mathbf{x}$ como

$$\mathbf{x} = \mathbf{x}_1 + \mathbf{x}_2,$$

donde $\mathbf{x}_1 \in \hat{S}$ y $\mathbf{x}_2 \in aff(S)^{\perp}$. Así, para $\mathbf{y} \in S \subseteq aff(S)$ se tiene que

$$\mathbf{y}^T\mathbf{x} = \mathbf{y}^T(\mathbf{x}_1 + \mathbf{x}_2) = \mathbf{y}^T\mathbf{x}_1 + \mathbf{y}^T\mathbf{x}_2 = \mathbf{y}^T\mathbf{x}_1 \leq 1.$$

Luego, $\mathbf{x} \in S^*$.

Ahora para demostrar la suficiencia supóngase que $\mathbf{x} \in S^*$. Por ser *aff(S)* un subespacio vectorial se puede reescribir $\mathbf{x}$ como $\mathbf{x} = \mathbf{x}_1 + \mathbf{x}_2$ donde $\mathbf{x}_1 \in aff(S)$ y $\mathbf{x}_2 \in aff(S)^{\perp}$. Así para $\mathbf{y} \in S$

$$\mathbf{y}^T\mathbf{x} = \mathbf{y}^T\mathbf{x}_1 + \mathbf{y}^T\mathbf{x}_2 = \mathbf{y}^T\mathbf{x}_1 \leq 1,$$

lo que implica que $\mathbf{x}_1 \in S^*$ y por consiguiente $\mathbf{x}_1 \in \hat{S}$. Se sigue que $\mathbf{x} \in \hat{S} + aff(S)^{\perp}$ y finalmente que $S^* = \hat{S} + aff(S)^{\perp}$. □

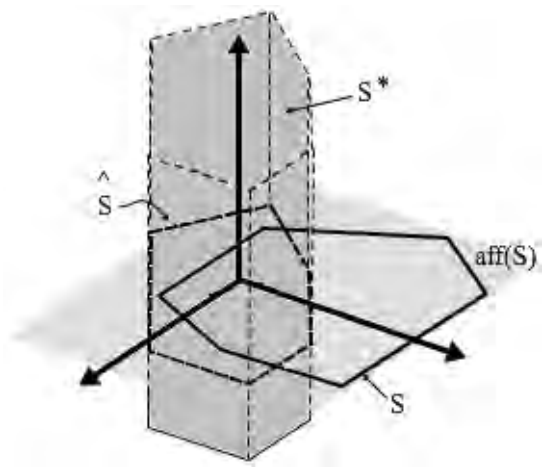


Figura 1.6: Conjunto S (polígono negro) junto con su polar S^* (poliedro punteado) y el polar restringido a *aff(S)* (polígono punteado); $aff(S)$ se muestra como un plano sombreado.

Una interpretación geométrica del teorema anterior se presenta en la figura 1.6. En ésta se muestra el conjunto S (polígono negro) de dimensión no completa, por lo tanto su conjunto polar S^* (poliedro punteado) resulta en un poliedro proyectado sobre la dimensión faltante; finalmente la restricción del polar a *aff(S)*, $\hat{S}$, se muestra como el poliedro punteado.

Definición 1.3.9. *La* función de soporte p_S *de un conjunto* $S \subseteq \mathbb{R}^n$ *se define como*

$$p_S(\mathbf{z}) = \sup_{\mathbf{x} \in S} \mathbf{z}^T \mathbf{x},$$

para todo $\mathbf{z} \in \mathbb{R}^n$.

De manera análoga se puede definir la función soporte p_{S^*}. Para los siguientes resultados, se considerará que S es un conjunto compacto convexo tal que $\mathbf{0} \in ri(S)$.

Teorema 1.3.10. *La función de soporte* $p_{S^*}(\mathbf{z})$ *es finita si y solo si* $\mathbf{z} \in aff(S)$.

Demostración. Por el teorema 1.3.8, todo $\mathbf{x} \in S^*$ puede expresarse como $\mathbf{x} = \mathbf{x}_1 + \mathbf{x}_2$ con $\mathbf{x}_1 \in \hat{S}$ y $\mathbf{x}_2 \in aff(S)^\perp$.

Para demostrar la suficiencia supóngase que $\mathbf{z} \notin aff(S)$. Expresando $\mathbf{z}$ de la forma $\mathbf{z} = \mathbf{z}_1 + \mathbf{z}_2$ con $\mathbf{z}_1 \in aff(S)$ y $\mathbf{z}_2 \in aff(S)^\perp$, resulta

$$\begin{aligned} p_{S^*}(\mathbf{z}) &= \sup_{\mathbf{x}_1+\mathbf{x}_2 \in S^*} (\mathbf{z}_1^T\mathbf{x}_1 + \mathbf{z}_1^T\mathbf{x}_2 + \mathbf{z}_2^T\mathbf{x}_1 + \mathbf{z}_2^T\mathbf{x}_2) \\ &= \sup_{\mathbf{x}_1+\mathbf{x}_2 \in S^*} (\mathbf{z}_1^T\mathbf{x}_1 + \mathbf{z}_2^T\mathbf{x}_2) \\ &= \sup_{\mathbf{x}_1 \in \hat{S}} \mathbf{z}_1^T\mathbf{x}_1 + \sup_{\mathbf{x}_2 \in aff(S)^\perp} \mathbf{z}_2^T\mathbf{x}_2. \end{aligned}$$

Por el teorema 1.3.7, $\hat{S}$ es compacto convexo. De topología se sabe que toda función real continua definida en un conjunto compacto alcanza sus valores extremos y por ser el producto interno continuo, la función $p_{S^*}(\mathbf{z})$ restringida a $\hat{S}$ alcanza su máximo y por lo tanto el valor de $\mathbf{z}_1^T\mathbf{x}_1$ es finito [38]. Por otra parte, $aff(S)^\perp$ es un subespacio vectorial y consecuentemente no acotado; esto implica que $\mathbf{z}_2^T\mathbf{x}_2$ puede crecer arbitrariamente. Por lo tanto si $\mathbf{z} \notin aff(S)$ entonces $p_{S^*}(\mathbf{z})$ no es finita.

Inversamente, si $\mathbf{z} \in aff(S)$ entonces,

$$p_{S^*}(\mathbf{z}) = \sup_{\mathbf{x}_1+\mathbf{x}_2 \in S^*} (\mathbf{z}^T\mathbf{x}_1 + \mathbf{z}^T\mathbf{x}_2) = \sup_{\hat{S}} \mathbf{z}^T\mathbf{x}_1.$$

Los argumentos anteriores demuestran que $\sup_{\hat{S}} \mathbf{z}_1^T\mathbf{x}_1$ es finita.

Por lo tanto $p_{S^*}(\mathbf{z})$ es finita si y solo si $\mathbf{z} \in aff(S)$. □

El teorema anterior concluye que si se evalúa la función de soporte p_{S^*} en *aff(S)* la función se puede reescribir como

$$p_{S^*}(\mathbf{z}) = \operatorname*{máx}_{\mathbf{x} \in S^*} \mathbf{z}^T\mathbf{x}.$$

El siguiente corolario presenta resultados que se obtienen de manera natural del teorema anterior; más aún, se probará que si $\mathbf{z} \in aff(S)$ distinto de $\mathbf{0}$ entonces $p_{S^*}(\mathbf{z})$ no sólo es finita sino positiva y $p_{S^*}(\mathbf{z})^{-1}\mathbf{z}$ no sólo está en S sino en su frontera relativa.

Corolario 1.3.11. *Si* $\mathbf{z} \in aff(S)$*, existe* $\tilde{\mathbf{x}} \in \hat{S}$ *tal que* $p_{S^*}(\mathbf{z}) = \mathbf{z}^T\tilde{\mathbf{x}}$ *y entonces* $p_{S^*}(\mathbf{z}) = p_{\hat{S}}(\mathbf{z})$*. Si* $\mathbf{z}$ *es distinto de cero, entonces* $p_{S^*}(\mathbf{z}) > 0$*.*

Demostración. Por el teorema anterior, si $\mathbf{z} \in$ *aff(S)* la función de soporte de S^* es finita, lo que implica que para todo $\mathbf{z} \in aff(S)$ existe un $\tilde{\mathbf{x}}$ tal que

$$p_{S^*}(\mathbf{z}) = \max_{\mathbf{x}\in S^*} \mathbf{z}^T\mathbf{x} = \mathbf{z}^T\tilde{\mathbf{x}}.$$

Más aún, el teorema 1.3.8 indica que $\mathbf{x} = \mathbf{x}_1 + \mathbf{x}_2$ con $\mathbf{x}_1 \in \hat{S}, \mathbf{x}_2 \in aff(S)^\perp$ y resulta que

$$p_{S^*}(\mathbf{z}) = \max_{\mathbf{x}\in S^*} \mathbf{z}^T\mathbf{x} = \max_{\hat{S}} \mathbf{z}^T\mathbf{x}_1 = p_{\hat{S}}(\mathbf{z}).$$

Por otra parte, como $\mathbf{0} \in ri(S)$ el teorema 1.3.5 indica que $\mathbf{0} \in int(S^*)$. Sea $r > 0$ tal que $B(r, \mathbf{0}) \subset S^*$. Si $\mathbf{z} \neq \mathbf{0}$ entonces existe un $\mathbf{y} \in B(r, \mathbf{0})$ tal que $\mathbf{z}^t\mathbf{y} > 0$, por lo tanto $p_{S^*} > 0$. □

Se puede ir más allá y demostrar que el valor de la función de soporte $p_{S^*}(\mathbf{z})$ está acotado al intervalo $[0, 1)$ siempre y cuando $\mathbf{z}$ esté en el interior relativo de S.

Lema 1.3.12. *Si* $\mathbf{z} \in aff(S)$ *distinto de* $\mathbf{0}$*, entonces* $\frac{1}{p_{S^*}(\mathbf{z})}\mathbf{z} \in S$*.*

Demostración. Por la definición 1.3.9 se tiene que $p_{S^*}(\mathbf{z}) \geq \mathbf{z}^T\mathbf{x}$ para todo $\mathbf{x} \in S^*$. Por ser p_{S^*} finita, $1 \geq \frac{\mathbf{z}^T}{p_{S^*}(\mathbf{z})}\mathbf{x}$ para todo $\mathbf{x} \in S^*$. Esto implica que $\frac{1}{p_{S^*}(\mathbf{z})}\mathbf{z} \in (S^*)^* = S$. Luego, $\frac{1}{p_{S^*}(\mathbf{z})}\mathbf{z} \in S$. □

Teorema 1.3.13. *Si* $\mathbf{z} \in aff(S)$ *distinto de* $\mathbf{0}$*, entonces* $\frac{1}{p_{S^*}(\mathbf{z})}\mathbf{z} \in rb(S)$*.*

Demostración. Supóngase $\frac{1}{p_{S^*}(\mathbf{z})}\mathbf{z} \in ri(S)$, entonces existe $r > 0$ tal que $\overline{B(\frac{1}{p_{S^*}(\mathbf{z})}\mathbf{z}, r)} \subseteq S$. Sea $\mathbf{y} = \frac{1}{p_{S^*}(\mathbf{z})}\mathbf{z} + r\frac{z}{||z||}$ un elemento en dicha bola. Por el lema 1.3.12 se tiene que $\frac{1}{p_{S^*}(\mathbf{z})}\mathbf{z} \in S$ y por el corolario 1.3.11 se sabe que existe $\tilde{\mathbf{x}} \in \hat{S}$ tal que $p_{S^*}(\mathbf{z}) = \mathbf{z}^T\tilde{\mathbf{x}}$, así

$$\begin{aligned} \mathbf{y}^T\tilde{\mathbf{x}} &= \tfrac{1}{p_{S^*}(\mathbf{z})}\mathbf{z}^T\tilde{\mathbf{x}} + \mathbf{z}^T\tilde{\mathbf{x}}(\tfrac{r}{||\mathbf{z}||}) \\ &= 1 + p_{S^*}(\mathbf{z})(\tfrac{r}{||\mathbf{z}||}), \end{aligned}$$

por lo tanto $\mathbf{y}^T\tilde{\mathbf{x}} > 1$. Lo que implica que $\mathbf{y} \notin (S^*)^* = S$, contradiciendo las hipótesis. Se sigue que $\frac{1}{p_{S^*}(\mathbf{z})}\mathbf{z} \in rb(S)$. □

En la figura 1.7 se ilustra el teorema 1.3.13. Nótese que la magnitud del vector $\frac{1}{p_{S^*}(\mathbf{z})}\mathbf{z}$ es una fracción del vector $\mathbf{z}$, por lo que $\frac{1}{p_{S^*}(\mathbf{z})} \leq 1$ y así $p_{S^*}(\mathbf{z}) > 1$. Si $\mathbf{z}$ perteneciera a $ri(S)$ entonces para que el vector $\frac{1}{p_{S^*}(\mathbf{z})}\mathbf{z}$ pertenezca a la frontera relativa de S se tiene que satisfacer que $\frac{1}{p_{S^*}(\mathbf{z})} > 1$ o bien que $p_{S^*}(\mathbf{z}) \leq 1$. Lo anterior se demuestra de manera formal a continuación.

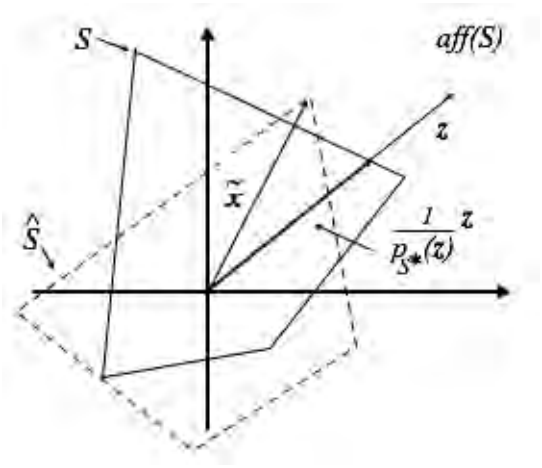


Figura 1.7: Ilustración del teorema 1.3.13.

Lema 1.3.14. *Si* $\mathbf{x}_0 \in ri(S)$ *y* $\mathbf{x}_1 \in S$*, entonces* $(1-\lambda)\mathbf{x}_0 + \lambda\mathbf{x}_1 \in ri(S)$ *para todo* $\lambda \in [0,1)$.

Demostración. Supongamos que para algún $\lambda \in [0,1)$ el punto $\mathbf{z} = (1-\lambda)\mathbf{x}_0 + \lambda\mathbf{x}_1 \notin ri(S)$. Esto implica que para todo $r > 0$ resulta que $B(\mathbf{z}, r) \cap aff(S) \nsubseteq S$, es decir $\mathbf{z} \in rb(S)$, sin embargo, por convexidad esto implicaría que $\mathbf{x}_0 \in rb(S)$. Por lo tanto, $\mathbf{z} \in ri(S)$ para todo $\lambda \in [0,1)$. □

Teorema 1.3.15. *Sea* $\mathbf{z} \in aff(S)$*, entonces* $\mathbf{z} \in ri(S)$ *si y solo si* $p_{S^*}(\mathbf{z}) < 1$.

Demostración. Trivialmente si $\mathbf{z} = \mathbf{0}$ entonces $p_{S^*}(\mathbf{0}) = 0 < 1$. Para demostrar la suficiencia considérese que se puede escribir a $\mathbf{z}$ como una combinación convexa de la forma

$$\mathbf{z} = (1 - p_{S^*}(\mathbf{z}))\mathbf{0} + p_{S^*}(\mathbf{z})\frac{1}{p_{S^*}(\mathbf{z})}\mathbf{z}. \tag{1.9}$$

Puesto que por hipótesis $\mathbf{0} \in ri(S)$ y $p_{S^*}(z) < 1$ el lemma anterior concluye que $\mathbf{z} \in ri(S)$. Para demostrar la necesidad, supóngase que $p_{S^*}(\mathbf{z}) \geq 1$ y obsérvese la ecuación (1.9). Por el lema 1.3.14 y el teorema 1.3.13 resulta que $\mathbf{z}$ está en la frontera o fuera de S y así $\mathbf{z} \notin ri(S)$. □

Capítulo 2

Programación lineal en dimensiones bajas

En este capítulo se presentan los conceptos básicos de programación lineal (PL) y un análisis riguroso sobre el algoritmo en tiempo lineal de Megiddo para problemas de PL en dimensiones bajas. Este capítulo se divide en dos secciones: en la sección 2.1 se introducen algunas definiciones y resultados necesarios de PL; en la sección 2.2 se trata de manera detallada el algoritmo lineal de Megiddo para problemas de PL de dimensiones dos, tres y la generalización del mismo a una dimensión fija arbitraria. Esta última sección resulta extensa por la cantidad de material necesario para cada algoritmo: primero se presenta una introducción al algoritmo, planteando de manera general la idea de éste y los principios en los que se basa; a continuación, se demuestran resultados importantes utilizados por Megiddo para la transformación de un problema PL en forma canónica a un problema de optimización de menor dimensión; posteriormente se presenta el algoritmo lineal de Megiddo para problemas de PL en cada dimensión; finalmente se analiza la complejidad computacional del algoritmo.

2.1. Conceptos básicos

Un problema de PL consiste en maximizar o minimizar una función lineal sujeta a un conjunto de m restricciones lineales y n varibles. El problema de *minimización en forma canónica* [42] se puede como

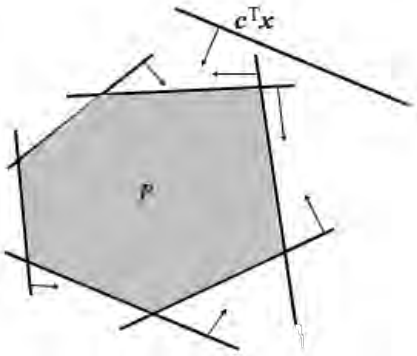
$$\text{mín}\, \mathbf{c}^T\mathbf{x}, \tag{2.1}$$

$$\text{sujeto a}$$

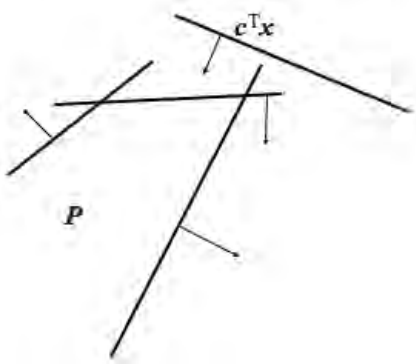
$$\mathbf{A}\mathbf{x} \geq \mathbf{b} \tag{2.2}$$

$$\mathbf{x} \geq \mathbf{0} \tag{2.3}$$

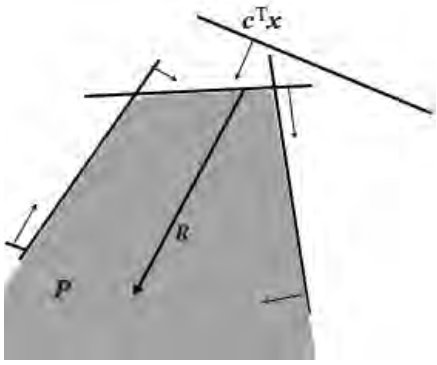
donde $\mathbf{x}, \mathbf{c} \in \mathbb{R}^n$, $\mathbf{b} \in \mathbb{R}^m$ y $\mathbf{A} \in \mathbb{R}^{m\times n}$. A la expresión $z = \mathbf{c}^T\mathbf{x}$ se le llama *función objetivo*. El objeto definido por la intersección de las restricciones en (2.2) y (2.3) es



(a) Problema factible acotado donde la solución óptima es alcanzada en un segmento de recta de P.



(b) Problema no factible.



(c) Problema no acotado.

Figura 2.1: Clasificación de problemas de PL.

un poliedro llamado *espacio de soluciones* y se denota por P [11]. A los renglones de $\mathbf{A}$ y $\mathbf{b}$ se hará referencia como $\mathbf{A}_j, b_j$ con $j \in [m]$.

Se dice que un problema de PL es *factible* si P es no vacío; caso contrario, se dice que el problema es *no factible*. A las condiciones (2.2) y (2.3) se les llama *condiciones de factibilidad* y un vector $\mathbf{x}$ es una *solución factible* si satisface estas condiciones. Un problema de minimización es *no acotado* si existe un rayo R dentro de P en la misma dirección que c, tal que la función objetivo decrece arbitrariamente a lo largo de éste, si no es así el problema resulta ser *acotado*. A los vectores que satisfacen las condiciones de factibilidad y minimizan la función objetivo se le llama *solución óptima*; al valor de la función objetivo evaluada en dicho vector se le llama *valor óptimo* [16].

Resumiendo, un problema de PL se puede clasificar según sus propiedades en tres casos distintos:

1. El problema es acotado y factible, así que tiene una solución óptima; en este caso el poliedro de soluciones es un politopo. En la figura 2.1a se ilus-

tra un problema de PL acotado factible el cual tiene como solución óptima el segmento de recta punteado, ya que es paralelo a la función objetivo.

2. El poliedro de soluciones P es vacío por lo que el problema es no factible (figura 2.1b).

3. El problema tiene una solución factible y es no acotado, por lo tanto no tiene una solución óptima (figura 2.1c).

Es importante observar que en el primer caso es posible que existan múltiples soluciones óptimas [42]. Más aun, la búsqueda de una solución óptima se puede restringir a un subconjunto de puntos de P, como se verá más adelante.

Una desigualdad $\mathbf{A}_j\mathbf{x} \geq b_j$ con $j \in [m]$ es *activa* en un punto $\mathbf{x}_0$ si $\mathbf{A}_j\mathbf{x}_0 = \mathbf{b}_j$ y es una *desigualdad superflua* si el hiperplano que define no determina la solución óptima del problema. Una restricción es *redundante* si no es necesaria para definir el polidero P. Un *vértice* de P es una solución factible donde al menos n restricciones son activas; el problema de PL resulta ser *degenerado* si al menos un vértice se define por más de n restricciones válidas activas; en caso de que cada vértice sea definido por exactamente n restricciones el problema es *no degenerado*.

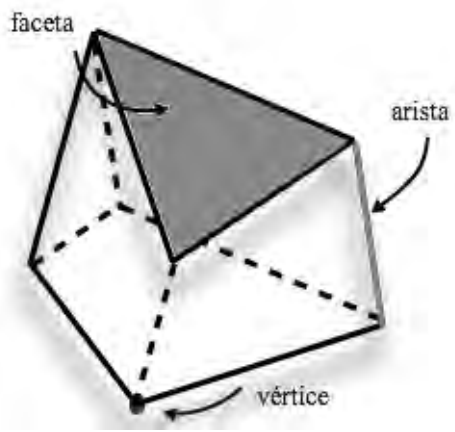


Figura 2.2: Ejemplo de un politopo de dimensión tres y todas sus caras.

Una *cara* F_J de un poliedro P se define como

$$F_J = \{\mathbf{x} \in P : \mathbf{A}_j\mathbf{x} = b_j, \quad j \in J\},$$

donde $J \subseteq [m]$. Nótese que las caras son poliedros en sí y $F_J \subseteq P$ para cualquier subconjunto J. Las caras de dimensión cero son vértices, las caras de dimensión 1 son aristas y las caras de dimensión $dim(P) - 1$ son llamadas *facetas* (figura 2.2). Las *caras triviales* de un poliedro son el vacío y el poliedro. Una forma de definir la frontera de un poliedro P es

$$b(P) = \cup_{j \in m^*} F_j, \tag{2.4}$$

donde F_j con $j \in m^*$ son las facetas de P. Nótese que la frontera de un poliedro no está exenta de tener caras de dimensión menor a $dim(P) - 1$ como vértices, aristas, etcétera. Los siguientes teoremas determinan la estructura geométrica de P y su relación con las soluciones óptimas.

Teorema 2.1.1. *Dado un problema de PL definido por (2.1), (2.2) y (2.3), el espacio de soluciones P es convexo.*

Demostración. Sean $\mathbf{x}_1, \mathbf{x}_2 \in P$ y $\lambda \in [0,1]$. Si $\mathbf{x} = (1-\lambda)\mathbf{x}_1 + \lambda\mathbf{x}_2$, por propiedades de álgebra lineal se tienen las siguientes equivalencias

$$\mathbf{A}\mathbf{x} = \mathbf{A}[(1-\lambda)\mathbf{x}_1 + \lambda\mathbf{x}_2] = (1-\lambda)\mathbf{A}\mathbf{x}_1 + \lambda\mathbf{A}\mathbf{x}_2 \leq (1-\lambda)\mathbf{b} + \lambda\mathbf{b} = \mathbf{b}.$$

Por lo tanto $\mathbf{x} \in P$. Se sigue que P es convexo. □

Teorema 2.1.2. *Dado un problema de PL definido por (2.1), (2.2) y (2.3). Si P es el politopo asociado al problema y $\mathbf{x}^*$ es una solución óptima del problema, entonces $\mathbf{x}^*$ pertenece a una cara F de P no trivial.*

Demostración. Supóngase que $\mathbf{x}^* \in int(P)$. Entonces existe $r > 0$ tal que $B(\mathbf{x}^*, r) \subset P$. Si $\mathbf{y} = \mathbf{x}^* - \frac{r}{2}\frac{\mathbf{c}}{||\mathbf{c}||}$ con c el vector asociado a la función objetivo, entonces

$$||\mathbf{y} - \mathbf{x}^*|| = \frac{r}{2}\frac{||\mathbf{c}||}{||\mathbf{c}||} = \frac{r}{2},$$

por lo que $\mathbf{y} \in B(\mathbf{x}^*, r)$. Sin embargo se tiene que

$$\mathbf{c}^T\mathbf{y} = \mathbf{c}^T\mathbf{x}^* - \frac{r}{2}\frac{\mathbf{c}^T\mathbf{c}}{||\mathbf{c}||} = \mathbf{c}^T\mathbf{x}^* - \frac{r}{2}||\mathbf{c}|| < \mathbf{c}^T\mathbf{x}^*,$$

lo que implica que $\mathbf{x}^*$ no es una solución óptima. Luego, $\mathbf{x}^* \in b(P)$, y por la expresión en (2.4) resulta que $\mathbf{x}^*$ está en una faceta de P. □

2.2. Algoritmo de Megiddo

Una vez presentados los criterios de existencia de una solución de un problema de PL, se puede analizar qué sucede para problemas de PL de dimensión fija. Megiddo inició el dearrollo de su algoritmo después de que en 1979 Kachiyan [24] dio a conocer el primer algoritmo polinomial de programación lineal. Megiddo argumentaba que éste no era un algoritmo "genuinamente" polinomial ya que dependía del tamaño de los coeficientes. Por ello, su objetivo era desarrollar un algoritmo que requiriera un número polinomial de operaciones en términos del tamaño del problema, presentando en 1983 el artículo [30] donde propone un algoritmo lineal para la resolución de problemas de PL de dimensiones dos y tres utilizando herramientas básicas de geometría.

El algoritmo de Megiddo tuvo poca repercusión en la comunidad de programación matemática ya que en 1984 Karmarkar [25] publicó su algoritmo polinomial, el cual, a diferencia del algoritmo elipsoidal era eficiente en la práctica. Sin embargo, en el ámbito de la geometría computacional tuvo mayor repercusión como lo muestran los artículos de Hurtado et al [21], de Clarkson [9] y Dyer en [14, 15, 17]. El enfoque de Megiddo ha sido aplicado en problemas como el centro ponderado de un árbol, el círculo más pequeño que contenga n puntos [30], el elipsoide de menor volumen [21], entre otros.

Para facilitar la descripción general del algoritmo, transfórmese el problema de PL en $\mathbb{R}^n$ definido por las ecuaciones (2.1), (2.2), (2.3) a la forma

$$\begin{aligned} &\text{mín}\, x_n \\ &\text{sujeto a} \\ x_n \geq \tilde{a}_1 x_1 + \tilde{a}_{i2} x_2 + \ldots + \tilde{a}_{in-1} x_{n-1} + b_i \quad &, i \in I_1, \\ x_n \leq \tilde{a}_{i1} x_1 + \tilde{a}_{i2} x_2 + \ldots + \tilde{a}_{in-1} x_{n-1} + b_i \quad &, i \in I_2, \\ 0 \geq \tilde{a}_{i1} x_1 + \tilde{a}_{i2} x_2 + \ldots + \tilde{a}_{in-1} x_{n-1} + b_i \quad &, i \in I_3, \end{aligned}$$

donde $|I_1|+|I_2|+|I_3| = n$ representan conjuntos de índices de restricciones. Esta formulación se obtiene mediante una transformación lineal (un cambio de variables) y se le llamará la *formulación de Megiddo* para un problema de PL en $\mathbb{R}^n$.

Considérense una pareja de restricciones *i,j* del conjunto I_k con $k = 1,2,3$. Los hiperplanos correspondientes a cada pareja de restricciones pueden ser paralelos, lo que implica que uno de ellos es redundante y por lo tanto puede ser eliminado, o bien se pueden intersectar en un hiperplano H_{ij} de dimensión $n-2$. Nótese que, H_{ij} divide a $\mathbb{R}^{n-1}$ en dos semiespacios y cada uno de ellos está dominado por alguna de las restricciones que definen a H_{ij}. Así, si se sabe en qué semiespacio yace la solución óptima, se sabe cual de los hiperplanos es redundante y puede ser eliminado.

Saber la ubicación de un punto con respecto a un hiperplano no es una tarea sencilla, pero Megiddo implementa una *búsqueda multidimensional* para resolver este problema (llamada la prueba de un hiperplano). Esta prueba no sólo determina algunas restricciones superfluas, también indica qué hiperplanos son los que conviene probar.

El algoritmo realiza a lo más $n-1$ pruebas para identificar la ubicación de la solución óptima respecto a todo el conjunto de restricciones. Para esto genera hiperplanos de la forma $x_i = \alpha_i$ con $i = 1,\ldots,n-1$, para determinar cual de los semiespacios que genera es de interés. Esto se lleva a cabo resolviendo de manera recursiva a lo más tres problemas de PL (también llamados problemas auxiliares) de dimensión $n-1$ por cada variable. El primer problema auxiliar consiste en resolver el problema sobre el hiperplano $x_i = \alpha_i$; obteniendo una solución sobre él que será la solución inicial de la iteración. Los otros dos problemas trabajan solamente con los hiperplanos activos en la solución inicial, ya que éstos

son suficientes para describir el comportamiento del problema alrededor de dicho punto. Estos últimos dos problemas determinan si la solución óptima está en el semiplano superior o inferior generados por el hiperplano a prueba.

Para cada subproblema construido por el algoritmo, el algoritmo calcula la mediana de un conjunto de elementos, de tal manera que a lo más la mitad de los elementos de dicho conjunto sean necesarios para encontrar la solución óptima. Lo anterior permite que en cada paso una fracción fija de las restricciones superfluas sean eliminadas. A la técnica de realizar una cantidad de trabajo en cada paso para reducir el tamaño del problema por una fracción constante se le llama *prune and search* (podar y buscar).

El algoritmo termina cuando el número de restricciones es suficientemente pequeño para poder resolver el problema de PL de manera directa (Megiddo propone que 4 es un número considerable para poder resolver el problema de manera directa), o bien cuando la solución inicial de alguna iteración coincida con la solución óptima. Cada subproblema del algoritmo es cuidadosamente diseñado para llevarse a cabo en un tiempo proporcional al número de restricciones presentes en cada iteración.

El hecho de que cada subproblema del algoritmo se lleve a cabo en un tiempo proporcional al número de restricciones presentes y éste disminuya en cada iteración, reafirma la complejidad lineal del algoritmo en función del número de restricciones.
En las siguientes secciones se discute a mayor profundidad el algoritmo para dimensiones dos y tres, y se discute su generalización a dimensiones fijas mayores. También se analiza su complejidad computacional.

2.2.1. Algoritmo de Megiddo en $\mathbb{R}^2$

En particular, para problemas de PL en el plano, la primera parte del algoritmo consiste en clasificar las restricciones y emparejarlas según el subconjunto al que pertenecen; claramente si para alguna pareja las rectas correspondientes a las restricciones tienen la misma pendiente, una de las restricciones será eliminada. De las parejas que no presenten esta propiedad se determinará su intersección, obteniendo una sucesión de puntos $\alpha_1, \alpha_2, ...\alpha_k$, de los cuales se seleccionará la mediana, α_m. Posteriormente, se aplica la prueba al hiperplano $x_1 = \alpha_m$, la cual consiste en comparar la cota superior con la cota inferior de las restricciones activas en α_m y elegir el menor valor para obtener la solución inicial. Los problemas auxiliares de PL de dimensión menor consisten en estudiar las derivadas laterales de una función auxiliar en $x_1 = \alpha_m$, para determinar la ubicación de mínimo global. A continuación, se expanden estas nociones.

El problema de PL en el plano puede expresarse como

$$\begin{aligned} \text{mín}\, c_1 x_1 + c_2 x_2 \\ \text{sujeto a} \\ a_{i1} x_1 + a_{i2} x_2 \geq b_i \quad i \in [m]. \end{aligned} \tag{2.5}$$

Por propósitos prácticos, la formulación anterior se puede transformar en una equivalente en tiempo lineal.

Lema 2.2.1. *El problema de PL definido en (2.5) en $\mathbb{R}^2$ es equivalente al problema*

$$\begin{aligned} \text{mín}\, y \\ \textit{sujeto a} \\ y \geq \tilde{a}_i x + \tilde{b}_i, \quad i \in I_1, \\ y \leq \tilde{a}_i x + \tilde{b}_i, \quad i \in I_2, \\ d_1 \leq x \leq d_2, \end{aligned} \tag{2.6}$$

donde $|I_1| + |I_2| \leq m$.

Demostración. Sin pérdida de generalidad, supóngase que $c_2 \neq 0$ (de no ser así se puede realizar el mismo procedimiento con c_1). Utilizando el cambio de variable $x = x_1$ y $y = c_1 x_1 + c_2 x_2$, el sistema (2.5) se transforma en

$$\begin{aligned} \text{mín}\, y \\ \left(a_{i1} - \frac{c_1}{c_2} a_{i2}\right) x + \frac{a_{i2}}{c_2} y \geq b_i \quad i \in [m]. \end{aligned}$$

Si para algún índice $i \in [m]$ la constante $\frac{a_{i2}}{c_2} > 0$, entonces al multiplicar la desigualdad (2.7) por $\frac{c_2}{a_{i2}}$ resulta que

$$y \geq \frac{c_2}{a_{i2}} \left[\left(\frac{c_1}{c_2} a_{i2} - a_{i1} \right) x + b_i \right]. \tag{2.7}$$

Análogamente, si para algún índice $i \in [m]$se tiene que $\frac{a_{i2}}{c_2} < 0$, al multiplicar la desigualdad (2.7) por $\frac{c_2}{a_{i2}}$ se obtiene

$$y \leq \frac{c_2}{a_{i2}} \left[\left(\frac{c_1}{c_2} a_{i2} - a_{i1} \right) x + b_i \right]. \tag{2.8}$$

Finalmente, si $a_{i2} = 0$ para algún $i \in [m]$ la desigualdad (2.7) es de la forma

$$a_{i1} x \geq b_i. \tag{2.9}$$

Para este caso se tiene alguna de las dos opciones siguientes: si $a_{i1} > 0$ entonces $x \geq \frac{b_i}{a_{i1}}$; si $a_{i1} < 0$ entonces $x \leq \frac{b_i}{a_{i1}}$. Se puede entonces definir las constantes $d_1 = \text{máx}\{\frac{b_i}{a_{i1}} \,:\, a_{i2} = 0,\ a_{i1} > 0\}$ y $d_2 = \{\frac{b_i}{a_{i1}} \,:\, a_{i2} = 0,\ a_{i1} < 0\}$ de tal forma que $x \in [d_1, d_2]$. Ahora, definiendo las siguientes constantes

$$\tilde{a}_i = c_1 - \frac{c_2 a_{i1}}{a_{i2}}, \quad \tilde{b}_i = b_i \frac{c_2}{a_{i2}},$$

las desigualdades (2.7) y (2.8) se transforman en las restricciones

$$\begin{aligned} y &\geq \tilde{a}_i x + \tilde{b}_i, \quad i \in I_1 \\ y &\leq \tilde{a}_i x + \tilde{b}_i, \quad i \in I_2 \end{aligned}$$

donde I_1, I_2 representan los conjuntos de todos los índices para los que se satisface $\frac{\alpha_{i2}}{c_2} > 0$ y $\frac{\alpha_{i2}}{c_2} < 0$ respectivamente, y $|I_1| + |I_2| \leq m$. Ya que el cambio de variables propuesto es biyectivo y continuo para c_1, c_2 fijos, la transformación tiene inversa y las formulaciones son equivalentes. A la formulación en (2.6) se le llamará *formulación de Megiddo para el plano*. Puesto que de ahora en adelante se utilizará esta formulación, se omitirán las tildes de las constantes que definen las restricciones. □

Una vez reformulado el problema de PL en el plano, se busca transformar nuevamente el problema para llegar a un problema unidimensional. Para esto defínanse las funciones

$$\begin{aligned} f_1(x) &= \text{máx}\{a_i x + b_i \,:\, i \in I_1\}, \\ f_2(x) &= \text{mín}\{a_i x + b_i \,:\, i \in I_2\}. \end{aligned}$$

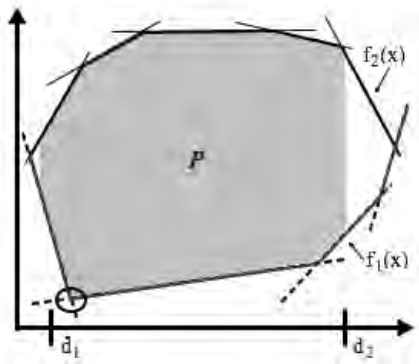
Nótese que f_1 y f_2 son convexa y cóncava respectivamente, ambas funciones son lineales por partes. Basándose en la formulación de Megiddo para el plano, f_1 y f_2 constituyen una cota inferior y superior de y respectivamente. Así, el problema de PL en (2.6) se puede reescribir como

$$\begin{aligned} &\text{mín}\, y \\ &\text{sujeto a} \\ &f_1(x) \leq y \leq f_2(x), && (2.10) \\ &d_1 \leq x \leq d_2. && (2.11) \end{aligned}$$

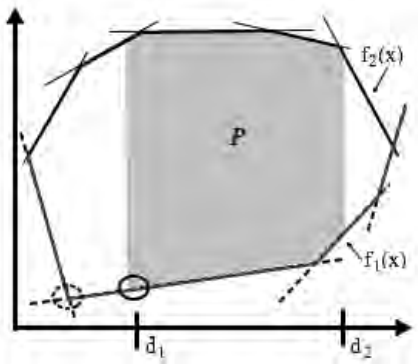
Al encontrar un valor de x factible el valor de y queda acotado. Puesto que se quiere minimizar la variable y, su valor mínimo será el valor mínimo que pueda tomar la función $f_1(x)$ sujeto a (2.10) y (2.11). Por lo tanto, el problema de PL en el plano se puede expresar como un problema unidimensional de la forma

$$\begin{aligned} &\text{mín}\ f_1(x) \\ &\text{sujeto a} \\ &f_1(x) \leq f_2(x), && (2.12) \\ &d_1 \leq x \leq d_2, && (2.13) \end{aligned}$$

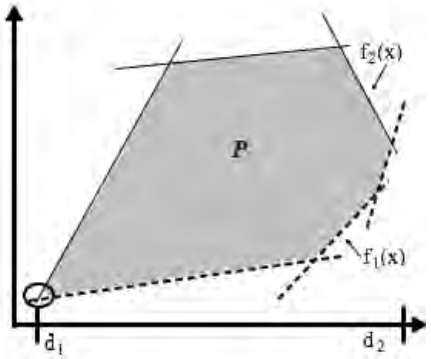
donde un valor de x es factible si satisface las restricciones (2.12) y (2.13). En esta formulación las restricciones (2.12) y (2.13) determinan a P. A esta formulación se le llamará la *formulación unidimensional* de un problema de PL en el plano.



(a) El valor óptimo del problema está determinado por $f_1(x)$.



(b) El valor óptimo del problema no es el mínimo de $f_1(x)$.



(c) El valor óptimo del problema se define por la intersección de $f_1(x)$ y $f_2(x)$.

Figura 2.3: Ejemplos de distintas soluciones para el problema unidimensional. El área gris indica el poliedro de soluciones.

Una ilustración de esta formulación se presenta en la figura 2.3. En la figura 2.3a la solución óptima se encuentra en P y satisface la condición de factibilidad de la forma $f_1(x) \leq f_2(x)$. En la figura 2.3b el valor mínimo de la función $f_1(x)$, denotado por un círculo punteado, no es el valor óptimo del problema de PL (marcado por un círculo continuo), ya que no se encuentra dentro de P. En la imagen 2.3c la solución óptima se encuentra en el punto de intersección de las dos funciones satisfaciendo la factibilidad de la forma $f_1(x) = f_2(x)$.

En adelante, se utilizarán las formulaciones de Megiddo y unidimensional de un problema en el plano. Una vez realizadas las transformaciones pertinentes y la definición de las funciones auxiliares se pasa a la primera etapa del algoritmo.

2.2.1.1 Eliminación de restricciones redundantes

En primera instancia el algoritmo reúne los elementos de I_1 en parejas de disjuntas y lo mismo sucede con los elementos de I_2. Nótese que a lo más, un elemento de cada conjunto de índices puede quedar sin pareja. De cada pareja de índices se puede eliminar una restricción u obtener un punto de intersección. Sea $i, j \in I_1$ una pareja de índices. Si $a_i = a_j$ entonces una de las restric-

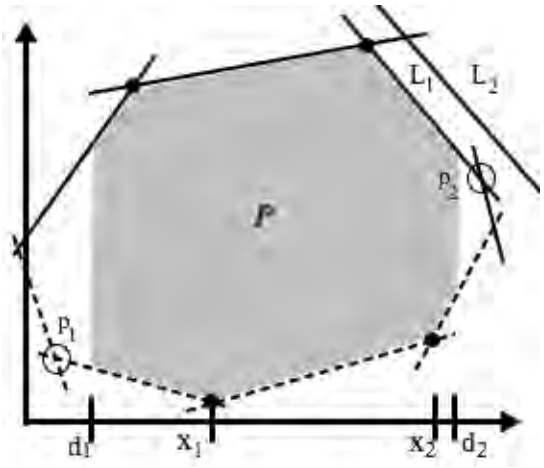


Figura 2.4: Eliminación de restricciones redundantes e identificación de puntos de intersección x_{ij}.

ciones $y \geq a_i x + b_i$ ó $y \geq a_j x + b_j$ es redundante ya que representan el mismo semiespacio trasladado sobre la vertical. Puesto que las restricciones correspondientes a I_1 definen semiespacios superiores, la restricción redundante es la que tenga menor ordenada al origen. Ahora, si $a_i \neq a_j$, las restricciones correspondientes se intersectan en el punto

$$x_{ij} = \frac{b_i - b_j}{a_j - a_i}.$$

Si x_{ij} no estuviera en el intervalo de definición del problema $[d_1, d_2]$, una de las restricciones que definen el punto es redundante y puede ser descartada: si $x_{ij} \leq d_1$ entonces se elimina la restricción que tenga pendiente menor; si $d_2 \leq x_{ij}$ se elimina la restricción con pendiente mayor.

Una argumentación análoga se sigue para las parejas de índices de I_2. Si $a_i = a_j$ se descarta la restricción que tenga componente ordenada al origen mayor. Si $a_i \neq a_j$ se obtiene un punto de intersección x_{ij}. Si $x_{ij} \notin [d_1, d_2]$ entonces $x_{ij} \leq d_1$ por lo que se remueve la restricción con pendiente mayor, o bien $d_2 \leq x_{ij}$ con lo que se descarta la restricción con pendiente menor. Nótese que, tanto para I_1 como I_2 el hecho de que $x_{ij} \notin [d_1, d_2]$ implica que no es un valor factible y puede ser eliminado.

En la figura 2.4 se presenta un ejemplo de este proceso: las rectas punteadas son las correspondientes a las desigualdades en I_1 y las rectas continuas las correspondientes a I_2. Las rectas L_1 y L_2 son paralelas y corresponden a una pareja de restricciones en I_2; puesto que el poliedro de soluciones está definido por la intersección de los semiespacios determinados por las restricciones en I_1 y I_2, la recta L_2 (cuya ordenada al origen es mayor que la de L_1) es innecesaria para definir P, y por lo mismo, para obtener la solución óptima. Por otra parte, los puntos de intersección marcados de negro son soluciones factibles, mientras que los marcados por círculos son soluciones no factibles. El punto p_1 corresponde a la intersección de dos restricciones en I_1 y, por ser no factible, una de las restricciones que lo definen es redundante para definir el poliedro de soluciones, y así, la solución óptima. Nótese que, la restricción con pendiente menor, es la que resulta redundante. De manera semejante, el punto p_2 corresponde a la intersección de dos restricciones pertenecientes a I_2 y no es una solución factible; se observa que la restricción con pendiente menor que define este punto, resulta redundante.

Finalizado este paso, hay a lo más $\lfloor \frac{m}{2} \rfloor$ puntos x_{ij} pertenecientes al intervalo de soluciones $[d_1, d_2]$; los puntos fuera de él son redundantes y por consiguiente una de las restricciones que definen a estos puntos es innecesaria, terminando con a lo más m restricciones al final de esta instancia.

En la figura 2.5 se presenta en forma de diagrama un bosquejo sobre el proceso de eliminación de restricciones redundantes y creación de puntos de intersección.

2.2.1.2 Prueba para un punto en una recta

Aquí se muestra la prueba medular del algoritmo utilizada para resolver el problema en su forma unidimensional. Ésta determina la existencia de un valor óptimo o la no factibilidad del problema. En general, supóngase x' tal que $d_1 \leq x' \leq d_2$. Si x' es factible,el algoritmo determina si es también óptimo, y si no, precisa si x' es menor o mayor que el valor óptimo. Si x' no es factible, el algoritmo determina si existe algún valor factible menor o mayor que x'; si no existe tal valor, entonces el algoritmo decreta que el problema no es factible. Para realizar este análisis, se definen las derivadas laterales de $f_1(x)$ y $f_2(x)$ en el punto x':

$$\begin{aligned}
df_1^- &= \text{mín}\{a_i \,:\, i \in I_1,\ a_i x' + b_i = f_1(x')\},\\
df_1^+ &= \text{máx}\{a_i \,:\, i \in I_1,\ a_i x' + b_i = f_1(x')\},\\
df_2^- &= \text{máx}\{a_i \,:\, i \in I_2,\ a_i x' + b_i = f_2(x')\},\\
df_2^+ &= \text{mín}\{a_i \,:\, i \in I_2,\ a_i x' + b_i = f_2(x')\},
\end{aligned}$$

donde df_1^- y df_2^- son las derivadas por la izquierda y df_1^+, df_2^+ las derivadas por la derecha de $f_1(x)$ y $f_2(x)$ respectivamente.

Asumiendo que x' satisface la factibilidad de la forma $f_1(x') < f_2(x')$, el análisis resulta sencillo, ya que sólo se debe estudiar el comportamiento de las de-

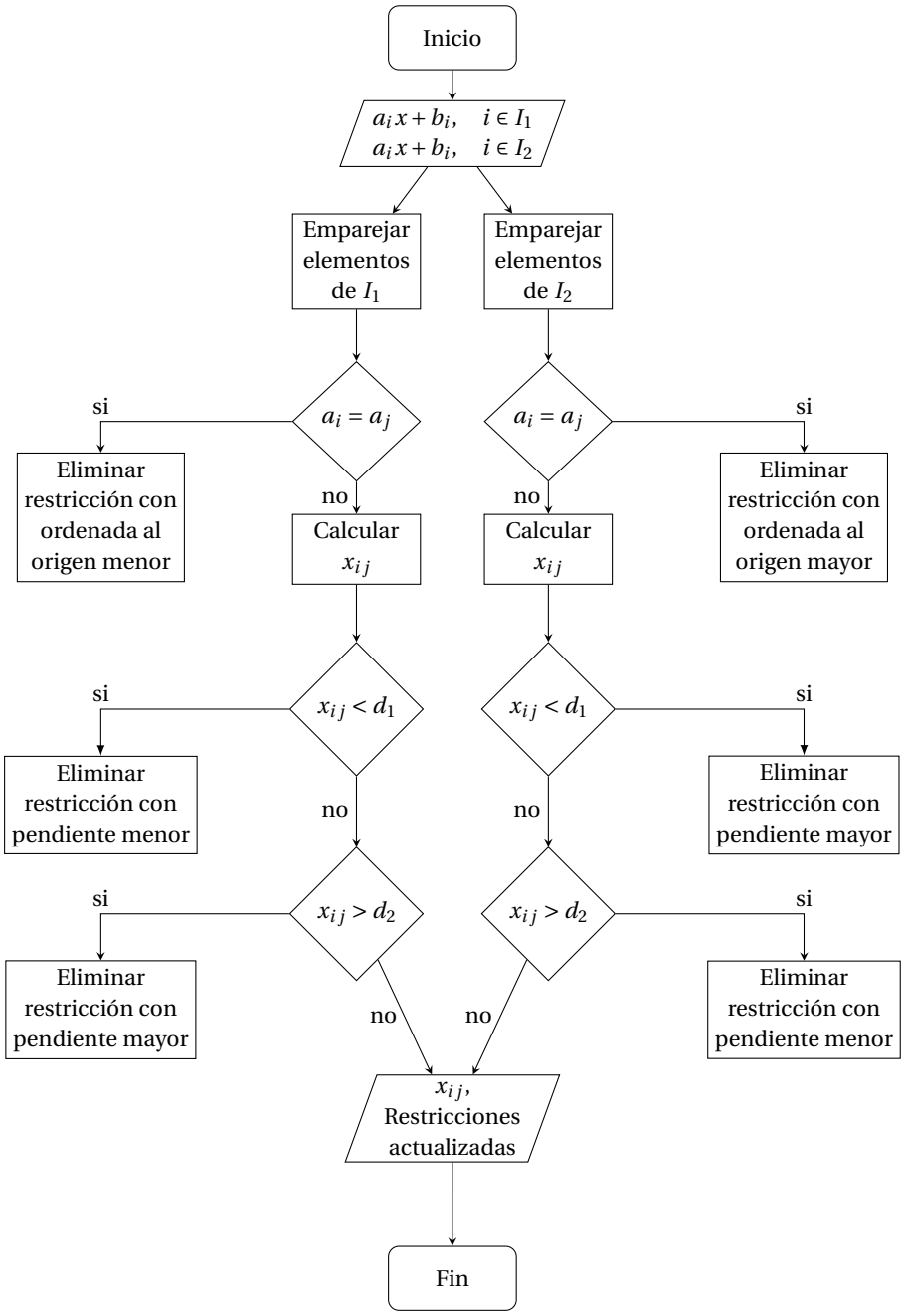


Figura 2.5: Diagrama de flujo para la eliminación de restricciones redundantes y creación de puntos.

rivadas laterales de $f_1(x)$. Entonces:

- Si $df_1^- > 0$ entonces f_1 es creciente en x' y la solución x^* debe ser menor que x'.

- Si $df_1^+ < 0$ la función resulta ser decreciente en x' y el valor óptimo del problema debe ser mayor que x'.
- En caso contrario, resulta que df_1^- es negativa y df_1^+ positiva, por lo que se concluye que $x' = x^*$.

En la figura 2.4 se tienen los puntos x_1 y x_2 que satisfacen la factibilidad de la forma $f_1(x') < f_2(x')$. En el punto x_2 la derivada izquierda de f_1 es positiva por lo que el óptimo debe ser menor. El punto x_1 resulta ser el óptimo ya que la derivada por la izquierda de f_1 es negativa y la derivada por la derecha de f_1 es positiva.

Si x' satisface la factibilidad de la forma $f_1(x') = f_2(x')$, entonces x' es un punto de intersección entre ambas funciones y se presentan nuevamente tres casos importantes. Éstos están ilustrados en la figura 2.6.

- Si $df_1^- > 0$ y $df_1^- \geq df_2^-$ entonces $f_1(x)$ es creciente en x' y x^* debe ser menor que x' (punto (x_2,0) en la figura 2.6).
- Si $df_1^+ < 0$ y $df_1^+ \leq df_2^+$ resulta que $f_1(x)$ es decreciente en x' y x^* debe ser mayor que x' (punto (x_1,0) en la figura 2.6).
- En caso contrario, se tiene que x' es un mínimo de $f_1(x)$ y por lo tanto la solución óptima.

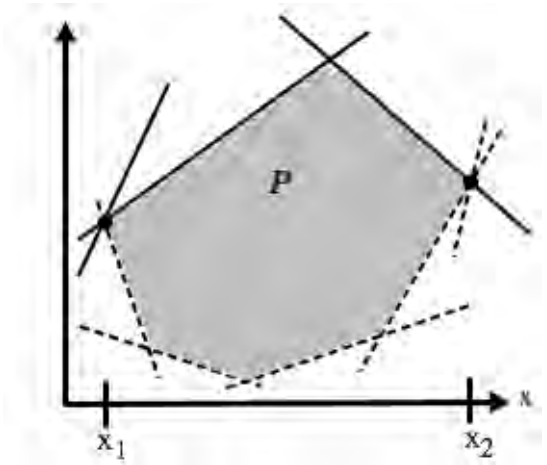


Figura 2.6: Problema en donde la factibilidad se satisface $f_1(x) = f_2(x)$ para los puntos x_1 y x_2.

Ahora, si x' no es factible, es decir, $f_1(x') > f_2(x')$, considérese la función auxiliar

$$f(x) = f_1(x) - f_2(x) \tag{2.14}$$

Por ser la diferencia de una función convexa con una función cóncava, $f(x)$ es convexa y $f(x') > 0$. Esto implica que si existen valores x tal que $f(x) \leq 0$,

entonces están en el mismo intervalo $[d_1, x)$ o $(x, d_2]$. Más aún, todo x tal que $f(x) \leq 0$ es un valor factible para el problema unidimensional. Se presentan entonces tres casos importantes a analizar para el caso de la no factibilidad:

- Si $df_1^- > df_2^-$ implica que la derivada por la izquierda de $f(x)$ en x' es positiva y por consiguiente, que $f(x)$ es creciente en dicho punto. Por lo tanto, un valor factible x solo puede estar a la izquierda de x'. Este caso es equivalente a que $df_1^+ > df_2^+$ (la coordenada x_2 en la figura 2.7 satisface este caso).

- Si $df_2^+ > df_1^+$ entonces la derivada por la derecha de $f(x)$ en x' es negativa y $f(x)$ es decreciente en x'. Se sigue que si existe un valor x tal que $f(x) \leq 0$ entonces debe ser mayor que x'. El análisis de este caso es equivalente al realizado para $df_1^- < df_2^-$ (esto se representa en la figura 2.7 por la coordenada x_1).

- Si $df_1^- - df_2^- \leq 0 \leq df_1^+ - df_2^+$ significa que la derivada por la izquierda de $f(x)$ es negativa y la derivada por la derecha es positiva, lo que implica que x' es un mínimo de $f(x)$ y sin embargo, no es factible. Por lo tanto el problema es no factible.

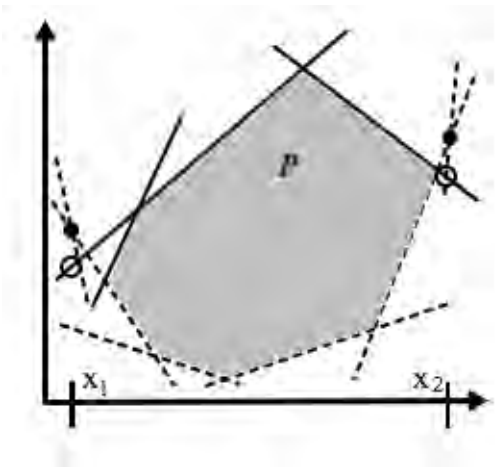


Figura 2.7: Problema en donde la factibilidad se satisface $f_1(x) > f_2(x)$ para algún punto x.

Así, dado un valor x' en un intervalo $[d_1, d_2]$, esta prueba puede determinar en tiempo lineal si el problema es factible o no, si el valor x' es óptimo o en su defecto si el valor óptimo es mayor o menor que x'. Es importante observar que una vez que se ha determinado si x^* es mayor o menor que x', el resto de la computación se continúa en alguno de los intervalos $[d_1, x']$ o $[x', d_2]$ y por lo tanto todo valor que corresponda al intervalo contrario es redundante. Por

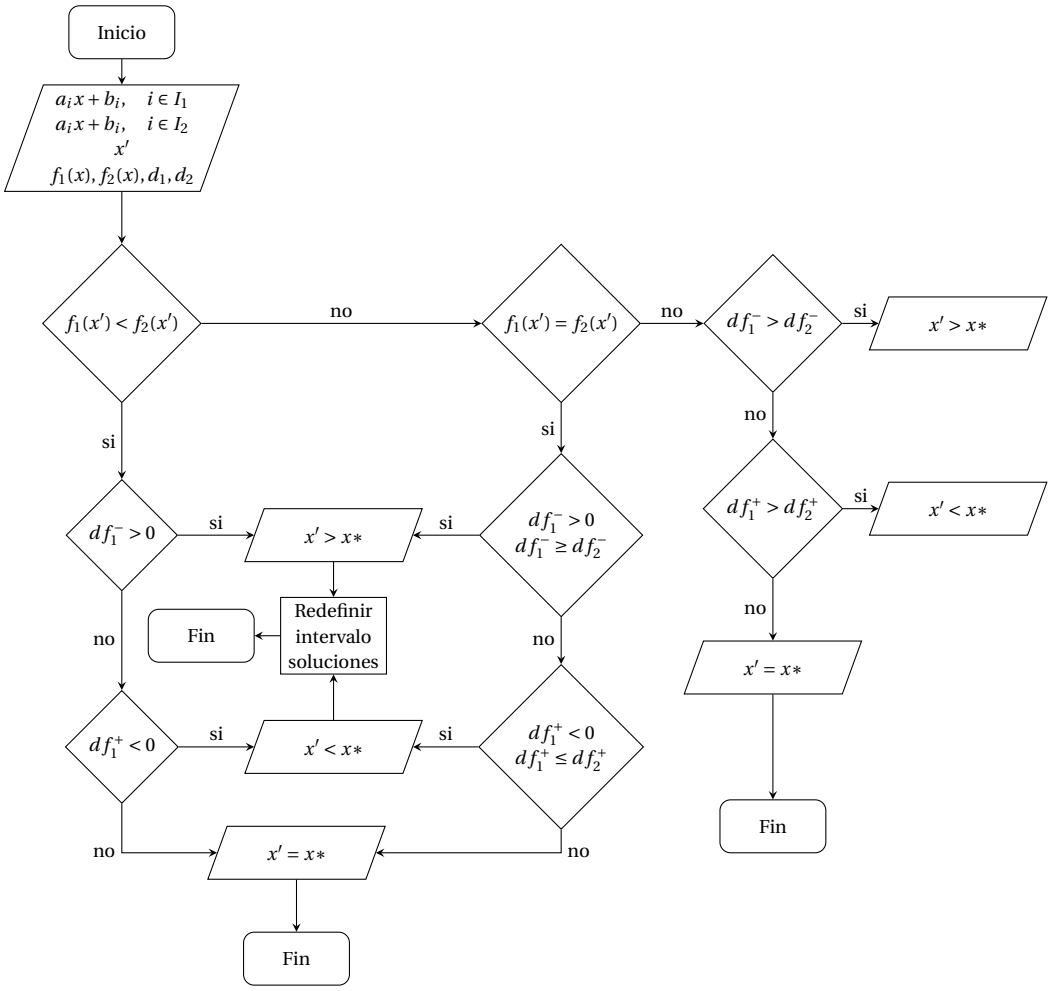


Figura 2.8: Diagrama de flujo para la prueba de un punto en una recta.

consiguiente, de cada punto redundante una de las restricciones que lo definan. puede ser eliminada.

En la figura 2.8 se esboza un diagrama de la prueba para un punto en una recta.

2.2.1.3 Algoritmo de Megiddo para $\mathbb{R}^2$

En esta sección se describe el algoritmo de Megiddo para el plano. Primero se realiza la eliminación de restricciones redundantes según las indicaciones descritas previamente y se obtiene un conjunto de puntos x_{ij} pertenecientes a P.

A continuación se calcula la mediana x_m del conjunto de los x_{ij} (de acuerdo a Aho et al. es posible calcular la mediana de un conjunto en tiempo lineal [1]). Al aplicar la prueba para un punto en la recta al punto x_m, se determina si el

problema es no factible, si el valor x_m es óptimo, o bien si la solución óptima se encuentra en $[d_1, x_m)$ o $(x_m, d_2]$. Si se llega a presentar uno de los primeros dos casos, el problema es resuelto y el algoritmo finaliza. Si no es así, se ha redefinido el intervalo de trabajo, ya que se sabe en cual de ellos se encuentra el óptimo.

Esto se traduce en la eliminación de al menos la mitad de los valores x_{ij} y por lo tanto se puede descartar una de las restricciones por cada uno de estos valores. Lo anterior resulta en la supresión de al menos un cuarto de restricciones iniciales, limitando el problema original a uno de a lo más $\lfloor \frac{3}{4} m \rfloor$ donde m se actualiza en cada iteración.

Puede suceder que al realizar el cálculo de los puntos de intersección no se obtenga alguno factible; esto puede deberse a que: P sea vacío, que los puntos obtenidos no sean factibles, que el óptimo sea uno de los valores extremos del intervalo $[d_1, d_2]$, o bien que la solución óptima sea la intersección de una restricción en I_1 y una en I_2. De ser así, se debe realizar nuevamente el primer paso del algoritmo.

Si después de repetir el proceso de eliminación e intersección de restricciones $\lfloor \frac{m}{2} \rfloor$ veces o si el número de restricciones deja de disminuir, y no se ha obtenido un punto de intersección factible, el problema es no factible y el algoritmo finaliza. Si la solución óptima está dada por la intersección de una restricción en I_1 y una en I_2, o bien por los puntos extremos de $[d_1, d_2]$, el algoritmo va acotando el intervalo de soluciones y eliminando restricciones hasta quedar con a lo más dos restricciones en cada conjunto I_1, I_2. Esta situación se resuelve de manera directa, es decir, simplemente se calculan todas las intersecciones posibles entre las restricciones restantes, (independientemente del conjunto de índices a pertenezcan), se determina cuales de ellas son factibles y éstas son evaluadas en f_1 (junto con los extremos de d_1 y d_2) para determinar la solución óptima. Esta última instancia es claramente lineal.

Una vez que el algoritmo obtiene el punto óptimo x^* y el valor óptimo $f_1(x^*)$ del problema unidimensional, se tiene que $y = f_1(x^*)$ y $x = x^*$ son las soluciones al problema en el plano en su forma de Megiddo en 2.6. Finalmente, para obtener la solución al problema de PL canónico original, se aplica la inversa de los cambios de variables utilizados en el lema 2.2.1 de la sección 2.2.1 y se sustituyen los valores de x, y para determinar el valor y punto óptimo. En la figura 2.9 se presenta un diagrama de flujo que sintetiza este algoritmo.

Nótese que la solución de este algoritmo, $(x^*, f_1(x^*))$ corresponde a un vértice del poliedro generado por las funciones $f_1(x)$ y $f_2(x)$ y por lo tanto del politopo de soluciones del problema 2.6. Como el cambio de variables realizado al problema de PL original es biyectivo, preserva la estructura de P por lo que la solución $(x^*, f_1(x^*))$ transformada con el cambio de variables, pertence también a una cara del politopo P.

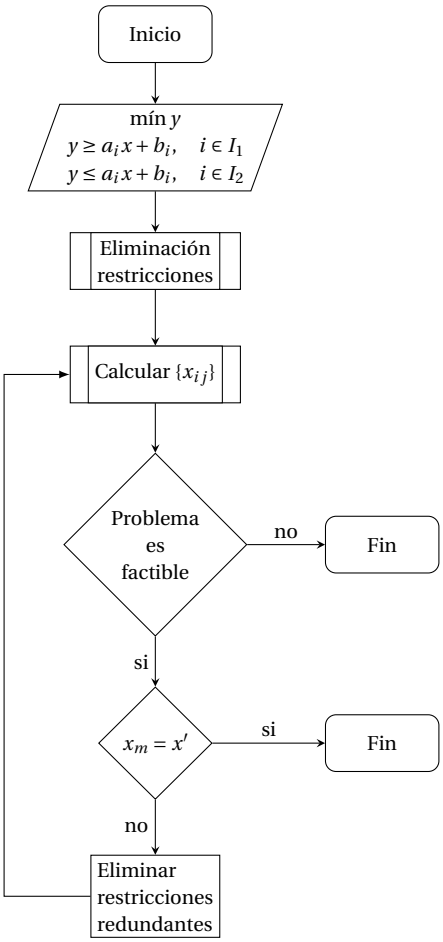


Figura 2.9: Diagrama de flujo para el algoritmo de Megiddo en $\mathbb{R}^2$.

2.2.2. Algoritmo de Megiddo en $\mathbb{R}^3$

Para problemas de PL en el espacio, la primera instancia del algoritmo clasifica las restricciones en distintos subconjuntos y las empareja según el subconjunto al que pertenecen. Si alguna pareja de restricciones corresponde al mismo plano trasladado sobre el plano xy, una de las restricciones será eliminada; de las parejas que no presenten esta propiedad se calculará su intersección, obteniendo una sucesión de rectas $L_1, L_2, ..., L_k$ y se calcula su ángulo. Una vez elegida la mediana de estos ángulos, α_m, las rectas se pueden clasificar según si su ángulo es mayor o menor que α_m. Posteriormente, se forman parejas de rectas tal que una tenga ángulo menor a α_m y otra mayor y se intersectan, obteniendo una sucesión de puntos $(x_{1,1}, x_{2,1}), (x_{1,2}, x_{2,2}), ..., (x_{1,k}, x_{2,k})$. Considerando la mediana $x_{2,m}$ de las primeras entradas de estos puntos se determina la recta $x_2 = x_{1,m}$ a analizar.

La prueba a la recta $x_2 = x_{1,m}$ está constituida en la resolución de a lo más

tres problemas auxiliares de PL de dimensión menor. Resolver el primer problema de PL auxiliar en dimensión menor consiste en minimizar la función $f(x_1, x_{2,m})$ sujeta a $F(x_1, x_{2,m}) \geq 0$ para obtener la solución inicial. Los siguientes problemas de PL auxiliares consisten en estudiar las derivadas laterales de estas funciones sobre el hiperplano, para determinar la ubicación del mínimo global. Una vez resuelta la prueba al hiperplano $x_2 = x_{2,m}$ se ha determinado en cual de los semiespacios generados por el hiperplano se encuentra la segunda coordenada de la solución óptima. De los puntos de intersección que estén en el semiespacio contrario se considera la mediana de su primera coordenada, $x_{1,m}$ y se realiza la prueba al hiperplano $x_1 = x_{1,m}$. Finalizada la prueba se tiene un resultado análogo a la primera coordenada y la intersección de los semiespacios determina un "cuadrante"donde se encuentra el óptimo. Es el cuadrante "opuesto" el que resulta irrelevante para encontrar la solución óptima y por lo tanto una de las rectas correspondiente a los puntos en él y uno de los planos que definan esa recta resultan redundantes. Esto concluye que al final de cada iteración se pueden eliminar aproximadamente $\frac{1}{16}$ de las restricciones en la iteración.

En el espacio, un problema de PL puede ser expresado como

$$\begin{aligned} \text{mín}\, c_1x_1 + c_2x_2 + c_3x_3 & \\ \text{sujeto a} & \\ a_{i1}x_1 + a_{i2}x_2 + a_{i3}x_3 \geq b_i, \quad & i \in [m]. \end{aligned} \tag{2.15}$$

Igual que en la sección anterior, esta formulación se transformará a una equivalente, según el siguiente lema.

Lema 2.2.2. *La formulación descrita en (2.15) para un problema de PL en $\mathbb{R}^3$ es equivalente a*

$$\begin{aligned} \text{mín}\, z & \\ \textit{sujeto a} & \\ z \geq \tilde{a}_i x + \tilde{b}_i y + \tilde{c}_i \quad & i \in I_1, \\ z \leq \tilde{a}_i x + \tilde{b}_i y + \tilde{c}_i \quad & i \in I_2, \\ 0 \geq \tilde{a}_i x + \tilde{b}_i y + \tilde{c}_i \quad & i \in I_3 \end{aligned} \tag{2.16}$$

donde $|I_1| + |I_2| + |I_3| = m$.

Demostración. Al aplicar los cambios de variable $x = x_1$, $y = x_2$ y $z = c_1x_1 + c_2x_2 + c_3x_3$ a las restricciones en (2.15) se obtiene

$$\begin{aligned} \text{mín}\, z & \\ \text{sujeto a} & \\ \left(a_{i1} - \frac{c_1 a_{i3}}{c_3}\right)x + \left(a_{i2} - \frac{c_2 a_{i3}}{c_3}\right)y + \frac{a_{i3}}{c_{i3}}z \geq b_i, & \end{aligned}$$

para todo $i \in [m]$. Si I_1 representa el subconjunto de índices que satisfacen $\frac{a_{i3}}{c_3} > 0$, entonces al multiplicar (2.17) por $\frac{c_3}{a_{i3}}$ se obtiene

$$z \geq \frac{c_3}{a_{i3}} b_i + \left(c_1 - \frac{c_3 a_{i1}}{a_{i3}}\right) x + \left(c_2 - \frac{c_3 a_{i2}}{a_{i3}}\right) y \quad i \in I_1. \tag{2.17}$$

Si I_2 consiste de todos índices para los que el valor de $\frac{a_{i3}}{c_3}$ es negativo, el resultado de multiplicar por $\frac{c_3}{a_{i3}}$ a las desigualdades correspondientes resulta

$$z \leq \frac{c_3}{a_{i3}} b_i + \left(c_1 - \frac{c_3 a_{i1}}{a_{i3}}\right) x + \left(c_2 - \frac{c_3 a_{i2}}{a_{i3}}\right) y \quad i \in I_2. \tag{2.18}$$

Finalmente, si el índice i es tal que $\alpha_{i3} = 0$, entonces la desigualdad (2.17) se simplifica a

$$0 \geq b_i - a_{i1} x - a_{i2} y \quad i \in I_3, \tag{2.19}$$

y el índice pertenecerá al conjunto de índices I_3.

Si para los índices en I_1 y I_2 se realiza el cambio de notación

$$\tilde{a}_i = c_1 - \frac{c_3 a_{i1}}{a_{i3}}, \quad \tilde{b}_i = c_2 - \frac{c_3 a_{i2}}{a_{i3}}, \quad \tilde{c}_i = \frac{a_{i3}}{c_3} b_i$$

y para los índices en I_3

$$\tilde{a}_i = -a_{i1}, \quad \tilde{b}_i = -a_{i2} \quad \tilde{c}_i = b_i,$$

entonces el sistema en (2.15) se puede reescribir como el problema (2.16). □

A la formulación en (2.16) se le llamará la *formulación de Megiddo para el espacio*. Como sólo se va a trabajar con esta formulación, se omitirán las tildes de los coeficientes. De manera similar al problema de PL en el plano, es de interés transformar el problema de PL en $\mathbb{R}^3$ a un problema de dimensión menor, en este caso dimensión 2. Así, asociadas a cada conjunto de restricciones en 2.16 se definen las funciones

$$\begin{aligned} f_1(x,y) &= \text{máx}\{a_i x + b_i y + c_i \,:\, i \in I_1\}, \\ f_2(x,y) &= \text{mín}\{a_i x + b_i y + c_i \,:\, i \in I_2\}, \\ f_3(x,y) &= \text{máx}\{a_i x + b_i y + c_i \,:\, i \in I_3\}, \end{aligned} \tag{2.20}$$

donde cada una de éstas es lineal por trozos y f_1, f_3 son convexas mientras que f_2 es cóncava. Nótese que, las restricciones en 2.16 imponen a las funciones anteriores las siguientes condiciones

$$\begin{aligned} f_1(x,y) \leq z \leq f_2(x,y), \\ f_3(x,y) \leq 0 \end{aligned}$$

Puesto que el problema consiste en minimizar la variable z, la expresión en (2.21) implica que el valor óptimo de z es el mínimo valor que puede alcanzar

la función $f_1(x, y)$ sujeto al cumplimiento de (2.21). Así, el problema (2.16) se puede reescribir como

$$\begin{aligned} \text{mín } & f_1(x, y) \\ & \text{sujeto a} \\ f_1(x, y) - f_2(x, y) &\leq 0 \\ f_3(x, y) &\leq 0. \end{aligned} \tag{2.21}$$

Definiendo la función convexa auxiliar $F(x, y) = \text{máx}\{f_1(x, y) - f_2(x, y)\ ,\ f_3(x, y)\}$, el problema formulado en (2.21) es equivalente a

$$\begin{aligned} \text{mín } & f_1(x, y) \\ & \text{sujeto a} \\ F(x, y) &\leq 0, \end{aligned}$$

reemplazando un problema en $\mathbb{R}^3$ por un problema equivalente en $\mathbb{R}^2$. A esta representación de un problema de PL en el espacio se le llamará *formulación bidimensional*. Nótese que, la factibilidad de un punto (x, y) está sujeta a que $f_1(x, y) - f_2(x, y) \leq 0$ y $f_3(x, y) \leq 0$.

De aquí en adelante, se utilizarán para trabajar la formulación de Megiddo y la formulación bidimensional de un problema en el espacio.

Eliminación de restricciones redundantes

Dada la formulación en (2.16), el algoritmo inicia eligiendo parejas disjuntas i, j de elementos de cada conjunto de índices I_1, I_2, I_3. El criterio para emparejar restricciones es seleccionar los índices $i, i+1$. Nótese que a lo más un elemento de cada conjunto puede quedar sin pareja. De cada pareja de índices se puede o bien eliminar una de las restricciones, o bien obtener una recta generada por la intersección de los planos correspondientes.

Sea i, j una pareja de índices de I_1 tal que de las restricciones correspondientes satisfacen que $(a_i, b_i) = (a_j, b_j)$. Entonces ambas restricciones representan el mismo plano trasladado sobre el plano xy y, por lo tanto, una de las restricciones es redundante. Puesto que para los índices en I_1 el área de interés definida por las restricciones es el semiespacio superior, se puede eliminar la restricción cuya constante sea menor. Si i, j son índices en I_2, argumentos semejantes indican que si $(a_i, b_i) = (a_j, b_j)$, entonces la restricción cuya constante sea mayor resulta redundante.

Si para cualquier pareja de índices i, j en I_1 o I_2 resulta que $(a_i, b_i) \neq (a_j, b_j)$, entonces los planos correspondientes $z = a_j x + b_j y + c_i$ y $z = a_i x + b_i y + c_i$ se intersectan en una recta L_{ij}, que al ser proyectada sobre el plano xy lo divide en dos semiplanos. Estas rectas están definidas como

$$L_{ij} = \{(x, y)\ :\ (b_j - b_i)y = (a_i - a_j)x + c_i - c_j\}.$$

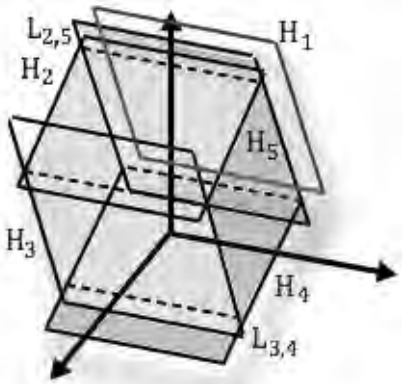


Figura 2.10: Eliminación de restricciones redundantes e identificación de rectas de intersección.

Nótese que las restricciones en I_3 corresponden a planos verticales sobre el plano xy, por lo que su proyección sobre el plano corresponde de manera directa con una recta. Por lo tanto, cada pareja de restricciones con índices en I_3 que no satisfagan la condición $(a_i, b_i) = (a_j, b_j)$ define una pareja de rectas en el plano y ésta se añade al conjunto de rectas generadas en esta subrutina. En caso contrario, se eliminará la restricción que tenga constante menor.

En la figura 2.10 se ilustra un ejemplo de esta subrutina: los planos H_1, H_2, H_5 corresponden a restricciones en I_2, mientras que H_3 y H_4 corresponden a I_1; las rectas punteadas son las intersecciones de los planos y la restricción correspondiente al plano H_1 es redundante. Una vez terminado este proceso, se ha eliminado a los más un cuarto de las restricciones originales para cada conjunto de índices en un tiempo lineal. Este proceso se ve bosquejado en el diagrama 2.11.

Prueba para un recta en el plano

Esta prueba es una generalización de la prueba para un punto en una recta. Su objetivo es determinar, dada una recta en el plano xy, cual de los semiplanos que genera es relevante para el problema. Para esto la prueba analiza una vecindad de un punto sobre la recta, determinando su factibilidad y si el valor de la función objetivo puede disminuir dentro de la vecindad.

Esta búsqueda de la solución óptima es transformada por Megiddo en la resolución de a lo más tres problemas de PL de dimensión menor, permitiendo la solución recursiva de problemas de PL en el espacio, al descomponerlos en problemas de PL de dimensión dos. Cada uno de los problemas de PL de dimensión dos es resuelto con el algoritmo de Megiddo en el plano descrito anteriormente. Así, la prueba consiste en determinar la ubicación de la solución óptima al

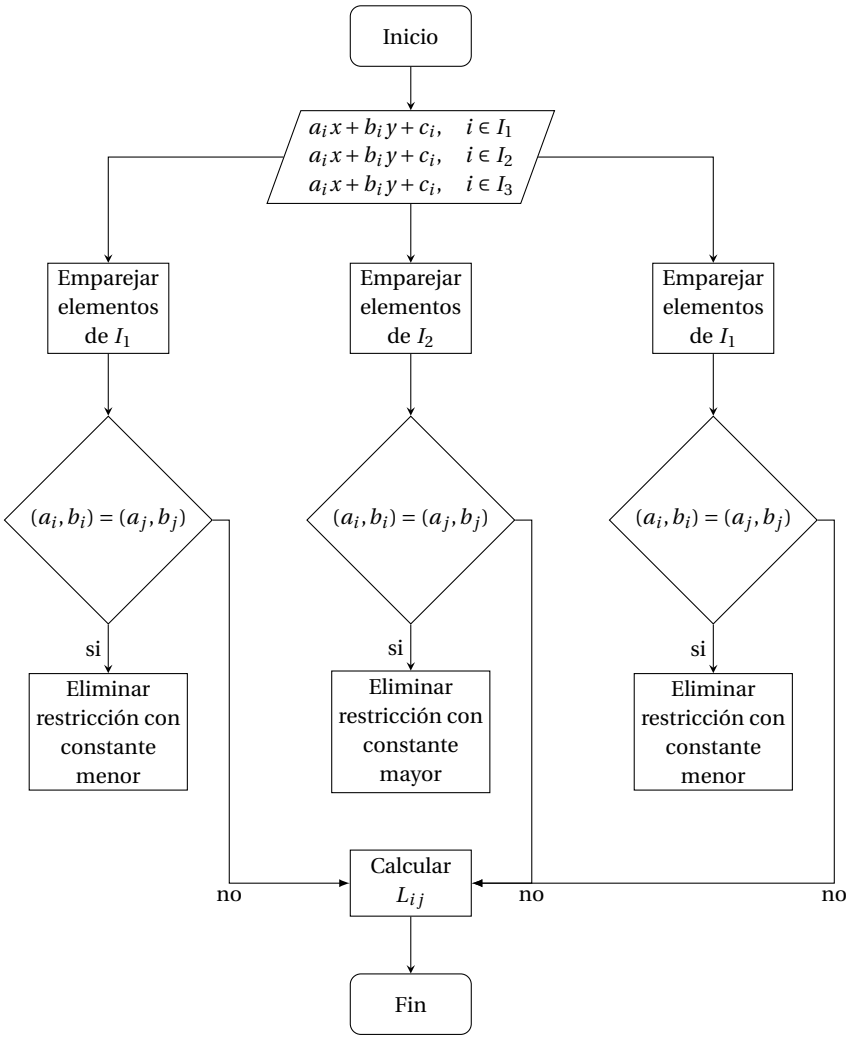


Figura 2.11: Diagrama eliminación de restricciones redundantes.

problema bidimensional respecto a una recta en el plano, y lo lleva a cabo resolviendo a lo más tres problemas de PL de dimensión dos.

Supóngase por razones prácticas, que la recta coincide con el eje x y considérese un punto cualquiera (x_0, y_0) sobre la recta $y = 0$. Si (x_0, y_0) es factible para el problema bidimensional, la prueba determina en cual semiplano definido por $y = 0$ se encuentra la solución óptima, o bien si se encuentra sobre la misma recta; si el punto no es factible, designa en cual semiplano se puede encontrar una solución factible o, en su defecto, la no factibilidad del problema. Por simplicidad supóngase que $(x_0, y_0) = (0, 0)$.

Se sigue de lo anterior, que para analizar la vecindad de (x_0, y_0) son necesarias solamente las restricciones que satisfagan la factibilidad de manera estricta (es decir, las restricciones activas en el punto) y las que violen la factibilidad

sobre ese punto. El resto de las restricciones no aportan información relevante para esta prueba. Se definen entonces los siguientes subconjuntos de índices correspondientes a las restricciones mencionadas

$$\begin{aligned} I_1^* &= \{i \in I_1 \,:\, c_i = f_1(0,0)\}, \\ I_2^* &= \{i \in I_2 \,:\, c_i = f_2(0,0) \text{ si } F(0,0) = f_1(0,0) - f_2(0,0) \geq 0\}, \\ I_3^* &= \{i \in I_3 \,:\, c_i = f_3(0,0) \text{ si } F(0,0) = f_3(0,0) \geq 0\}. \end{aligned} \tag{2.22}$$

Nótese que $I_k^* \subset I_k$ para k=1,2,3; además $I_2^* = \emptyset$ si $F(0,0) < 0$ o $F(0,0) = f_3(0,0)$ y I_3^* es vacío si $F(0,0) < 0$ o $F(0,0) = f_1(0,0) - f_2(0,0)$. Estos conjuntos tienen la información necesaria y suficiente para analizar el comportamiento del problema alrededor del punto.

Si el origen es un punto factible, entonces se desea saber si existe un valor (x, y) factible en una vecindad del origen que disminuya el valor de la función objetivo. Los siguientes resultados relacionan lo anterior con la solución de un problema auxiliar cuando el origen es un valor factible.

Proposición 2.2.3. *Suponiendo que* $(0,0)$ *es un valor factible, la existencia de un punto* (x, y) *factible tal que* $y > 0$, $f_1(x, y) < f_1(0,0)$ *es equivalente a la existencia de un valor real* λ *tal que*

1. $\text{máx}_{i \in I_1^*}\{a_i\lambda + b_i\} < 0$,
2. $\text{máx}_{i \in I_1^*}\{a_i\lambda + b_i\} \leq \text{mín}_{i \in I_2^*}\{a_i\lambda + b_i\}$,
3. $\text{máx}_{i \in I_3^*}\{a_i\lambda + b_i\} \leq 0$.

Demostración. Primero, supóngase que existe el punto (x, y) que satisface las hipótesis correspondientes. Si $\lambda = \frac{x}{y}$ se tiene que

$$\max_{i \in I_1^*}\{(a_i\lambda + b_i)y + c_i\} \leq f_1(\lambda y, y) < f_1(0,0) = \max_{i \in I_1}\{c_i\} = c_k, \tag{2.23}$$

para algún $k \in I_1$. Si existiera un índice j en I_1^* tal que $a_j\lambda + b_j \geq 0$, entonces de (2.23) resulta que

$$(a_j\lambda + b_j)y < c_k - c_j, \tag{2.24}$$

donde el término de la derecha en (2.24) es constante positivo. Sin embargo, para y arbitrariamente grande la desigualdad anterior se viola. Luego, para todo $i \in I_1^*$ se cumple $a_i\lambda + b_i < 0$, por lo que λ satisface el punto 1.

Ahora, si $I_2^* = \emptyset$ la condición 2 se cumple trivialmente, por lo que se puede suponer que el conjunto es no vacío. Esto implica que por ser el origen factible, $F(0,0) = f_1(0,0) - f_2(0,0) = 0$ y por consiguiente para todo $i \in I_1^*$ y $j \in I_2^*$ se tiene $c_i = c_j$. Por la factibilidad de (x, y) se tiene

$$\max_{i \in I_1^*}\{(a_i\lambda + b_i)y + c_i\} \leq f_1(\lambda y, y) \leq f_2(\lambda y, y) \leq \min_{i \in I_2^*}\{(a_i\lambda + b_i)y + c_i\}.$$

En particular, para todo $i \in I_1^*$, $j \in I_2^*$, se tiene

$$(a_i\lambda + b_i)y + c_i \leq (a_j\lambda + b_j)y + c_j$$

donde $c_j - c_i = 0$, así resulta $a_i\lambda + b_i \leq a_j\lambda + b_j$. Se sigue que λ satisface la condición 2. Finalmente si I_3^* fuera un conjunto vacío la condición 3 se satisface por vacuidad, así se puede suponer $I_3^* \neq \emptyset$. Como el origen es una solución factible, $F(0,0) = f_3(0,0) = 0$, lo que implica $c_i = 0$ para todo $i \in I_3^*$. Por otra parte, se tiene

$$\max_{i \in I_3^*}\{(a_i\lambda + b_i)y + c_i\} \leq f_3(\lambda y, y) \leq 0,$$

o equivalentemente, $(a_i\lambda + b_i)y \leq 0$ para toda $i \in I_3^*$, lo que implica $a_i\lambda + b_i \leq 0$. Luego, λ satisface la condición 3.

Inversamente, supóngase que existe un valor $\lambda \in \mathbb{R}$ que satisfaga las tres condiciones descritas. Sea $x = \lambda y$ con $y > 0$ suficientemente pequeño para que se cumpla

$$\max_{i \in I_1}\{(a_i\lambda + b_i)y + c_i\} \leq \max_{i \in I_1^*}\{(a_i\lambda + b_i)y + c_i\}, \tag{2.25}$$

$$\min_{i \in I_2}\{(a_i\lambda + b_i)y + c_i\} \geq \min_{i \in I_2^*}\{(a_i\lambda + b_i)y + c_i\}, \tag{2.26}$$

$$\max_{i \in I_3}\{(a_i\lambda + b_i)y + c_i\} \leq \max_{i \in I_3^*}\{(a_i\lambda + b_i)y + c_i\}. \tag{2.27}$$

De la condición 1 y de la definición del conjunto I_1^* se tiene

$$\max_{i \in I_1^*}\{(a_i\lambda + b_i)y + c_i\} = \max_{i \in I_1^*}\{a_i\lambda + b_i\}y + \max_{i \in I_1}\{c_i\} < \max_{i \in I_1}\{c_i\}. \tag{2.28}$$

Uniendo los resultados de las desigualdades (2.25) y (2.28) se llega a

$$f_1(x,y) \leq \max_{i \in I_1^*}\{(a_i\lambda + b_i)y + c_i\} < \max_{i \in I_1}\{c_i\} = f_1(0,0),$$

Ahora, como se mencionó anteriormente, si $I_2^* \neq \emptyset$ la factibilidad del origen implica que $f_1(0,0) = f_2(0,0)$ por lo que de la condición 2 resulta

$$\begin{aligned} \max_{i \in I_1^*}\{(a_i\lambda + b_i)y + c_i\} &= \max_{i \in I_1^*}\{a_i\lambda + b_i\}y + \max_{i \in I_1}\{c_i\} \\ \leq \max_{i \in I_2^*}\{a_i\lambda + b_i\}y + \max_{i \in I_2}\{c_i\} &= \max_{i \in I_2^*}\{(a_i\lambda + b_i)y + c_i\}. \end{aligned} \tag{2.29}$$

Así, de la expresión en (2.29) y las desigualdades en (2.25) y (2.26) resulta $f_1(x,y) \leq f_2(x,y)$. Finalmente de la condición 3 y la definición de I_3^* se satisface

$$\max_{i \in I_3^*}\{(a_i\lambda + b_i)y + c_i\} = \max_{i \in I_3^*}\{a_i\lambda + b_i\}y + \max_{i \in I_3}\{c_i\} \leq 0, \tag{2.30}$$

y por (2.27) $f_3(x,y) \leq 0$. Por lo tanto, el punto (x,y) disminuye el valor de la función objetivo y es factible. □

La proposición anterior se puede presentar como un problema de PL de la siguiente forma: sea $\eta \in \mathbb{R}$ una variable auxiliar tal que

$$\max_{i \in I_1^*}\{a_i\lambda + b_i\} \leq \eta \leq \min_{i \in I_2^*}\{a_i\lambda + b_i\}.$$

Si además $\eta < 0$ y λ satisface las condiciones en la proposición (2.2.3) entonces la solución óptima del problema bidimensional está en el semiplano superior en $\mathbb{R}^2$. Esto se puede traducir a una formulación de Megiddo para el plano auxiliar

$$\begin{aligned} &\min \eta \\ &\text{sujeta a} \\ &\eta \geq a_i\lambda + b_i,\ i \in I_1^*, \\ &\eta \leq a_i\lambda + b_i,\ i \in I_2^*, \\ &0 \geq a_i\lambda + b_i,\ i \in I_3^*. \end{aligned} \tag{2.31}$$

Si la resolución de este problema tiene como resultado que $\eta \geq 0$ entonces la solución óptima se encuentra en el semiplano inferior, o bien el origen es la solución factible. La siguiente proposición presenta un resultado análogo a la proposición 2.2.3 para el caso en que el punto (x, y) se encuentre en el semiplano inferior.

Proposición 2.2.4. *Suponiendo que* $(0,0)$ *es un valor factible, la existencia de un punto* (x, y) *tal que* $y < 0$, $f_1(x, y) < f_1(0,0)$ *y* $F(0,0) \leq 0$ *es equivalente a la existencia de un valor real* λ *tal que*

1. $\min\{a_i\lambda + b_i \,:\, i \in I_1^*\} > 0$,
2. $\min\{a_i\lambda + b_i \,:\, i \in I_1^*\} \geq \max\{a_i\lambda + b_i \,:\, i \in I_2^*\}$,
3. $\min\{a_i\lambda + b_i \,:\, i \in I_3^*\} \geq 0$.

Demostración. La demostración resulta análoga a la demostración de la proposición 2.2.3. □

La proposición anterior se puede representar como un problema de PL definiendo $\eta \in \mathbb{R}$ como una variable auxiliar tal que

$$\max_{i \in I_1^*}\{a_i\lambda + b_i\} \geq \eta \geq \max_{i \in I_2^*}\{a_i\lambda + b_i\}$$

Si además $\eta > 0$ y λ satisface las condiciones en la proposición 2.2.4 entonces la solución óptima del problema bidimensional está en el semiplano superior en $\mathbb{R}^2$. El problema de PL asociado es el siguiente

$$\begin{aligned} &\max \eta \\ &\eta \leq a_i\lambda + b_i,\ i \in I_1^* \\ &\eta \geq a_i\lambda + b_i,\ i \in I_2^* \\ &0 \leq a_i\lambda + b_i,\ i \in I_3^*. \end{aligned} \tag{2.32}$$

Si las soluciones a los problemas (2.31) y (2.32) resultan $\eta \geq 0$ y $\eta \leq 0$ respectivamente, entonces buscar soluciones factibles fuera de la recta $y = 0$ no mejora la función objetivo y por lo tanto, el origen es la solución factible al problema bidimensional.

Ahora, si el origen no es una solución factible, se busca un punto (x, y) que intuitivamente se acerque más a ser factible o bien que $F(x, y) < F(0,0)$ y, si es posible, logre ser una solución factible. Las siguientes proposiciones presentan los argumentos para mostrar la equivalencia entre un problema de optimización auxiliar y la existencia de dicho punto en alguno de los semiplanos generados por la recta $y = 0$.

Proposición 2.2.5. *Si el origen no es una solución factible, la existencia de un punto (x, y) con $y > 0$ y que satisfaga $F(x, y) < F(0,0)$ es equivalente a la existencia de un valor real λ tal que*

1. $\text{máx}_{i\in I_1^*}\{a_i\lambda + b_i\} < \text{mín}_{i\in I_2^*}\{a_i\lambda + b_i\}$,
2. $\text{máx}_{i\in I_3^*}\{a_i\lambda + b_i\} < 0$.

Demostración. Supóngase que existe el punto (x, y) con $y > 0$ tal que $F(x, y) < F(0,0)$ y tomando $\lambda = \frac{x}{y}$ se tiene $f_1(x, y) - f_2(x, y) < f_1(0,0) - f_2(0,0)$ si y solo si

$$\operatorname*{máx}_{i\in I_1}\{(a_i\lambda + b_i)y + c_i\} - \operatorname*{mín}_{i\in I_2}\{(a_i\lambda + b_i)y + c_i\} < \operatorname*{máx}_{i\in I_1}\{c_i\} - \operatorname*{mín}_{i\in I_2}\{c_i\}. \tag{2.33}$$

Por otra parte

$$\begin{aligned} &\operatorname*{máx}_{i\in I_1}\{(a_i\lambda + b_i)y + c_i\} - \operatorname*{mín}_{i\in I_2}\{(a_i\lambda + b_i)y + c_i\} \\ &\geq \operatorname*{máx}_{i\in I_1^*}\{(a_i\lambda + b_i)y + c_i\} - \operatorname*{mín}_{i\in I_2}\{(a_i\lambda + b_i)y + c_i\} \\ &= \operatorname*{máx}_{i\in I_1^*}\{(a_i\lambda + b_i)y\} + \operatorname*{máx}_{i\in I_1}\{c_i\} - \operatorname*{mín}_{i\in I_2}\{(a_i\lambda + b_i)y + c_i\}, \end{aligned} \tag{2.34}$$

ya que $c_i = \text{máx}_{j\in I_1}\{c_j\}$ para todo $i \in I_1^*$. De las expresiones en (2.33) y (2.34) resulta

$$\operatorname*{máx}_{i\in I_1^*}\{(a_i\lambda + b_i)y\} - \operatorname*{mín}_{i\in I_2}\{(a_i\lambda + b_i)y + c_i\} < -\operatorname*{mín}_{i\in I_2}\{c_i\}.$$

Reacomodando los términos de la expresión anterior se tiene

$$\operatorname*{máx}_{i\in I_1^*}\{(a_i\lambda + b_i)y\} + \operatorname*{mín}_{i\in I_2}\{c_i\} < \operatorname*{mín}_{i\in I_2}\{(a_i\lambda + b_i)y + c_i\} \leq \operatorname*{mín}_{i\in I_2^*}\{(a_i\lambda + b_i)y + c_i\},$$

que es equivalente a

$$\operatorname*{máx}_{i\in I_1^*}\{(a_i\lambda + b_i)y\} + \operatorname*{mín}_{i\in I_2}\{c_i\} \leq \operatorname*{mín}_{i\in I_2^*}\{(a_i\lambda + b_i)y\} + \operatorname*{máx}_{i\in I_2}\{c_i\}.$$

Por lo tanto, $f_1(x, y) - f_2(x, y) < f_1(0,0) - f_2(0,0)$ si y sólo si

$$\operatorname*{máx}_{i\in I_1^*}\{a_i\lambda + b_i\}y < \operatorname*{mín}_{i\in I_2^*}\{a_i\lambda + b_i\}y.$$

Ahora, $f_3(x,y) < f_3(0,0)$ si y solo si $\max_{i\in I_3^*}\{(a_i\lambda + b_i)y\} < 0$ puesto que

$$\begin{aligned}\max_{i\in I_3}\{c_i\} &> \max_{i\in I_3}\{(a_i\lambda + b_i)y + c_i\}\\ &\geq \max_{i\in I_3^*}\{(a_i\lambda + b_i)y + c_i\}\\ &= \max_{i\in I_3^*}\{(a_i\lambda + b_i)y\} + \max_{i\in I_3}\{c_i\}.\end{aligned}$$

□

La proposición anterior indica que si el máximo entre $\max_{i\in I_1^*}\{a_i\lambda + b_i\} - \min_{i\in I_2^*}\{a_i\lambda + b_i\}$ y $\max_{i\in I_3^*}\{a_i\lambda + b_i\}$ es un valor negativo entonces existe un valor (x,y) en el semiplano superior que puede mejorar la no factibilidad del origen. A partir de esta proposición se define la función convexa lineal por trozos

$$F^*(\lambda) = \max\{\max_{i\in I_1^*}\{a_i\lambda + b_i\} - \min_{i\in I_2^*}\{a_i\lambda + b_i\}, \max_{i\in I_3^*}\{a_i\lambda + b_i\}\}. \tag{2.35}$$

Esta función es la generalización de la función auxiliar $f(x)$ descrita en (2.14 utilizada en la prueba de un punto en un recta cuando el punto no es factible. El algoritmo de Megiddo para el plano, también se puede utilizar para minimizar esta función de la siguiente manera: considérense parejas disjuntas i, j de elementos de cada uno de los conjuntos I_1^*, I_2^*, I_3^*; utilizando las restricciones correspondientes en su forma $a_k\lambda + b_k$ calcúlese el punto de intersección de estas parejas de restricciones. Posteriormente, se calcula la mediana x_m de estos puntos y se evalúa $F^*(x_m)$. Si

$$F^*(x_m) = \max_{i\in I_1^*}\{a_i x_m + b_i\} - \min_{i\in I_2^*}\{a_i x_m + b_i\}, \tag{2.36}$$

entonces se realiza el análisis de las derivadas laterales presentado en la sección 2.2.1 a la función en (2.36) para encontrar su mínimo. Por el contrario, si

$$F^*(x_m) = \max_{i\in I_3^*}\{a_i x_m + b_i\}, \tag{2.37}$$

se aplica el análisis de las derivadas laterales en la sección 2.2.1 a la función en (2.37). Una vez realizado el análisis, se ha redefinido el intervalo de búsqueda de una solución factible y se ha eliminado una proporción de las restricciones. Entonces se efectúan nuevas parejas con las restricciones sobrevivientes y se ejecuta nuevamente el análisis, hasta obtener el valor mínimo de la función F^*.

Si $F^*(\lambda) < 0$ entonces se debe buscar soluciones factibles en el semiplano superior. En caso contrario puede existir una solución factible en el semiplano inferior, o bien el problema no es factible. La siguiente proposición presenta condiciones análogas a la proposición 2.2.5 para encontrar un punto factible en el semiplano inferior del plano.

Proposición 2.2.6. *La existencia de un punto (x,y) tal que $y < 0$, $f_1(x,y) < f_1(0,0)$ y $f(0,0) \leq 0$ es equivalente a la existencia de un λ tal que*

1. $\text{mín}\,\{a_i\lambda + b_i \,:\, i \in I_1^*\} > \text{máx}\,\{a_i\lambda + b_i \,:\, i \in I_2^*\}$,

2. $\text{mín}\,\{a_i\lambda + b_i \,:\, i \in I_3^*\} > 0$.

Demostración. La demostración es semejante a la proposición anterior. □

De manera análoga a la proposición anterior, la proposición 2.2.6 induce la construcción de una función convexa lineal por trozos

$$F^*(\lambda) = \text{mín}\{\text{mín}_{i\in I_1^*}\{a_i\lambda + b_i\} - \text{máx}_{i\in I_2^*}\{a_i\lambda + b_i\}, \text{mín}_{i\in I_3^*}\{a_i\lambda + b_i\}\}. \tag{2.38}$$

Esta función es minimizada por el algoritmo de Megiddo para $\mathbb{R}^2$ de manera semejante a la función (2.35). Si el valor mínimo de esta función es positivo, entonces la proposición 2.2.6 indica que existe un valor factible en el H_-. De no ser el caso, se ha determinado que la función objetivo tiene su valor mínimo en el origen, sin embargo, no es un valor factible y por lo tanto el problema es no factible.

Una vez descrita la teoría pertinente se presenta la prueba a una recta en un plano. La prueba inicia resolviendo el primer problema de PL auxiliar de dimensión dos,

$$\begin{aligned} \text{mín}\, & f_1(x,0) \\ \text{sujeta a} & \\ F(x,0) \le 0, & \end{aligned} \tag{2.39}$$

para obtener una solución óptima del problema bidimensional restringido a la recta $y = 0$. La aplicación de este algoritmo puede dar como resultado que el problema no es factible, que el problema no es acotado, o bien puede encontrar una solución óptima $(x^*, 0)$. Claramente, si al resolver el problema auxiliar resulta que es no acotado, entonces el problema original es no acotado y la prueba termina, pues no hay nada que hacer.

Por lo tanto, supóngase que la solución del problema auxiliar (2.39) genera un valor óptimo $(x^*, 0)$ el cual puede ser factible o no. Ahora, se estudiará el comportamiento de una función objetivo en una vecindad de este punto. Sin pérdida de generalidad supóngase que $x^* = 0$ (si no es así, se puede trasladar el sistema coordenado en tiempo lineal). Genérense los subconjuntos de índices correspondientes a restricciones activas en el origen I_1^*, I_2^* y I_3^*.

Si $\mathbf{0}$ es un valor factible se resuelve el problema de PL (2.31). Si $\eta < 0$ la proposición 2.2.3 indica que la solución óptima se encuentra en H_+. Si $\eta \ge 0$ se resuelve el tercer problema auxiliar (2.32): si $\eta > 0$ según la proposición 2.2.4 el óptimo se encuentra en el semiplano inferior; si $\eta \ge 0$ significa que el valor de $f_1(x, y)$ no puede ser mejorada en ningún semiplano y por lo tanto, el origen es el valor óptimo.

Ahora, si el origen es no factible, se minimiza la función (2.35). Si el valor de $F^*(\lambda) < 0$ entonces la proposición 2.2.5 indica que se debe buscar soluciones

factibles en H_+, en caso contrario, se minimiza la segunda función auxilar (2.38). La proposición 2.2.6 señala que si el valor de $F^*(\lambda) > 0$ entonces se debe buscar soluciones factibles en H_-, en caso contrario, la función $f_1(x, y)$ tiene su mínimo en el origen y éste no es una solución factible, por lo que el problema es no factible. En la figura 2.12 se presenta un diagrama de flujo que bosqueja este algoritmo.

Así, dada una recta y un punto sobre ella, esta prueba puede determinar en tiempo lineal si el problema es factible o no, si el punto es óptimo, o bien en cual cuadrante del plano se puede encontrar la solución óptima. Es importante observar que en la prueba se realizan dos transformaciones del sistema coordenado: la primera transformación tiene por objeto que la recta de interés coincida con el eje x; la segunda transformación se realiza para trasladar la solución óptima del problema bidimensional restringida al eje x para que corresponda con el origen. Esta última traslación se realiza siempre y cuando el primer problema auxiliar resulte factible.

Algoritmo de Megiddo para $\mathbb{R}^3$

A grandes rasgos el algoritmo determina dos rectas, $y = y_m$ y $x =_m$, a las cuales aplica la prueba descrita en la sección anterior. Al finalizar ambas pruebas, se tiene un semiplano por cada variable en donde se puede encontrar la solución óptima (o una solución factible). Al intersectar estos semiplanos se obtiene un cuadrante en donde se sabe existe que exista una solución óptima, acotando así el poliedro de soluciones.

Entonces, dado un problema de PL en $\mathbb{R}^3$ en formulación de Megiddo, se procede a realizar la ronda de eliminación de restricciones redundantes y la obtención de las rectas de intersección de planos presentada anteriormente. De este primer subproblema, considérese el conjunto de todas las rectas resultantes L_{ij} que por razones prácticas se renombran sin ningún orden en particular como $L_1, L_2, ..., L_k$, donde k el número de rectas de intersección. Se calcula el ángulo $\theta_i \in [-\frac{\pi}{2}, \frac{\pi}{2}]$ entre cada recta L_i y la parte positiva del eje x, para $i = 1, ..., k$. Sea θ_M la mediana de estos ángulos.

Considérese la transformación lineal que lleva el eje x a la recta $y = \tan(\theta_M)$. Esto define un nuevo sistema coordenado en donde a lo más la mitad de las rectas L_i tienen ángulo no positivo y la otra mitad ángulo no negativo. Reúnanse en pares una recta con ángulo no positivo y una con ángulo no negativo. Obsérvese que a lo más puede quedar una recta sin aparear y se tienen $\lfloor \frac{k}{2} \rfloor$ pares. Para cada pareja de rectas, se calcula su punto de intersección (x_{ij}, y_{ij}). Si las rectas no se intersectan, significa que las rectas son paralelas al eje x y entonces se define el valor y_{ij} como el promedio entre las coordenadas y de las rectas L_i, L_j. Al calcular la mediana y_M del conjunto de valores y_{ij} se obtiene la primera recta $y = y_M$ y se le aplica la prueba de la sección anterior.

Una vez determinado el semiplano de interés para la coordenada y, se identifican los puntos de intersección cuya coordenada y pertenezca al semiplano

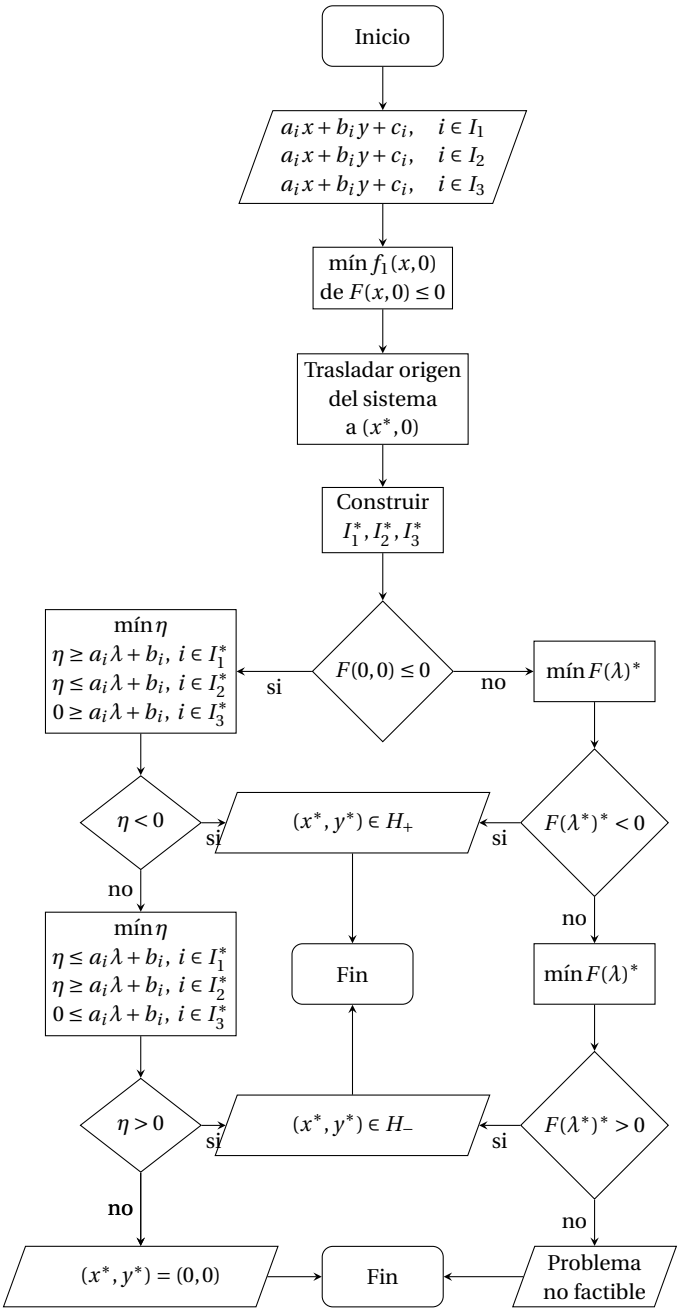


Figura 2.12: Esquema de la prueba para una recta en el plano.

contrario. De estos puntos se calcula la mediana x_M de los valores x_{ij} y se define la segunda recta de evaluación $x = x_M$. Ahora, se aplica la prueba a la recta $x = x_M$ obteniendo un semiplano de interés para la coordenada x. La intersección de los cuadrantes de interés para cada coordenada resulta en un cuadrante

en donde puede encontrarse la solución óptima, una solución factible, o bien se ha concluido la no factibilidad del problema. En la figura 2.13 se ilustra un ejemplo del sistema definido por las rectas $x = x_M$ y $y = y_M$; el cuadrante donde se puede encontrar la solución óptima se indica sombreada.

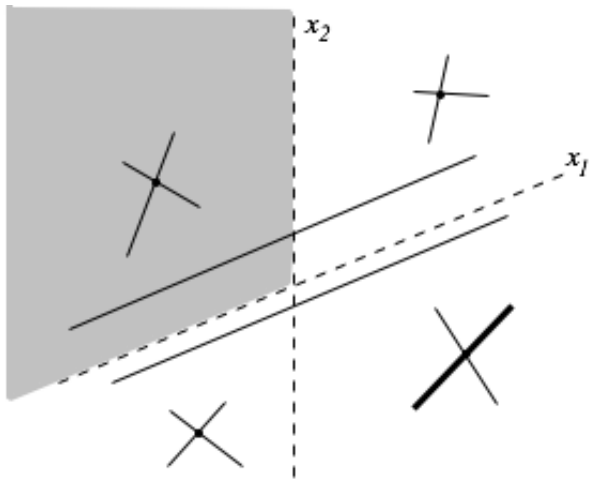


Figura 2.13: Cuadrantes definidos por las rectas $x = x_m$ y $y = y_m$.

Por lo tanto, todos los puntos que se encuentren en el cuadrante "opuesto" son redundantes. Para cada uno de estos puntos una de las rectas que lo define es redundante, a saber, la recta que no intersecte el cuadrante de interés (en la figura 2.13 dicha recta está resaltada). Finalmente cada recta redundante está definida por una pareja de hiperplanos y uno de ellos será redundante para la siguiente iteración del algoritmo. Si la pareja de hiperplanos pertenece a I_1 o I_3, el hiperplano redundante será aquel hiperplano que tenga valores mayores en el cuadrante de interés; para el caso de parejas de hiperplanos en I_2 será aquel que tenga valores menores en el cuadrante de interés.

Puesto que en cada aplicación de la prueba para una recta en el plano se declara que al menos la mitad de los valores y_{ij} y x_{ij} son redundantes, resulta que aproximadamente un cuarto de los puntos (x_{ij}, y_{ij}) son redundantes. Eso implica que al menos $\frac{1}{8}$ de las rectas que definan esos puntos son irrelevantes y por consecuencia al menos $\frac{1}{16}$ de restricciones eliminadas para la siguiente iteración. En la figura 2.14 se presenta el diagrama de flujo de este algoritmo.

2.2.3. Complejidad computacional y generalización de algoritmos de Megiddo

Los algoritmos de Megiddo para problemas de PL resultan ser una metodología ingeniosa que se valen de las propiedades geométricas del problema para resolverlo de forma efectiva. Lo novedoso de sus algoritmos es la técnica

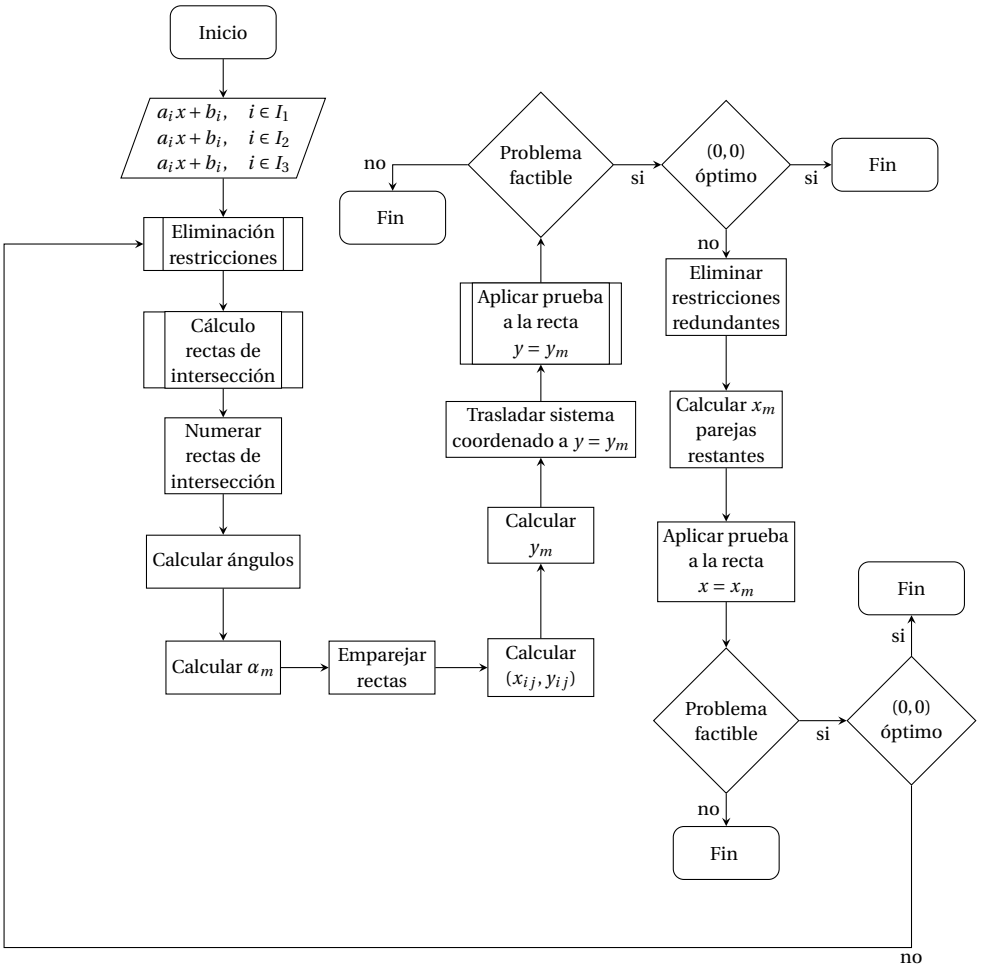


Figura 2.14: Diagrama de flujo del algoritmo de Megiddo en el espacio.

de *prune and search* la cual es muy similar a la idea de dividir y conquistar. Sin embargo, la característica de los algoritmos *prune and search* es que la recursión se va restringiendo a una fracción constante de las restricciones originales (lo cual implica una complejidad computacional menor) hasta que reste un número pequeño de restricciones suficientes para hacer el problema tratable eficientemente. Cada iteración del algortimo requiere un tiempo proporcional al número de restricciones disponibles en ese momento y por lo tanto todo el proceso es lineal.

Entonces, los algoritmos de tipo *prune and search* realizan un trabajo lineal en cada iteración para reducir el número de restricciones por una fracción k. Para el caso de PL en el plano, en cada iteración se eliminan a lo más un cuarto de restricciones ($k = \frac{1}{4}$) en tiempo Cm, donde C es una constante que indica el costo de una iteración, lo que implica que el tiempo total del algoritmo es de la

forma

$$Cm(1+(1-k)+(1-k)^2+(1-k)^3+....) = Cm\left(1+\frac{3}{4}+\frac{3}{4}\right)^2+\left(\frac{3}{4}\right)^3+....,$$

convergiendo a $4Cm$ y por lo tanto el algoritmo tiene una complejidad lineal. De manera análoga para $\mathbb{R}^3$, se elimina en cada iteración $\frac{1}{16}$ de restricciones por lo que el tiempo total de trabajo del algoritmo es

$$Cm\left(1+\frac{15}{16}+\frac{15}{16}^2+\frac{15}{16}^3+....\right),$$

lo cual converge a un tiempo de $16Cm$, y así el algoritmo sigue siendo lineal.

Naturalmente, era interesante saber si este enfoque podía ser generalizado a cualquier problema de PL con dimensión fija. En [31] Megiddo demostró que esto era posible; sin embargo, la generalización resulta ser nada trivial debido a que conlleva una gran cantidad de transformaciones y un número significativo de pruebas a hiperplanos, además que la constante de proporcionalidad crece de manera exponencial. Si el problema es de dimensión d y m restricciones el algoritmo de Megiddo va eliminando una fracción de restricciones cada vez menor al aumentar la dimensión del problema, por lo que la constante de proporcionalidad crece rápidamente. En general, en cada iteración se resuelven a lo más $3\cdot 2^{d-1}$ problemas de tamaño a lo más $m\times(d-1)$, reduciendo un problema de tamaño $m\times d$ a uno de tamaño a lo más $m(1-2^{1-2^d})\times n$ obteniendo un algoritmo de complejidad $O(2^{2^d}m)$, el cual es polinomial si $d\leq \log\log m+O(1)$ [31].

Esto indica que la implementación de este algoritmo para problemas de PL de dimensión mayor que tres se muestra ineficiente en la práctica comparado con algortimos existentes; además la generalización de todos los conceptos y procesos descritos resultan ser no triviales, por lo que se pone en duda la practicidad del algoritmo para problemas de dimensiones mayores a tres. Sin embargo, para problemas de PL de dimensión menor o igual a tres el algoritmo se manifiesta atractivo.

De manera independiente Clarkson [9] y Dyer [14] propusieron mejoras al algoritmo de Megiddo, obteniendo una complejidad de $O(3^{d^2}n)$ que resulta ser polinomial si $d=O(\sqrt{\log n})$.

Capítulo 3

El problema de sujeción robótica

La mano humana cumple con tres funciones importantes: explorar, sujetar y manipular objetos, independientemente de su forma [3]. Por lo tanto, resulta natural que la automatización de distintos procesos de producción llevados a cabo por maquinaria automatizada introdujera la necesidad de la mano robótica. La creación de manos robóticas con múltiples dedos sirvió, entre otras cosas, para analizar los fundamentos mecánicos de la sujeción y manipulación de objetos por manos humanas. Las ventajas de éstas son una mayor estabilidad debido al número de puntos de contacto que sujetan al objeto; son capaces de sostener objetos con formas arbitrarias; es posible aplicar varios momentos angulares al objeto restringido [59], entre otras. La primera mano robótica diseñada para una manipulación diestra fue la mano de Salisbury [51]. El análisis y modelado de la sujeción realizado por Salisbury sentó las bases para las ramas de análisis y síntesis de sujeción, y la investigación de manipulaciones diestras que se llevan a cabo hoy en día [47]. Su trabajo destacó además por utilizar teoría de torsores para trabajar el problema de agarre de una mano robótica.

La teoría de sujeción robótica tiene como propósito analizar y emular la capacidad de restricción y manipulación de un objeto por una mano humana (en relación con la muñeca y la palma de la mano) dado un conjunto adecuado de puntos de contacto y la estabilidad de dicha sujeción [6],[59]. En el estudio de una sujeción robótica, se presentan dos tipos de aprehensión que resultan ser los más importantes, estos son llamados sujeciones de tipo *force-closure* y *form-closure*. Estos términos fueron acuñados por Reuleaux [48] al estudiar la mecánica de algunas máquinas. La primera máquina era una rueda hidráulica, cuyo eje se ubicaba en una ranura de una sección transversal semicircular. Para el correcto funcionamiento de la máquina, era necesario que el peso de la rueda mantuviera el contacto o cercanía entre el eje y la ranura, por lo que el término *force-closure* describe sujeciones en las que su correcto funcionamiento requiere de la aplicación de una fuerza [57]. Sin embargo, si el contacto es mantenido debido a la geometría de los objetos en contacto (como el caso de una pelota dentro de un cilindro), es cuando se produce una sujeción *form-closure*.

En análisis de sujeción, la determinación de alguna de las dos propiedades anteriores queda sujeta al *modelo de contacto* que se emplee. Un modelo de contacto mapea las fuerzas que pueden ser transmitidas de los dedos a la superficie a través de los puntos de contacto a las *fuerzas generalizadas* [26]. Así, la principal diferencia entre una sujeción *force-closure* y una *form-closure* es que para la primera se utilizan modelos de contacto con fricción o modelos de dedos suaves, mientras que la segunda se basa en modelos de contacto sin fricción [63].

Para sujeciones en el espacio, determinar la propiedad de *force-closure* resulta más complicado que determinar la propiedad de *form-closure* debido a la no-linealidad que presentan los modelos de contacto utilizados [62]. Salisbury y Roth [51] demostraron que una condición necesaria y suficiente para tener *force-closure* es que las *fuerzas generalizadas primitivas* generen positivamente todo el espacio 6-dimensional de fuerzas generalizadas. Esto es equivalente a que el origen del espacio de fuerzas generalizadas pertenezca a la envoltura convexa de las fuerzas generalizadas primitivas [33]. De esta forma, distintas equivalencias para la condición de *force-closure* han sido propuestas en las últimas décadas, pocas de ellas para sujeciones en $\mathbb{R}^3$ debido a la mencionada complejidad.

Es de particular interés el estudio de la propiedad *force-closure* y distintos problemas relacionados. Algunos de ellos son determinar la calidad de la sujeción y la optimización de las fuerzas de contacto ejercidas para mantener la sujeción del objeto. En 1989 Nakamura et al. [40] propusieron la primera prueba cualitativa exacta para determinar si una sujeción es *force-closure*. La prueba consiste en la solución de 12 problemas de programación no-lineal con restricciones. Por su parte, en 1992 Liu [27] propuso una técnica de reducción recursiva que permite determinar si el origen del espacio está contenido en la envoltura convexa de las fuerzas generalizadas primitivas en dimensiones superiores, resolviéndolo en la dimensión más baja posible. Posteriormente, Liu y Wang [29] presentaron en 1998 la formulación de una prueba cualitativa para sujeción *force-closure* en $3D$ como un problema de ray-shooting.

En 1999 Bicchi [5] demostró la relación entre una sujeción *force-closure* y el equilibrio de una ecuación diferencial ordinaria incluyendo en su trabajo pruebas tanto cualitativas como cuantitativas. En 2006 Phoka et al. [45] implementan los métodos de algoritmos genéticos, estrategia evolutiva, gradiente conjugado y downhill simplex basados en la distancia Q, como pruebas cualitativas para sujeción *force-closure* en objetos con superficies curvas. La distancia Q utiliza un compacto convexo que contenga al origen como métrica para determinar si el origen pertenece o no en el interior relativo de las fuerzas primitivas generalizadas. También utiliza algoritmos genéticos que explotan las características de la distancia Q para encontrar sujeciones *force-closure* óptimas en curvas paramétricas y superficies paramétricas en $2D$ y $3D$ respectivamente [44]. Independientemente, Niparnan et al. [41] utilizaron métodos heurísticos para una prueba cualitativa sin linearización del cono de fricción. En 2009 Zheng et al.

[61] presentan un algoritmo que calcula la mínima distancia de un punto a un cono convexo en un espacio n-dimensional, aplicándolo como prueba cualitativa de *force-closure*, determinan la equivalencia entre el hecho de que la suma de fuerzas generalizadas normales negativas pertenezcan al cono de fricción y que la distancia entre dicha suma y el cono convexo sea cero.

Por su parte, Trinkle fue el primero en formular la propiedad de *force-closure* como un problema de programación lineal. En su trabajo de 1992 [56], Trikle presenta por primera vez una prueba cuantitativa para determinar si una mano robótica presenta sujeción *force-closure* para una cantidad arbitraria de puntos de contacto. En 2003 Zhu y Wang [63] expusieron el problema de cuantificar una sujeción al minimizar la distancia Q, la cual expresa el mejor desempeño de una sujeción frente a perturbaciones externas. Posteriormente, Han, Trinkle y Li presentan [20] una transformación de las restricciones no lineales del cono de fricción a desigualdades de matrices lineales. Así, formulan los problemas anteriores como un conjunto de problemas de optimización convexa con desigualdades de matrices lineales. Zhu y Ding [62] presentan en 2006 un algoritmo iterativo basado en observaciones de la condición geométrica de la sujeciones del tipo *force-closure*. Con esto, y la discretización del cono de fricción proponen un algoritmo cuantitativo. En 2009 Sahbani et al. en [18] demuestran que las fuerzas generalizadas asociadas a tres puntos de contacto no coplanares en $3D$ forman una base en el espacio de fuerzas generalizadas. Utilizando este hecho, implementan un algoritmo heurístico para una prueba cuantitativa para sujeciones *force-closure*. Finalmente, en 2010 Al-Gallaf [2] presenta una regla basada en redes neuro-difusas para optimización de fuerzas en tiempo real para una mano con n dedos.

La calidad, la estabilidad y la optimización de fuerzas de contacto de una sujeción de tipo *force-closure* son los problemas principales abordados en los artículos anteriores, y a los que se da seguimiento en este trabajo. A continuación se da una breve introducción a los conceptos básicos del problema de sujeción robótica.

3.1. Preliminares

En el estudio de la sujeción robótica se presentan dos tipos de problemas principales: análisis de sujeción y síntesis de sujeción [26]. El primero consiste en determinar si la sujeción de un objeto por una mano robótica y n puntos de contacto es *estable*, es decir, si la sujeción es capaz de resistir cualquier perturbación aplicada al objeto. Para el análisis de sujeción se desarrollan e implementan pruebas cualitativas y cuantitativas basadas en distintas propiedades de la sujeción y las características del problema a trabajar. Las pruebas cualitativas, simplemente determinan si la sujeción estudiada cumple alguna propiedad, como *force-closure* o *form-closure* (como se verá más adelante), las cuales determi-

nan la estabilidad de la sujeción estudiada. Las pruebas cuantitativas aportan una medida de qué tan bueno es el tipo de sujeción. La síntesis de sujeción es el problema de determinar la posición correcta de la mano respecto al objeto y encontrar un conjunto de puntos de contacto adecuado para la manipulación óptima de un objeto determinado [47]. Algunos artículos donde se desarrollan algoritmos de síntesis de sujeción son [3], [13], [50], [63].

La *sujeción* de un objeto por una mano robótica se puede describir mediante un conjunto de *puntos de contacto* y/o un conjunto de *fuerzas generalizadas aplicadas* a la superficie de dicho objeto.

Al punto donde el dedo robótico toca la superficie del objeto se le llama *punto de contacto*. Es importante mencionar que las características de fricción en estos puntos depende del material de las superficies en contacto y presenta poca dependencia del área de contacto entre éstas [46]. Para balancear cualquier fuerza y/o torque externos, la mano robótica puede aplicar una fuerza al objeto a través de sus dedos. Esta fuerza es conocida como la *fuerza de sujeción* o *contacto*. En cada uno de los puntos de contacto se define una *fuerza generalizada*, la cual es aplicada con dirección al centro de masa del objeto y es estrictamente dependiente de la fuerza de sujeción. Una fuerza generalizada $\mathbf{w}$ es un vector con componentes determinados por la fuerza de sujeción y el torque; es decir, $\mathbf{w} = (\mathbf{f}, \boldsymbol{\tau})$, donde $\mathbf{f}$ es la fuerza de sujeción y $\boldsymbol{\tau}$ es el torque o momento angular del objeto y depende de $\mathbf{f}$ y del centro de masa de objeto. Al espacio de todas las fuerzas generalizadas posibles que se pueden aplicar a un objeto en cada punto de contacto es llamado *espacio de fuerzas generalizadas*.

Existen distintos tipos de sujeción que pueden ser llevados a cabo por una mano robótica. Por lo tanto, para determinar el tipo de aprehensión adecuado para sujetar o manipular un objeto específico, se debe analizar el caso particular a estudiar y asignarle un *modelo de contacto* apropiado, el cual determina las fuerzas o torques que pueden ser ejercidas por la mano sobre objeto. Estos modelos crean una caracterización de las fuerzas generalizadas así como los movimientos de los dedos sobre la superficie del objeto [23]. Dicha caracterización es especificada por la geometría de las supeficies en contacto y las caracterísitcas de fricción determinadas por los materiales de las superficies [23]. Los principales modelos de contacto son:

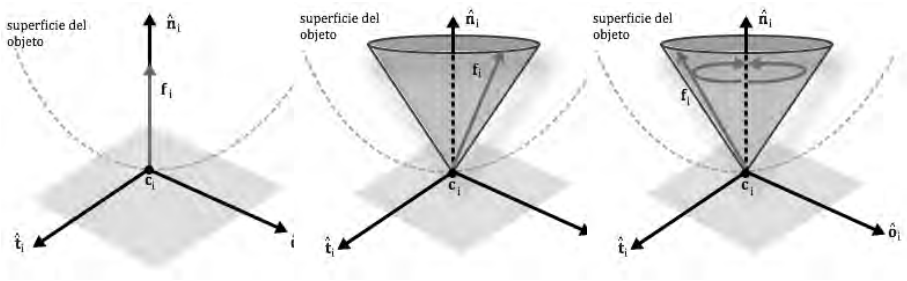
a) Modelo de contacto puntual con fricción.

b) Modelo de contacto puntual sin fricción.

c) Modelo de dedo suave.

Los *modelos dóciles* o de *dedo suave* (figura 3.1c), suponen que la mano es deformable y el objeto que sostiene es rígido. Estos modelos resultan ser un poco complicados debido a la complejidad de una modelación detallada de las deformaciones posibles de los dedos.

Los *modelos puntuales* o de *cuerpos rígidos*, suponen que tanto los dedos como el objeto sujetado son sólidos y rígidos, por lo que no se generan deformaciones en los puntos de contacto y las áreas de contacto entre el dedo y la superficie del objeto son puntuales [23]. Las fuerzas de contacto en estos modelos surgen de las restricciones de incompresibilidad e impenetrabilidad entre objetos rígidos y fuerzas friccionales superficiales. Los modelos puntuales son sencillos de modelar y resolver, y resultan apropiados para determinar si una sujeción es capaz de sostener cierto objeto.



(a) Modelo de contacto sin fricción. **(b)** Modelo de contacto con fricción. **(c)** Modelo de contacto con dedo suave.

Figura 3.1: Modelos de contacto utilizados en la robótica.

Los modelos puntuales pueden ser *sin fricción* o *con fricción*. Los primeros sólo pueden transmitir fuerzas a lo largo del vector normal a la superficie del objeto en el punto de contacto (ver figura 3.1a). Este tipo de modelo difícilmente puede representar una situación real y, por lo tanto, su implementación puede resultar en efectos negativos para el desempeño de la mano robótica.

El modelo de contacto con fricción (figura 3.1b) es utilizado cuando existe suficiente fricción entre las superficies y por lo tanto se pueden transmitir fuerzas en dirección normal y tangencial a la superficie del objeto pero los componentes de momento no son transmitidos [26]. Un ejemplo de estos modelos es el modelo de fricción de Coulomb o fricción seca, cuyas hipótesis indican que la fricción se opone al movimiento y su magnitud es independiente de la velocidad en el punto de contacto [26]. Este modelo supone que la fuerza de fricción F_f entre dos cuerpos en contacto es proporcional a la fuerza normal F_n, $F_f \leq \mu F_n$ donde μ es el coeficiente de fricción dependiente de los materiales en contacto [46].

Si al ejercer una fuerza en el punto de contacto la dirección de deslizamiento del dedo robótico no es uni-direccional, la fuerza de contacto pertenece a un *cono de fricción*. Este cono de fricción tiene vértice en el punto de contacto y semiángulo $\lambda = \tan^{-1}(\mu)$, y todas las fuerzas de sujeción que pueden ser aplicadas al objeto son restringidas al interior del cono para evitar que se deslice el dedo. La sujeción de un objeto está en equilibrio si la suma de toda las fuerzas y momentos actuando sobre el cuerpo resulta cero.

Una sujeción en equilibrio puede ser estable o inestable. Así, una sujeción

puede neutralizar total o parcialmente la movilidad del objeto dada la presencia de perturbaciones externas sobre éste [39]. Por lo tanto, una vez determinado el modelo de contacto adecuado para el problema de sujeción a estudiar, se analiza el conjunto de puntos de contacto de ésta para determinar si es capaz de resistir la aplicación de fuerzas externas al objeto.

Si la sujeción es capaz de aplicar fuerzas generalizadas específicas en cada punto de contacto para neutralizar cualquier fuerza externa aplicada al objeto, entonces es del tipo *force-closure* [47]. Esto implica que la propiedad de *force-closure* es indispensable para tener una sujeción estable. Por su parte, una sujeción *form-closure* determina la capacidad de neutralizar movimientos basándose únicamente en restricciones de contacto unilaterales sin fricción; es considerada una propiedad netamente geométrica, es decir, depende de la geometría de la sujeción más que de las fuerzas que pudieran ser aplicadas por los dedos robóticos [5]. *Force-closure* se puede considerar como una versión más realista que *form-closure*, en el sentido de permitir a las fuerzas de fricción ayudar a mantener el balance de las fuerzas generalizadas aplicadas [47].

El presente trabajo se enfoca en el problema de análisis de sujeción de determinar si ésta es de tipo *force-closure* y los distintos problemas relacionados con la sujeción.

3.2. Los problemas de sujeción *force-closure*

Debido a la no linealidad del cono de fricción, determinar si una sujeción es *force-closure* resulta un problema complicado. Sin embargo, la propiedad de *force-closure* es un prerrequisito para una sujeción estable y, por lo tanto, distintos métodos y algoritmos han sido creados para comprobar dicha propiedad [60]. La propiedad de *force-closure* está asociada a distintos problemas de interés:

a) Prueba cualitativa de *force-closure*.

b) Evaluación cuantitativa de *force-closure*.

c) Optimización de fuerzas de contacto.

Los problemas anteriores han sido abordados con diferentes perspectivas: desde programación lineal y no-lineal hasta ecuaciones diferenciales y algoritmos genéticos. Cada uno de estos problemas parte de la siguiente premisa: supongamos una mano robótica con n dedos sujetando un objeto, al cual se aplica alguna perturbación externa. El problema *a*) consiste en el desarrollo e implementación de algoritmos para determinar si la sujeción es o no *force-closure*. El problema *b*) utiliza herramientas similares al problema anterior y determina la calidad de una sujeción de tipo *force-closure*, es decir, se analiza la sujeción y se concluye qué tan estable es la propiedad o "cuánto" le falta para perderla. El problema *c*) es el siguiente: si en cada punto de contacto el dedo robótico ejerce

una fuerza de sujeción, determinar el conjunto de fuerzas de contacto que neutralice la fuerza externa y minimice el esfuerzo de la mano robótica.

En el presente trabajo, se estudia e implementa el enfoque iniciado por Liu y Wang en [29] y mejorado por Zheng y Qian en [60] para resolver el problema a). La propuesta de Liu y Wang fue la formulación de una prueba cualitativa para sujeción *force-closure* en el espacio con fricción como un problema de ray-shooting: dado un politopo y un punto en su interior, el problema ray-shooting consiste en encontrar la intersección de un rayo que parte de dicho punto con la frontera del politopo. Para esto, se basaron en dos propiedades conocidas:

1) Una sujeción es de tipo *force-closure* si y solo si el origen del espacio de fuerzas generalizadas pertenece al interior de la envoltura convexa de las fuerzas generalizadas primitivas.

2) Existe una dualidad polar entre la envoltura convexa de un conjunto de puntos y un politopo representado por un sistema de desigualdades lineales.

Demostrando la equivalencia entre la propiedad 1) y un problema de ray-shooting, simplifican el problema original *a*) a un problema de programación lineal, el cual es equivalente al problema ray-shooting por la propiedad 2. Más aún, presentan una solución al problema *c*) al transformarlo en un problema ray-shooting utilizando la norma L_1 como índice de desempeño de las fuerzas generalizadas y haciendo la simplificación a un problema de PL. Hasta ese momento, el enfoque de Liu y Wang era la prueba cualitativa más rápida y usada [12],[27],[28].

Posteriormente, Zheng y Qian, influenciados por la alta eficiencia y la no limitación de puntos de contacto del enfoque de Liu, utilizaron resultados de teoría de poliedros para mejorar el método. Tomando en consideración la dimensión de la envoltura convexa de las fuerzas generalizadas primitivas evitan resultados erróneos, algo que no se había considerado hasta ese momento. Obviar la dimensión de la envoltura convexa, resulta en un problema de ray-shooting no acotado y/o que el origen sea confundido por un punto interior de la envoltura convexa. Más aún, aumentaron la eficiencia del método al eliminar algunos pasos del algoritmo propuesto por Liu y propusieron un índice de estabilidad para la optimización de fuerzas generalizadas a partir de la formulación ray-shooting.

Capítulo 4

Modelación de la sujeción robótica con programación lineal

En este capítulo se presenta un desarrollo formal y matemático del problema de sujeción con una mano robótica visto en el capítulo 3. Se introducen los preliminares de análisis de sujeción así como la formulación de las fuerzas generalizadas relacionadas con las fuerzas de contacto de una sujeción *force-closure*. Además, se explican los modelos propuestos por Liu y Wang en [29] y por Zheng y Qian en [60], se observan sus diferencias, así como la descripción y validación de sus algoritmos. El contenido de este capítulo se basa en el libro de Pratichizzo y Trinkle, *Grasping Handbook of Robotics* [47], y en el libro de León, Morales, y Sancho, *From Robot to Human Grasping Simulation* [26].

4.1. Modelación matemática

Se pueden construir modelos matemáticos para describir el comportamiento de una mano robótica aplicando un conjunto de fuerzas para sujetar un objeto. Naturalmente, el comportamiento deseado es la inmovilización del objeto, lo que implica que se desea evitar el deslizamiento de los puntos de contacto o la separación de los dedos de la superficie. Las sujeciones que cumplen estos requisitos se llaman sujeciones de tipo *closure*. A continuación se presentan los sistemas de referencia, definiciones y notación pertinentes para modelar el caso particular de sujeciones *force-closure*.

4.1.1. Fuerzas de contacto

A lo largo de la siguiente sección y el resto de la tesis, se considerarán afirmativas las siguientes hipótesis:

- Tanto los dedos de la mano robótica como el objeto sujetado son rígidos.
- Los contactos entre ambas superficies son puntuales.

- El espacio de trabajo de la sujeción es $\mathbb{R}^l$ con $l = 2,3$, mientras que el espacio de trabajo de las fuerzas generalizadas es $\mathbb{R}^g$ con $g = 3,6$ respectivamente.
- Cada dedo toca al objeto en un punto regular, es decir, un punto cuya vecindad en el objeto sea una curva o una superficie suave.
- El número de dedos sujetando al objeto es mayor o igual a dos en el caso de sujeciones en $\mathbb{R}^2$ y mayor igual que tres en el caso de $\mathbb{R}^3$; ésto es un prerrequisito para tener *force-closure* [5].

En un modelo de sujeción robótica se definen tres tipos de sistemas de referencia. El primero es el *sistema inercial* ($\mathbb{R}^l$), que es el espacio donde vive el objeto y en el que las leyes del movimiento cumplen las leyes de Newton. El segundo es el *sistema de referencia del objeto* ($\{O\}$), el cual determina la posición y la orientación del objeto con respecto al sistema inercial. Finalmente, se tienen los *sistemas de referencia locales* ($\{C_i\}$), los cuales están definidos en cada punto de contacto.

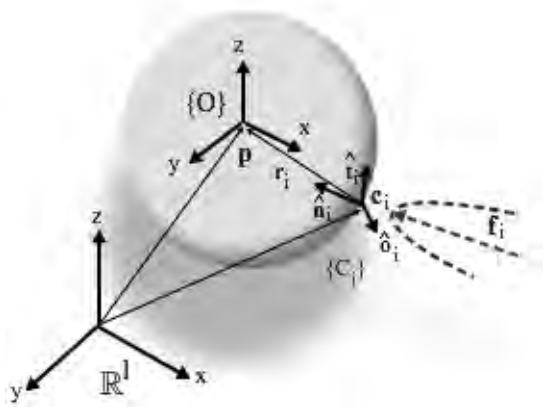


Figura 4.1: Sistemas de referencia utilizados en la modelación matemática de una sujeción *force-closure*.

Considérese una mano robótica con n dedos sujetando un objeto rígido en el sistema de referencia inercial $\mathbb{R}^l$ (figura 4.1). Sea $\mathbf{p}$ un vector en el sistema inercial contenido en el objeto (el centro de masa del objeto es usualmente utilizado como el vector $\mathbf{p}$). El sistema de referencia del objeto $\{O\}$ está fijo en éste y tiene como origen el vector $\mathbf{p}$. La posición del i-ésimo punto de contacto en referencia al sistema inercial está determinada por el vector $\mathbf{c}_i$ y $\mathbf{r}_i = \mathbf{c}_i - \mathbf{p}$ define la posición del i-ésimo punto de contacto respecto a $\mathbf{p}$. Puesto que cada punto de contacto es regular hay un *plano tangente de contacto* P_i bien definido con origen en $\mathbf{c}_i$. Así, se define el i-ésimo sistema de referencia local $\{C_i\}$ con ejes $\{\hat{\mathbf{n}}_i, \hat{\mathbf{o}}_i, \hat{\mathbf{t}}_i\}$ definidos en $\{O\}$, donde $\hat{\mathbf{n}}_i$ es el vector unitario normal al plano

tangente de contacto y $\hat{\mathbf{t}}_i, \hat{\mathbf{o}}_i$ son dos vectores ortogonales unitarios cualesquiera que viven en el plano de contacto P_i. Nótese que si la sujeción vive en $\mathbb{R}^2$ entonces los sistemas de referencia local pierden su componente $\hat{\mathbf{t}}_i$ y lo mismo sucederá con las fuerzas de contacto.

Una *fuerza de contacto o sujeción* $\mathbf{f}$ es una fuerza producida por un dedo y aplicada a la superficie del objeto, con el propósito de balancear o neutralizar alguna fuerza o momento externo (ver figura 4.1). En cada punto de contacto se define una fuerza de contacto $\mathbf{f}_i$, donde $\mathbf{f}_i$ es expresado en el sistema coordenado local $\{C_i\}$ como

$$\mathbf{f}_i = \begin{pmatrix} f_{i,n} \\ f_{i,o} \\ f_{i,t} \end{pmatrix} \tag{4.1}$$

El contacto de la mano con el objeto crea además un *momento* o *torque*, el cual describe la tendencia de una fuerza a hacer girar o rotar un objeto respecto a algún eje. Este torque se define como $\mathbf{r}_i \times \mathbf{f}_i$. La *i-ésima fuerza generalizada* $\mathbf{w}_i \in \mathbb{R}^g$ se define como la combinación de la fuerza de contacto y el momento aplicado por el i-ésimo dedo a $\mathbf{p}$. Ésta es una representación del doble efecto que ejerce $\mathbf{f}_i$ sobre el objeto por cada uno de los dedos y se puede interpretar como la combinación de una fuerza lineal y un momento rotacional a lo largo de un eje. La i-ésima fuerza generalizada se describe como

$$\mathbf{w}_i = \mathbf{G}_i \mathbf{f}_i \tag{4.2}$$

donde la matriz $\mathbf{G}_i$ se llama *matriz de sujeción en el* i*-ésimo contacto* y se define como

$$\mathbf{G}_i = \begin{pmatrix} \hat{\mathbf{n}}_i & \hat{\mathbf{o}}_i & \hat{\mathbf{t}}_i \\ \mathbf{r}_i \times \hat{\mathbf{n}}_i & \mathbf{r}_i \times \hat{\mathbf{o}}_i & \mathbf{r}_i \times \hat{\mathbf{t}}_i \end{pmatrix}.$$

Cabe mencionar que el producto cruz de dos vectores $\mathbf{x}$, $\mathbf{y}$ en el plano está definido como el determinante de la matriz $(\mathbf{xy})$, es decir, $x_1 y_2 - x_2 y_1$.

A la matriz $G \in \mathbb{R}^{g \times 3n}$ se le llama *matriz de sujeción total*, y se compone como

$$\mathbf{G} = [\mathbf{G}_1, \mathbf{G}_2, ..., \mathbf{G}_n].$$

Una fuerza generalizada como su nombre lo indica, es la generalización de una fuerza, libre de sistema coordenado, lo que implica que ésta es invariante del sistema de referencia en el que se trabaje. Esto le da flexibilidad, reduce errores y emula los fenómenos naturales, ya que en la naturaleza no hay sistemas coordenados fijos, y las leyes son dictadas simplemente en un sistema físico [58]. Por lo tanto, su implementación en el estudio de sujeciones robóticas resulta de gran practicidad.

4.1.2. Modelo de contacto

Como se mencionó en el capítulo 3, un modelo de contacto relaciona las fuerzas de contacto con las fuerzas generalizadas correspondientes. De los modelos de contacto mencionados, son de particular interés los modelos de contacto puntual con fricción. El modelo que se usará es el modelo de *fricción de Coulomb*, el cual se basa en la idea de que la fricción se opone al movimiento y su magnitud es independiente de la velocidad y el área de contacto [26].

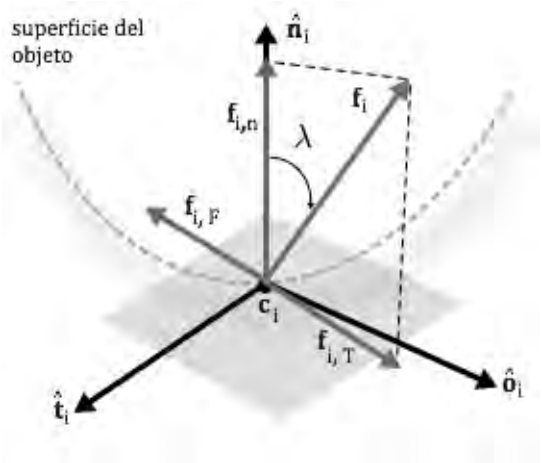


Figura 4.2: Ilustración del modelo de fricción de Coulomb.

Dado un punto de contacto $\mathbf{c}_i$ y el plano de contacto P_i tangente en este punto, la fuerza de contacto $\mathbf{f}_i$ se descompone en una fuerza tangencial $\mathbf{f}_{i,T} = (f_{i,o}, f_{i,t})$ sobre P_i, la cual indica la dirección en que el dedo se puede mover y la componente normal $\mathbf{f}_{i,n}$ de $\mathbf{f}_i$ sobre el eje $\hat{\mathbf{n}}_i$ (ver figura 4.2). La fuerza $\mathbf{f}_i$ también crea por reacción una fuerza de fricción $\mathbf{f}_{i,f}$ en el plano P_i que se opone a la fuerza tangencial y cuya magnitud de acuerdo a la ley de Coulomb es $\mu f_{i,n}$, donde μ es el *coeficiente de fricción*. Este coeficiente depende de los materiales del objeto y los dedos, y se determina empíricamente.

Mientras la fuerza tangente sea menor o igual a la fuerza de fricción, no hay desplazamiento de las superficies en contacto. Así, la condición para que no haya desplazamiento de los dedos es

$$|\mathbf{f}_{i,T}| \leq \mu |\mathbf{f}_{i,n}|. \tag{4.3}$$

Claramente, esta propiedad es indispensable para una sujeción estable. Cuando $|\mathbf{f}_{i,T}| = \mu|\mathbf{f}_n|$, se dice que la *fuerza de fricción* es *limitante*, lo que significa que el contacto no presenta perturbación alguna, sin embargo, un deslizamiento o movimiento está a punto de ocurrir.

Al ángulo λ que se forma entre el vector $\mathbf{f}_i$ y el vector normal se le llama *ángulo de fricción*. Entonces las componentes de $\mathbf{f}_i$ en el vector normal y el vector

tangencial tienen como magnitud

$$|\mathbf{f}_{i,n}| = |\mathbf{f}_i| \cos\lambda, \quad |\mathbf{f}_{i,T}| = |\mathbf{f}_i| \sin\lambda, \tag{4.4}$$

respectivamente. Sustituyendo 4.4 en la expresión 4.3 se tiene que

$$|\mathbf{f}_i| \sin\lambda \leq \mu |\mathbf{f}_i| \cos\lambda$$

y por lo tanto $\lambda \leq \tan^{-1}\mu$. Nótese que cuando la fuerza de fricción es limitante, $\lambda = \tan^{-1}(\mu)$. Esto implica que para evitar deslizamiento entre las superficies en contacto, $\mathbf{f}_i$ debe estar restringida a un cono con vértice en $\mathbf{c}_i$ y eje sobre $\hat{\mathbf{n}}_i$, donde el semiángulo del cono es λ cuando la fricción es limitante.

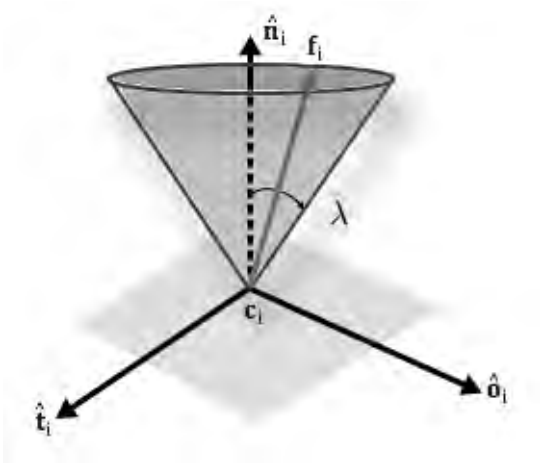


Figura 4.3: Cono de fricción que delimita las fuerzas que se pueden aplicar a un objeto para evitar que los dedos se deslicen.

A este cono se le llama *cono de fricción* y se puede ver una representación de él en la figura 4.3. Matemáticamente, el i-ésimo cono de fricción K_i correspondiente a $\mathbf{f}_i$ se puede expresar formalmente como

$$K_i = \left\{ (f_{i,n}, f_{i,t}, f_{i,o}) : \sqrt{f_{i,t}^2 + f_{i,o}^2} \leq \mu f_{i,n},\ f_{in} \geq 0 \right\},$$

imponiendo a la modelación una restricción no lineal. En el caso de sujeciones en el plano, el cono de fricción pierde su término cuadrático, transformándose en un cono sobre el plano generado por dos vectores unitarios. Así, el i-ésimo cono de fricción para sujeciones en $\mathbb{R}^2$ es descrito por

$$K_i = \left\{ (f_{i,n}, f_{i,o}) : (f_{i,n}, f_{i,o}) = \frac{1}{\sqrt{\mu^2+1}} (c_1(\mu, 1) + c_2(-\mu, 1)) \right\}.$$

4.1.3. Discretización del cono de fricción

La formulación del problema de PL de la sujeción robótica se basa en la linealización del cono de fricción. Éste es aproximado por un cono poliédrico regular de m lados, es decir, como el espacio generado por combinaciones no negativas de m generadores (figura 4.4).

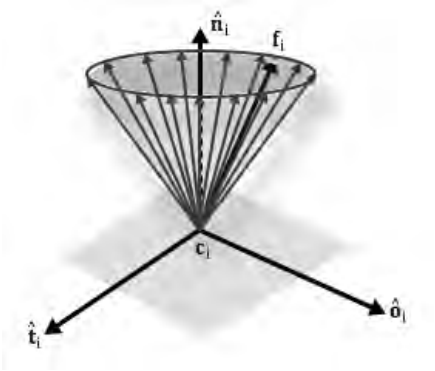


Figura 4.4: Cono de fricción discretizado.

Estos generadores son las aristas del cono poliédrico y se expresan en el sistema local como

$$\mathbf{s}_i = \begin{pmatrix} 1 \\ \mu\cos(\frac{2j\pi}{m}) \\ \mu\sin(\frac{2j\pi}{m}) \end{pmatrix}, \qquad i \in [n],\ j \in [m]. \tag{4.5}$$

Por lo tanto, la i-ésima fuerza de contacto $\mathbf{f}_i$ puede ser expresada como

$$\mathbf{f}_i = \sum_{j=1}^{m} \alpha_{ij}\mathbf{s}_j, \qquad \alpha_{ij} \geq 0, \tag{4.6}$$

donde cada una de sus componentes es especificada por

$$f_{i,n} = \sum_{j=1}^{m} \alpha_{ij}, \quad f_{i,o} = \mu\sum_{j=1}^{m} \alpha_{ij}\cos(\frac{2j\pi}{m}), \quad f_{i,t} = \mu\sum_{j=1}^{m} \alpha_{ij}\sin(\frac{2j\pi}{m}). \tag{4.7}$$

De igual forma, la i-ésima fuerza generalizada $\mathbf{w}_i$ se puede discretizar de la forma

$$\mathbf{w}_i = \sum_{j=1}^{m} \alpha_{ij}\mathbf{w}_{ij}. \tag{4.8}$$

donde $\mathbf{w}_{ij} = \mathbf{G}_i\mathbf{s}_j$ son llamadas *fuerzas generalizadas primitivas*. Estos vectores son los que cargan con toda la información necesaria para describir el comportamiento general de la sujeción.

Por lo tanto, el conjunto total de fuerzas generalizadas, $\mathbf{w}_{net}$, que pueden ser transmitidas al objeto por todos los puntos de contacto es una combinación lineal de las fuerzas generalizadas primitivas, es decir

$$\mathbf{w}_{net} = \sum_{i=1}^{n} \mathbf{w}_i = \sum_{i=1}^{n} \sum_{j=1}^{m} \alpha_{ij} \mathbf{w}_{ij} = \mathbf{W}\boldsymbol{\alpha}^T, \quad \alpha_{ij} \geq 0, \tag{4.9}$$

y el vector de coeficientes $\boldsymbol{\alpha}$ es de la forma

$$\boldsymbol{\alpha} = [\alpha_{11}, ..., \alpha_{1m}, \alpha_{21}, ..., \alpha_{nm}]. \tag{4.10}$$

Finalmente se define como $\mathscr{W}$ la envoltura convexa de las fuerzas generalizadas primitivas.

4.2. Prueba cualitativa para sujeción force-closure

En esta sección se introduce la formulación de Liu y Wang para el problema de determinar si una sujeción tiene la propiedad de *force-closure* [29]. Posteriormente se señalan los fallos y errores que tiene esta teoría y se presentan las mejoras realizadas por Zheng y Qian en [60]. Ellos logran aislar estos errores y desarrollan una clasificación para la envoltura convexa $\mathscr{W}$, mejorando así la teoría de Liu y Wang.

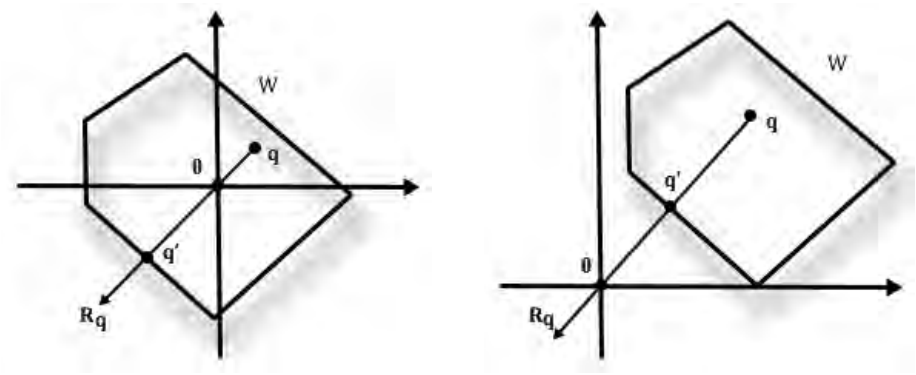
4.2.1. Antecedentes de Liu y Wang

Liu y Wang, presentaron en 1998 [29] un test cualitativo para determinar si una sujeción es *force-closure* basándose en las propiedades de convexidad de $\mathscr{W}$. La formulación presentada transforma el problema original en un problema de ray-shooting y éste a su vez, lo transforma en un problema de programación lineal. Su trabajo se basa en dos propiedades importantes:

- Una sujeción es de tipo *force-closure* si el origen del espacio de fuerzas generalizadas $\mathbb{R}^g$ pertenece al interior de la envoltura convexa de los fuerzas generalizadas primitivas, es decir, $\mathbf{0} \in int(\mathscr{W})$ [33].
- La envoltura convexa de los fuerzas generalizadas primitivas es el polar de politopo dado por un sistema de desigualdades lineales [37].

La prueba propuesta por Liu y Wang se basa en el siguiente argumento: sea $\mathscr{W}$ la envoltura convexa de los fuerzas generalizadas primitivas y $\mathbf{q} \in int(\mathscr{W})$. El rayo $R_{\mathbf{q}}$ que parte de $\mathbf{q}$ y pasa por el origen $\mathbf{0}$ de $\mathbb{R}^g$ intersecta una faceta de $\mathscr{W}$ en un punto $\mathbf{q}'$. Una sujeción de tipo *force-closure* es equivalente a que la distancia entre $\mathbf{q}$ y $\mathbf{q}'$ sea mayor que la distancia $\mathbf{q}$ a $\mathbf{0}$ (ver figuras 4.5a y 4.5b).

Por lo tanto, el algoritmo inicia encontrando un punto interior $\mathbf{q}$ de $\mathscr{W}$. Después detecta el punto de intersección $\mathbf{q}'$ entre el rayo $R_{\mathbf{q}}$ y una determinada faceta de $\mathscr{W}$. Finalmente, compara las distancias $d(\mathbf{q}, \mathbf{q}')$ y $d(\mathbf{0}, \mathbf{q})$. Claramente el



(a) Envoltura convexa $\mathscr{W}$ perteneciente a sujeción tipo *force-closure*.

(b) Envoltura convexa $\mathscr{W}$ perteneciente a sujeción sin propiedad de *force-closure*.

Figura 4.5: Figuras representativas de una sujeción *force-closure* y una no *force-closure*.

punto

$$\mathbf{q} = \frac{1}{nm}\sum_{i=1}^{m}\sum_{i=1}^{n}\mathbf{w}_{ij},$$

llamado el *centroide* de $\mathscr{W}$, es un punto interior de $\mathscr{W}$ y es el punto de elección.

Ahora, el problema de encontrar la faceta intersectada por el rayo $R_{\mathbf{q}}$ está íntimamente relacionado con un problema ray-shooting. Este problema consiste en: dado un conjunto de puntos $M \subset \mathbb{R}^n$ tal que $\mathbf{0} \in conv(M)$, y un rayo R que parta de $\mathbf{0}$, encontrar la faceta de $conv(M)$ intersectada por R. Se puede transformar el problema de encontrar la faceta intersectada por $R_{\mathbf{q}}$ a un problema de ray-shooting de manera sencilla trasladando $\mathscr{W}$ por -$\mathbf{q}$ y transformando el rayo $R_{\mathbf{q}}$ al rayo $R_{\mathbf{0}}$ que inicia en el origen y pasa por $-\mathbf{q}$.

Si $\mathscr{W}_T$ es la traslación de $\mathscr{W}$ por -$\mathbf{q}$, entonces su polar se define como

$$\mathscr{W}_T^* = \{\mathbf{x} \in \mathbb{R}^n : (\mathbf{w}_{ij}^T - \mathbf{q})\mathbf{x} \leq 1,\ \forall\, \mathbf{w}_{ij} \in \mathscr{W}\}, \tag{4.11}$$

el cual ya se sabe que es un politopo que define una correspondencia entre facetas y vértices de $\mathscr{W}_T$ y su polar. Luego el problema ray-shooting es equivalente a solucionar el problema de PL de maximizar el valor de $R_{\mathbf{0}}^T\mathbf{x}$ para $\mathbf{x} \in \mathscr{W}_T^*$, o bien

$$\begin{aligned} &\text{máx}\, R_{\mathbf{0}}^T\mathbf{x} \\ (\mathbf{w}_{ij}^T - \mathbf{q})\mathbf{x} \leq 1,\ &i \in [n],\ j \in [m]. \end{aligned}$$

En la figura 4.6 se presenta una ilustración del conjunto $\mathscr{W}_T$, su polar $\mathscr{W}_T^*$ y el rayo $R_{\mathbf{0}}$; la solución del problema de PL, $\mathbf{x}^*$ da como resultado el punto de intersección $\mathbf{q}'$. Entonces, si $\mathbf{x}^*$ es la solución óptima de este problema de programación lineal, el polar de $\mathbf{x}^*$ es la faceta F^* intersectada por el rayo $R_{\mathbf{0}}$. El cálculo del punto de intersección $\mathbf{q}'$ y de las distancias $d(\mathbf{0}, -\mathbf{q})$ y $d(\mathbf{0}, \mathbf{q}')$ es inmediato. Si $d(\mathbf{0}, \mathbf{q}') > d(\mathbf{0}, -\mathbf{q})$ entonces la sujeción es de tipo *force-closure*; en caso contrario la sujeción no es *force-closure*.

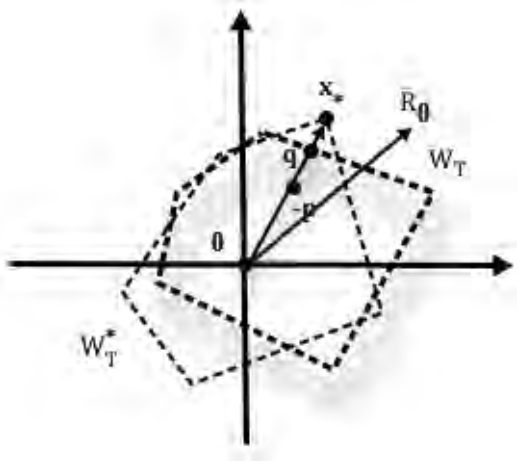


Figura 4.6: Envoltura convexa $\mathcal{W}_T$, su polar $\mathcal{W}_T^*$ y el rayo $R_{\mathbf{0}}$.

4.2.2. Propuesta de Zheng y Qian

A pesar del extensivo uso que se le ha dado a $\mathcal{W}$ a lo largo de la literatura, su dimensión no se había considerado como un problema. Con esto en mente, Zheng y Qian propusieron la siguiente clasificación de $\mathcal{W}$ dependiendo de su dimensión y posición relativa al origen $\mathbf{0}$ del espacio de las fuerzas generalizadas $\mathbb{R}^g$:

a) $dim(\mathcal{W}) < g$ y $\mathbf{0} \notin aff(\mathcal{W})$ (figura 4.7a).

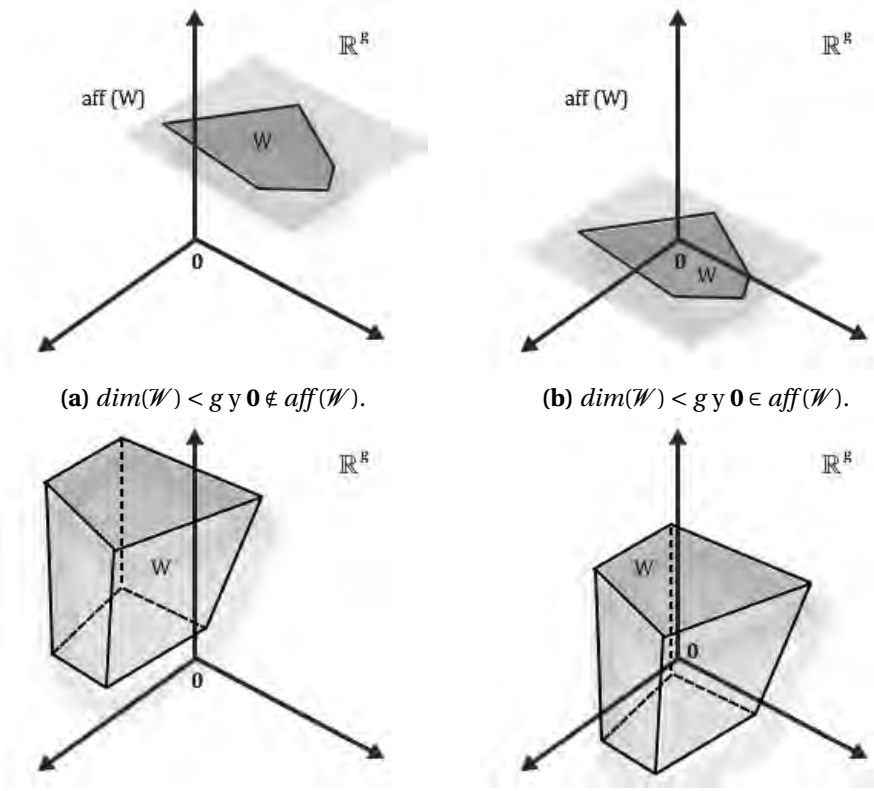
b) $dim(\mathcal{W}) < g$ y $\mathbf{0} \in aff(\mathcal{W})$ (figura 4.7b).

c) $dim(\mathcal{W}) = g$ y $\mathbf{0} \notin ri(\mathcal{W})$ (figura 4.7c).

d) $dim(\mathcal{W}) = g$ y $\mathbf{0} \in ri(\mathcal{W})$ (figura 4.7d).

Zheng et.al. encontraron inconsistencias en la formulación de ray-shooting de Liu y Wang generadas por la falta de consideración de la dimensión de $\mathcal{W}$. Si la dimensión de $\mathcal{W}$ no es completa, la formulación no presenta solución alguna para sujeciones en el caso *a*), y para problemas con las características del caso *b*) en donde $\mathbf{0} \in ri(\mathcal{W})$, la formulación considera al origen $\mathbf{0}$ como un punto interior de $\mathcal{W}$.

Por consiguiente, Zheng y Qian se dieron a la tarea de reestructurar la propuesta de Liu y Wang tomando en cuenta la dimensión de $\mathcal{W}$. Su trabajo se basa en la caracterización de una sujeción *force-closure* dada por Mishra en [33]: una sujeción es *force-closure* si y sólo si $\mathbf{0} \in int(\mathcal{W})$. Una consecuencia inmediata del teorema A.0.1 es la siguiente: como $\mathcal{W}$ es convexo, el origen $\mathbf{0}$ pertenece al interior de $\mathcal{W}$ siempre y cuando $dim(\mathcal{W}) = g$ y $\mathbf{0} \in ri(\mathcal{W})$.



(a) $dim(\mathscr{W}) < g$ y $\mathbf{0} \notin aff(\mathscr{W})$.

(b) $dim(\mathscr{W}) < g$ y $\mathbf{0} \in aff(\mathscr{W})$.

(c) $dim(\mathscr{W}) = g$ y $\mathbf{0} \notin ri(\mathscr{W})$.

(d) $dim(\mathscr{W}) = g$ y $\mathbf{0} \in ri(\mathscr{W})$.

Figura 4.7: Ilustración de los distintos tipos de $\mathscr{W}$.

Sea $\mathbf{w}_c$ el *centroide* de $\mathscr{W}$ definido como

$$\mathbf{w}_c = \frac{1}{mn} \sum_{i=1}^{n} \sum_{j=1}^{m} \mathbf{w}_{ij}.$$

El teorema 1.3.1 indica que toda combinación convexa estrictamente positiva de elementos de un convexo es un punto interior relativo de éste, por lo tanto $\mathbf{w}_c \in ri(\mathscr{W})$. Definiendo $\mathscr{T}$ como la traslación de $\mathscr{W}$ por el centroide $\mathbf{w}_c$, es decir,

$$\mathscr{T} = \mathscr{W} - \mathbf{w}_c, \tag{4.12}$$

se observa que $\mathscr{T}$ es también un politopo en $\mathbb{R}^g$ tal que $\mathbf{0} \in ri(\mathscr{T})$ y por el corolario 1.2.3 resulta que $aff(\mathscr{T})$ es un subespacio vectorial de $\mathbb{R}^g$. Por otra parte, la matriz de vértices de $\mathscr{T}$ es

$$\mathbf{T} = [\mathbf{w}_{11} - \mathbf{w}_c, \ldots, \mathbf{w}_{ij} - \mathbf{w}_c, \ldots, \mathbf{w}_{mn} - \mathbf{w}_c] \in \mathbb{R}^{g \times mn}.$$

Se utilizarán también $\mathscr{T}^*$ (el conjunto polar de $\mathscr{T}$) y $p_{\mathscr{T}^*}$ (la función de soporte)

$$p_{\mathscr{T}^*}(\mathbf{z}) = \max_{\mathbf{x} \in \mathscr{T}^*} \mathbf{z}^T \mathbf{x}. \tag{4.13}$$

Por las definiciones de $\mathscr{T}$ y $\mathscr{W}$ se puede describir $\mathscr{T}^*$ como

$$\mathscr{T}^* = \{\mathbf{z} \in \mathbb{R}^g : \mathbf{z}^T(\mathbf{w}_{ij} - \mathbf{w}_c) \leq 1, i \in [n], j \in [m]\}. \tag{4.14}$$

Proposición 4.2.1. *Las siguientes proposiciones son verdaderas:*

1. $dim(\mathscr{W}) = g$ *si y solo si* $dim(\mathscr{T}) = g$.
2. $dim(\mathscr{T}) = g$ *si y solo si el rango*$(\boldsymbol{T}) = g$.
3. $\boldsymbol{0} \in ri(\mathscr{W})$ *si y solo si* $-\boldsymbol{w}_c \in ri(\mathscr{T})$.
4. $-\boldsymbol{w}_c \in ri(\mathscr{T})$ *si y solo si* $p_{\mathscr{T}^*}(-\boldsymbol{w}_c) < 1$.

Demostración. El primer enunciado resulta de la congruencia de $\mathscr{W}$ y $\mathscr{T}$. El segundo inciso se obtiene por ser $aff(\mathscr{T})$ un subespacio vectorial, el cual es generado por $\mathbf{T}$ y así resulta que $dim(\mathscr{T}) = rango(\mathbf{T})$. El tercer inciso es consecuencia directa de la definición de $\mathscr{T}$ en (4.12). Finalmente, el teorema 1.3.10 indica que $p_{\mathscr{T}^*}(-\mathbf{w}_c)$ es finita si y solo si $-\mathbf{w}_c \in aff(\mathscr{T})$, y el teorema 1.3.15 asevera que si $-\mathbf{w}_c \in aff(\mathscr{T})$ entonces el último inciso se satisface. □

De la proposición anterior se deben hacer dos observaciones importantes. La primera es el hecho de que $p_{T^*}(-\mathbf{w}_c) < 1$ implica que $-\mathbf{w}_c \in ri(\mathscr{T})$ siempre y cuando se verifique primero que $-\mathbf{w}_c \in aff(\mathscr{T})$. La segunda es que si la dimensión de $\mathscr{W}$ sea g implica que $rango(\mathbf{G}) = g$, pero no necesariamente lo inverso. Esto se debe a que la dimensión de $\mathscr{W}$ es igual al rango de $\mathbf{T}$ el cual puede ser menor que el rango de $\mathbf{W}$ si la dimensión de $\mathscr{W}$ no es completa.

La proposición 4.2.1, demuestra que las propiedades de interés de $\mathscr{W}$ son igualmente determinadas por $\mathscr{T}$, más aun se tiene una relación directa entre la pertenencia de $-\mathbf{w}_c$ al interior relativo de $\mathscr{T}$ y los valores de la función de soporte. La ventaja de trabajar con $\mathscr{T}$ es el hecho de que su espacio afín sea un espacio vectorial. Además, se puede concluir que el problema de determinar si $-\mathbf{w}_c$ es una combinación afín de $\mathscr{T}$ es equivalente a que un sistema lineal de ecuaciones sea consistente. La siguiente proposición describe esta propiedad.

Proposición 4.2.2. $-\mathbf{w}_c \in aff(\mathscr{T})$ *si y solo si el sistema lineal*

$$\mathbf{Tz} = -\mathbf{w}_c \tag{4.15}$$

es consistente. Más aun, la consistencia de este sistema es equivalente a $rango(\boldsymbol{T}) = rango(\boldsymbol{T}')$, *donde* $\boldsymbol{T}'$ *es la matriz aumentada de* $\boldsymbol{T}$.

Demostración. Como $aff(\mathscr{T})$ es un subespacio vectorial generado por $\mathbf{T}$, un vector $\mathbf{y} \in aff(\mathscr{T})$ se puede expresar como combinación lineal de las columnas de $\mathbf{T}$ si el sistema $\mathbf{Tx} = \mathbf{y}$ tiene solución. Luego, $-\mathbf{w}_c \in aff(\mathscr{T})$ si y solo si el sistema $\mathbf{Tz} = -\mathbf{w}_c$ es consistente.

Ahora, el teorema de Rouché-Capelli de álgebra lineal [53] dice que un sistema de ecuaciones lineales es consistente si y solo si el rango de la matriz asociada al sistema es igual al rango de la matriz aumentada, lo que concluye la demostración de este teorema. □

Entonces una sujeción robótica es *force-closure* si y solo si $dim(\mathscr{W}) = g$ y $\mathbf{0} \in ri(\mathscr{W})$, lo cual es equivalente a que $rango(\mathbf{T}) = g$ y $p_{T^*}(-\mathbf{w}_c) < 1$ siempre y cuando $\mathbf{Tz} = -\mathbf{w}_c$ sea consistente. Por otra parte, del teorema 1.3.10 se sabe que la función soporte de T^* es finita por lo que

$$p_{T^*}(\mathbf{z}) = \sup_{\mathbf{x} \in T^*} \mathbf{z}^T \mathbf{x} = \operatorname*{máx}_{\mathbf{x} \in T^*} \mathbf{z}^T \mathbf{x}.$$

De la ecuación anterior y la definición de polaridad, la evaluación de $p_{T^*}(-\mathbf{w}_c)$ se traduce en resolver el problema de PL

$$\begin{gathered} \operatorname{máx} -\mathbf{w}_c^T \mathbf{x} \\ \text{sujeto a} \\ (\mathbf{w}_{ij} - \mathbf{w}_c)^T \mathbf{x} \leq 1, \quad i \in [n];\ j \in [m], \end{gathered} \tag{4.16}$$

Por lo tanto una sujeción es *force-closure* si $rango(\mathbf{T}) = g$, $rango(\mathbf{T}) = rango(\mathbf{T}')$ y el valor óptimo del sistema (4.16) es estrictamente menor que 1. En el diagrama presentado en la figura 4.8 se expone una síntesis de las equivalencias presentadas en esta sección.

$$\begin{tikzcd}
 & & & & \mathbf{Tz} = -\mathbf{w}_c \text{ es consistente} \arrow[d, leftrightarrow] \\
 & & \mathbf{0} \in ri(\mathscr{W}) \arrow[r, leftrightarrow] & -\mathbf{w}_c \in ri(\mathscr{W}) \arrow[r, leftrightarrow] & p_{\mathscr{T}^*}(-\mathbf{w}_c) < 1 \arrow[r] & -\mathbf{w}_c \in aff(\mathscr{T}) \\
force\text{-}closure \arrow[r, leftrightarrow] & \mathbf{0} \in int(\mathscr{W}) \arrow[ur, leftrightarrow] \arrow[dr, leftrightarrow] & & & & \\
 & & dim(\mathscr{W}) = g \arrow[r, leftrightarrow] & dim(\mathscr{T}) = g \arrow[r, leftrightarrow] & rango(\mathbf{T}) = g \arrow[r] & rango(\mathbf{W}) = g
\end{tikzcd}$$

Figura 4.8: Equivalencias para obtener la propiedad de *force-closure*.

Dada la clasificación de $\mathscr{W}$ al principio de la sección 4.2.2, el análisis anterior puede arrojar alguno de los siguientes casos:

a) $rango(\mathbf{T}) < g$ y $rango(\mathbf{T}) \neq rango(\mathbf{T}')$.

b) $rango(\mathbf{T}) < g$ y $rango(\mathbf{T}) = rango(\mathbf{T}')$.

c) $rango(\mathbf{T}) = g$ y $p_{\mathscr{T}^*}(-\mathbf{w}_c) \geq 1$.

d) $rango(\mathbf{T}) = g$ y $p_{\mathscr{T}^*}(-\mathbf{w}_c) < 1$.

Claramente, si el análisis de la sujeción resulta en el caso *d* la sujeción es *force-closure*. En el caso de obtener como resultado *b)*, sucede que la sujeción satisface la propiedad de *force-closure parcial*, la cual determina la propiedad de *force-closure* en un espacio de dimensión menor que la del espacio de las fuerzas

generalizadas [60]. Los casos *a)* y *c)* concluyen que la sujeción no presenta la condición de *force-closure*.

Ya que el trabajo de Zheng y Qian se basa en la propuesta de Liu y Wang, se pueden obtener los mismos resultados de los últimos utilizando las conclusiones de esta sección. Sea $R_{\mathbf{w}_c}$ el rayo que nace en $\mathbf{w}_c$ y pasa por el origen $\mathbf{0}$

$$R_{\mathbf{w}_c} = \{-\beta \mathbf{w}_c + \mathbf{w}_c : \beta \geq 0\},$$

donde $\mathbf{w}_c \neq 0$. Si $-\mathbf{w}_c \in aff(\mathcal{T})$ entonces $R_{\mathbf{w}_c}$ también pertenece a $aff(\mathcal{T})$ y el teorema 1.3.10 a su vez garantiza que $p_{\mathcal{T}^*}(-\mathbf{w}_c)$ es finito. Así, como $\mathcal{T}^*$ es no vacío, el problema de programación lineal (4.16) siempre tiene una solución y puede ser resuelto en tiempo $O(mn)$ [60]. Por lo que dado un punto interior y el rayo que emana de él, es posible resolver el problema de ray-shooting sobre $\mathcal{T}^*$ y por polaridad encontrar el punto q de intersección entre el rayo y $\mathcal{T}$.

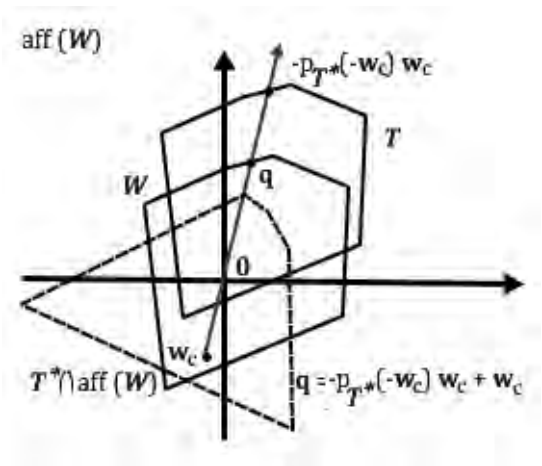


Figura 4.9: Interpretación del método de Zheng y Qian como un problema de ray-shooting.

Más aún, el teorema 1.3.13 indica que $R_{\mathbf{w}_c}$ intersecta la frontera de $\mathcal{T}$ en el punto $-\frac{\mathbf{w}_c}{p_{\mathcal{T}^*}(-\mathbf{w}_c)}$. Entonces la intersección de $R_{\mathbf{w}_c}$ y la frontera de $\mathcal{W}$ ocurre en el punto $\mathbf{q} = -\frac{\mathbf{w}_c}{p_{\mathcal{T}^*}(-\mathbf{w}_c)} + \mathbf{w}_c$. En la figura 4.9 se presenta una ilustración del análisis anterior.

Considerando las distancias en el espacio euclidiano $\mathbb{R}^g$, $d(\mathbf{w}_c, \mathbf{0}) = ||\mathbf{w}_c||$ y $d(\mathbf{w}_c, \mathbf{q}) = ||\mathbf{w}_c - (-\frac{\mathbf{w}_c}{p_{\mathcal{T}^*}(-\mathbf{w}_c)} + \mathbf{w}_c)||$ se tiene que

$$\frac{d(\mathbf{w}_c, \mathbf{0})}{d(\mathbf{w}_c, \mathbf{q})} = \frac{||\mathbf{w}_c||}{||\frac{\mathbf{w}_c}{p_{\mathcal{T}^*}(-\mathbf{w}_c)}||} = \frac{||\mathbf{w}_c||}{\frac{1}{p_{\mathcal{T}^*}(-\mathbf{w}_c)}||\mathbf{w}_c||} = p_{\mathcal{T}^*}(-\mathbf{w}_c).$$

Esto implica que $p_{\mathcal{T}^*}(-\mathbf{w}_c)$ expresa la razón entre la distancia entre $\mathbf{w}_c$ y $\mathbf{0}$ y la distancia de $\mathbf{w}_c$ y el punto de intersección. Por lo tanto, si $p_{\mathcal{T}^*}(-\mathbf{w}_c) < 1$ significa que $d(\mathbf{w}_c, \mathbf{0}) < d(\mathbf{w}_c, \mathbf{q})$ y así $\mathbf{0} \in ri(\mathcal{W})$ o bien, la sujeción es de tipo *force-closure*.

Esta interpretación geométrica de la función de soporte evaluada en el centroide ayuda a eliminar el cálculo de las distancias mencionadas y por lo tanto simplificar el algoritmo de Liu y Wang.

A partir de las relaciones encontradas en la sección anterior se puede desarrollar un algoritmo para determinar si una sujeción es *force-closure*. La validación del mismo está determinada por las demostraciones vistas en esta sección.

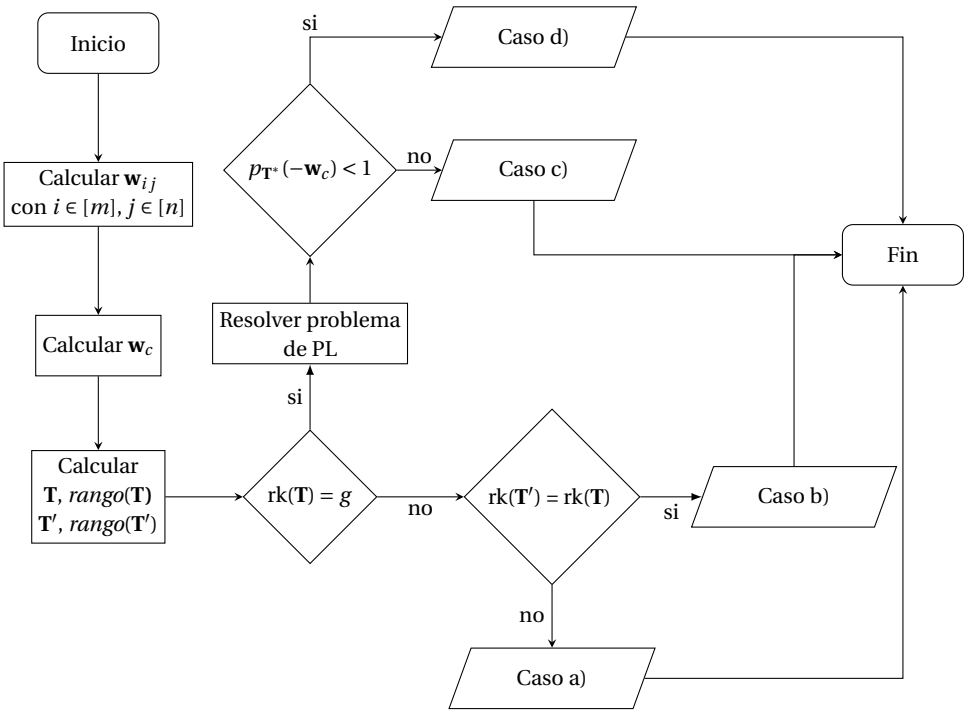


Figura 4.10: Diagrama de flujo para el algoritmo cualitativo de Zheng y Qian.

Así, dado un objeto sujetado con n dedos, los puntos de contacto $\mathbf{r}_i$ de los dedos respecto al sistema de referencia local y los vectores normales en cada uno de estos puntos, se inicia el algoritmo calculando las fuerzas primitivas generalizadas. Posteriormente, se calcula en centroide y la matriz de vectores de la traslación de $\mathscr{W}$ por $\mathbf{w}_c$, y se determinan los rangos de $\mathbf{T}$, $\mathbf{T}'$. Si $rango(\mathbf{T}) \neq rango(\mathbf{T}')$ entonces La figura 4.10 contiene un diagrama de flujo con el algoritmo cualitativo de Zheng y Qian para *force-closure*.

Capítulo 5

Resultados e Implementación

En este capítulo se presentan los resultados obtenidos de implementar el algoritmo de Megiddo en la prueba cualitativa de Zheng y Qian para determinar la propiedad de *force-closure* en distintas sujeciones robóticas. Además, se presenta una evaluación del algoritmo de Megiddo en tiempo lineal para problemas de PL surgido de su codificación.

Como se mencionó en la sección 2.2.3, tanto la teoría como el desarrollo práctico del algoritmo de Megiddo se complica conforme la dimensión del problema aumenta. Para problemas de sujeción robótica en $\mathbb{R}^3$ el espacio de trabajo es $\mathbb{R}^6$, por lo que la constante de proporcionalidad del algoritmo es del orden de 10^{19}; más aún, la dificultad de la construcción del algoritmo para problemas de tales dimensiones es tan compleja que no se considera un trabajo pertinente para los propósitos de este proyecto. Por lo tanto, se decidió trabajar con sujeciones en el plano. Esto significa que el espacio de trabajo (el espacio de las fuerzas generalizadas) es de dimensión tres y por lo tanto se puede implementar el algoritmo descrito a lo largo de estas páginas.

Puesto que no se ha encontrado una implementación pública del algoritmo de Megiddo, se corroboraron los resultados obtenidos usando la paquetería de *Mathematica* para resolución de problemas de PL. Se seleccionó el método simplex para comparar los tiempos de corrida de la implementación del algortimo. Cabe mencionar que durante el proceso de desarrollo del código, se encontraron inconsistencias, ambigüedades y errores en el artículo de Megiddo, algunos de los cuales fueron corregidos y otros simplemente se señalarán a continuación. Con esto en mente, de ninguna manera se considera la implementación desarrollada como un código óptimo y depurado, se realizó la mejor versión posible.

5.1. Resultados

Se consideró como objeto a sujetar un cuadrado con centro de masa en el origen y lados de magnitud uno, paralelos a los ejes. Se implementaron 21 con-

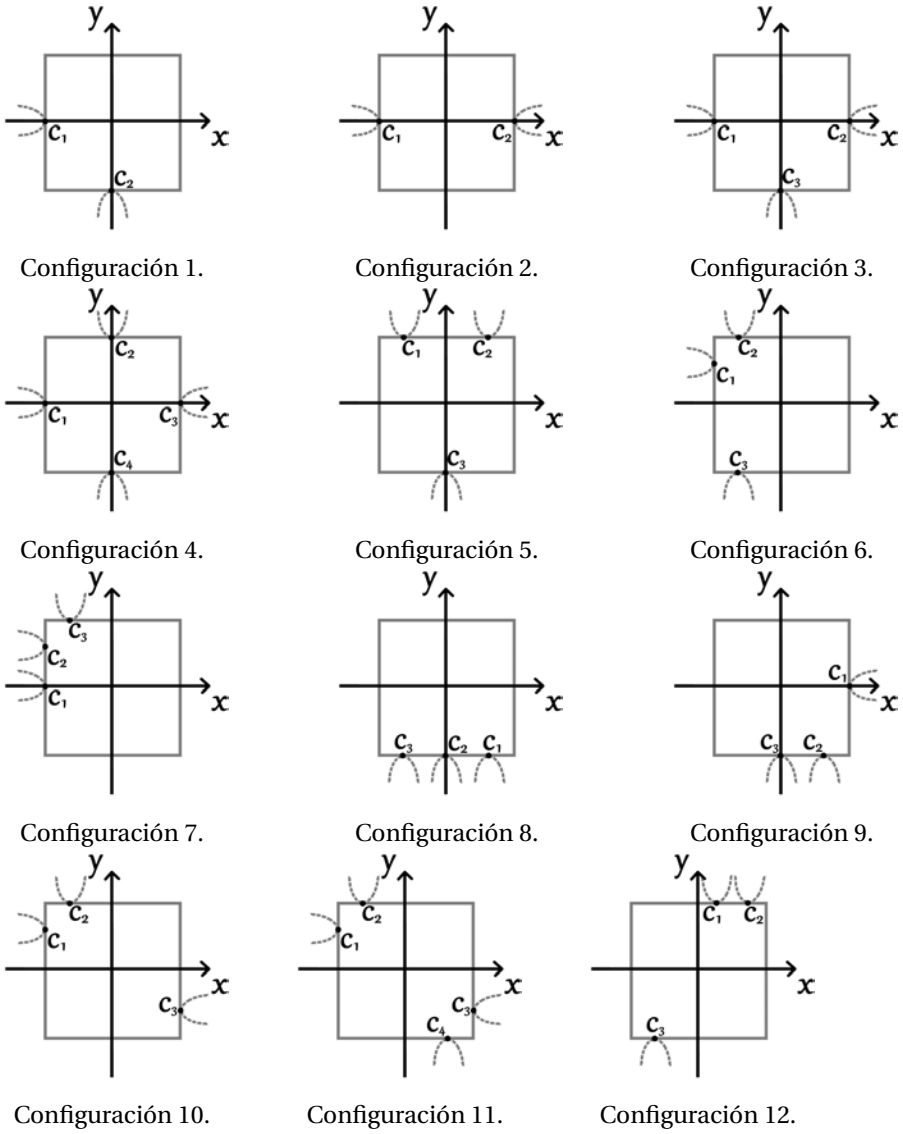
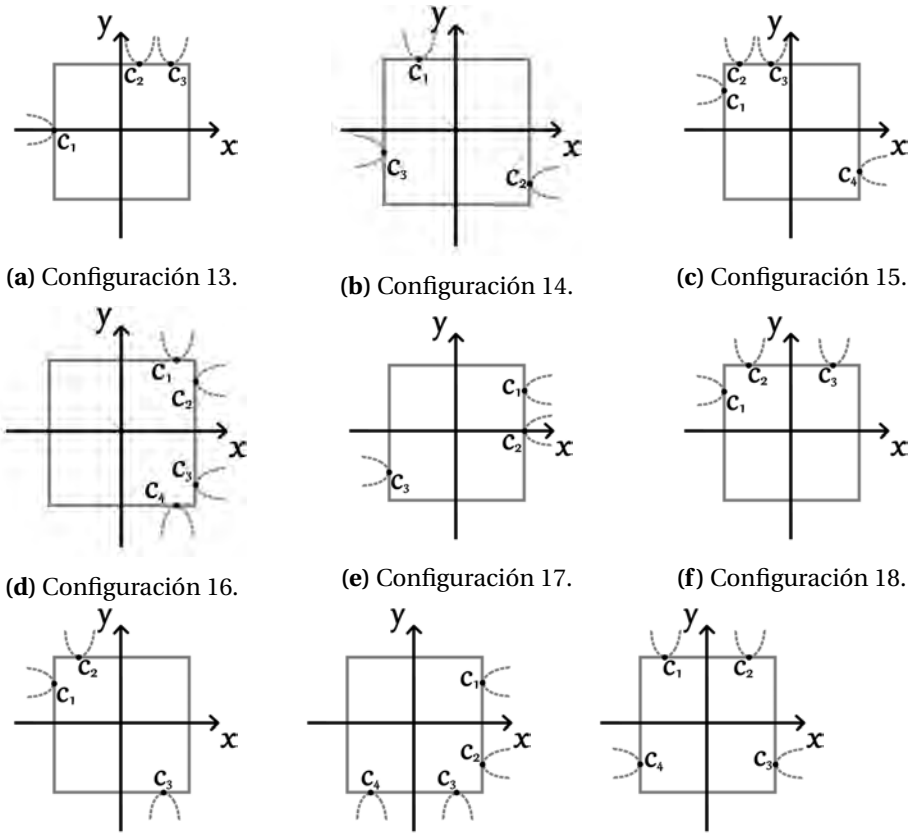


Figura 5.1: Ilustración de las distintas configuraciones de sujeción probadas.

figuraciones básicas de sujeciones entre 2 y 4 dedos. Además, se verificaron que las rotaciones dextrógiras de 90 grados de cada una coincidieran con los resultados de la configuración inicial; para una configuración x, la rotación por 90 grados se denotará por $x.1$, para la rotación de 180 grados por $x.2$ y las rotaciones por 270 grados como $x.3$. En las figuras 5.1 y 5.2 se presentan las 21 configuraciones básicas. Nótese que la configuración 2 sólo tiene dos rotaciones de interés, la original y la que resulta de girar la figura 90 grados; en la configuración 4 no se considera ninguna rotación. Lo anterior resulta en un total de 79 configuraciones de sujeciones que fueron evaluadas para determinar si cuen-

tan con la propiedad de *force-closure*. Posteriormente, se realizaron pequeñas modificaciones en algunas configuraciones para comparar el comportamiento de la prueba cualitativa y la intuición que dan las imágenes.



(a) Configuración 13. **(b)** Configuración 14. **(c)** Configuración 15.

(d) Configuración 16. **(e)** Configuración 17. **(f)** Configuración 18.

(g) Configuración 19. **(h)** Configuración 20. **(i)** Configuración 21.

Figura 5.2: Ilustración de las diferentes configuraciones de sujeción probadas.

Las configuraciones 2, 4 y 11 resultan ser *force-closure* de manera directa. Esto se debe a que el rango de $\mathbf{T}$ es de dimensión completa y el centroide $\mathbf{w}_c$ es el origen de $\mathbb{R}^3$; esto implica que no hay necesidad de resolver el problema de PL. Por otra parte la configuración 8 no es *force-closure* ya que el sistema $\mathbf{Tx} = -\mathbf{w}_c$ no es consistente.

Para el resto de las configuraciones se implementó el algoritmo de Megiddo para determinar el valor de la función de soporte $p_{\mathcal{T}^*}(-\mathbf{w}_c)$ descrita en (4.13). En la tabla 5.1 se presentan los resultados de la implementación de la prueba cualitativa de Zheng y Qian; en ésta se indica el valor de la función de soporte, la respuesta a la prueba cualitativa (si la sujeción es *force-closure*), y los tiempos de ejecución tanto del algoritmo de Megiddo como de Simplex para las configuraciones que corresponda. La última columna indica en qué iteración y en cuál de las pruebas (pruebas a las rectas $x = x_m$ y $y = y_m$) el algoritmo de Megiddo

Configuración	$p(-\mathbf{w}_c)$	Force Closure	Tiempo Megiddo	Tiempo Simplex	Instancia de conclusión
1	4	No	0.000628	0.000144	Iteración 1, directo
3	0.5714	Si	0.00238	0.000155	Iteración 1, prueba 1
5	0.25	Si	0.00151	0.000185	Iteración 1, prueba 1
6	0.6666	Si	0.00713	0.000187	Iteración 3, directo
7	4	No	0.00738	0.000222	Iteración 3, directo
9	4	No	0.00151	0.000206	Iteración 3, prueba 1
10	2.08510	No	0.00793	0.000172	Iteración 3, directo
12	2	No	0.00151	0.000166	Iteración 3, directo
13	4	No	0.0065	0.000226	Iteración 3, directo
14	1	No	0.0077	0.000218	Iteración 3, directo
15	1.70833	No	0.01523	0.000166	Iteración 5, directo
16	0.6666	Si	0.002301	0.000163	Iteración 1, prueba 1
17	1	No	0.00257	0.000625	Iteración 3, directo
18	4	No	0.00679	0.00016	Iteración 3, directo
19	2.654	No	0.007792	0.000117	Iteración 3, directo
20	4	No	0.0117	0.000167	Iteración 4, prueba 1
21	0.6666	Si	0.0020	0.000225	Iteración 1, prueba 1

Cuadro 5.1: Resultados de las corridas

obtuvo la solución óptima; el concepto de directo se refiere a que en alguna iteración el número de restricciones fue suficientemente pequeño y el problema se resolvió de manera directa. Para el caso del algoritmo de Megiddo en el espacio, se consideró que aquella iteración en donde intersectar planos resulte en solamente una recta, es en la que se procede a resolver el problema de manera directa.

Puesto que los problemas propuestos son de tamaño $d \times 3$ con $d = \{4, 6, 8\}$, el tiempo de ejecución está entre los órdenes de magnitud de 10^{-4} y 10^{-2} segundos para Megiddo, mientras que simplex es más rápido corriendo siempre en un tiempo del orden de 10^{-4} segundos. Naturalmente, el algoritmo fue más rápido en tiempo de cómputo en las configuraciones con menos dedos. En la mayor parte de los casos, el algoritmo de Megiddo realizó un número de iteraciones necesarias hasta llegar a la cantidad mínima de restricciones para poder buscar la solución de manera directa.

De las configuraciones básicas se presentaron inconsistencias para los problemas 18.1, 18.3. El algoritmo de Megiddo determina que el origen es no factible y que además, minimiza el valor de la función objetivo, lo que determina que el problema no tiene solución. En contraste las rotaciones 18.2 y 18.4 proceden hasta encontrar una cantidad de restricciones suficientemente pequeña y encuentra así de manera satisfactoria la solución óptima factible. A pesar de estar identificado el problema aún no se ha logrado descubrir si el error yace en

el código o en la formulación de Megiddo.

Se observa que las configuraciones 3, 5, 6, 16 y 21 presentan la cualidad de *force-closure*, lo que implica que éstas son capaces de neutralizar cualquier fuerza externa (realista) que se aplique al objeto. Recordando que si el valor de la función de soporte está en el intervalo $(0,1)$ la sujeción es *force-closure* y mientras más chico sea su valor mejor es la calidad de la configuración, se concluye que de la muestra de sujeciones *force-closure*, la configuración 5 es la mejor con un valor de 0.25. Por otra parte, las configuraciones 1, 7, 9, 10, 12, 13, 14, 15, 17, 18, 19 y 20 no conservan la propiedad. De éstas las configuraciones 14 y 17 presentan el valor 1 en la función de soporte, lo que significa que realizando una pequeña modificación en la configuración, la propiedad de *force-closure* se puede llegar a alcanzar. Estos resultados coinciden con la intuición visual proporcionada por las imágenes en las figuras 5.1 y 5.2.

Para poder observar el comportamiento de la función de soporte se hicieron pequeñas variaciones en algunas configuraciones para mostrar cómo estos pequeños cambios modifican la calidad de la sujeción y cómo esto se ve reflejado en el valor de la función de soporte. La configuración 14 tiene puntos de contacto $c_1 = \left(-\frac{1}{2}, 1\right), c_2 = \left(1, -\frac{3}{4}\right), c_3 = \left(-1, -\frac{1}{4}\right)$, si al cambiar el punto de contacto c_2 a $\left(1, -\frac{2}{3}\right)$ la función de soporte tiene un valor de 0.81481, lo cual trasforma la configuración a una *force-closure*; de manera semejante, la configuración 17 tiene como puntos de contacto $c_1 = \left(1, \frac{1}{2}\right), c_2 = (1,0), c_3 = \left(-1, -\frac{1}{2}\right)$, si se modifica el segundo punto de contacto al $c_2 = \left(1, -\frac{1}{4}\right)$ entonces el valor óptimo de la función objetivo es 0.78571 y nuevamente se ha encontrado una mejoría en la configuración de la sujeción.

5.2. Implementación

La implementación de la prueba cualitativa de Zheng y Qian para determinar si una sujeción es *force-closure* no tuvo mayores complicaciones. Por otra parte la implementación del algoritmo de Megiddo, como se ha mencionado anteriormente, resultó complicada y requirió un riguroso análisis del algoritmo presentado en los artículos [30] y [31]. A partir de esto se encontraron errores o ambigüedades que se tuvieron que resolver; de éstos algunos merecen la pena discutirse y se presentan a continuación.

Es importante mencionar que el algoritmo se basa mayormente en el cálculo de la mediana de distintos elementos (ángulos, coordenadas de puntos), por lo que para mantener la linealidad del algoritmo es necesario que este proceso también lo sea. Para calcular la mediana de un conjunto L de t números, un enfoque es ordenar los elementos y finalmente seleccionar el elemento que corresponda a la posición $\frac{t}{2}$. El simple hecho de ordenar un conjunto requiere tiempo $O(n \log n)$, por lo que Megiddo propone utilizar un algoritmo lineal que simplemente aproxime la mediana. Este algoritmo es llamado mediana de medianas y se describe en el apéndice B. Se debe mencionar que el cálculo de la mediana

para los ángulos de las rectas $L_1, ..., L_k$ en el algoritmo para $\mathbb{R}^3$ no está bien definida. Esto se debe a que una recta de la forma $ax + b = 0$ tiene ángulo $-\frac{\pi}{2}$ o bien $\frac{\pi}{2}$, lo cual puede modificar la mediana del conjunto de ángulos, resultando en errores al momento de transformar el sistema coordenado.

Una observación significativa para el algoritmo es que al resolver el problema de manera directa la solución óptima puede estar definida por cualquier subconjunto de restricciones, independientemente del conjunto de índices al que pertenezcan. Sin embargo, es importante notar que para problemas en el plano, la solución óptima puede estar definida por alguno de los extremos del intervalo de soluciones $[d_1, d_2]$. En este caso, el algoritmo de resolución directa calcula todos los puntos de intersección posibles entre las restricciones restantes y añade d_1 y d_2 al conjunto de posibles soluciones; posteriormente analiza cuáles de ellas son soluciones factibles y selecciona la óptima (si ésta existe).

Por otra parte, en el algoritmo para problemas de PL en $\mathbb{R}^3$, las restricciones en I_3 aportan información diferente que las correspondientes a I_1 y I_2. Ya que las restricciones pertenecientes a I_3 son de la forma $ax + b + y + c = 0$, éstas son restricciones paralelas al eje z, y al ser proyectadas sobre el plano xy, se obtiene una recta, a diferencia de las otras restricciones en las que se obtiene todo el espacio. Por eso, al momento de intersectar los planos correspondientes a las restricciones, sólo se debe tomar en cuenta las que pertenezcan a los conjuntos I_1 e I_2, ya que la intersección de éstos, proyectada en el plano resulta en una recta. Las restricciones en I_3 simplemente deben omitir este paso.

Después, a partir de la proposición 2.2.4 se construye el tercer problema auxiliar para la prueba de una recta en el plano, el cual determina si la existencia de una solución factible que mejore la función objetivo pertenece al semiplano inferior. En el artículo [30], Megiddo declara que esto es equivalente a un problema de minimización, sin embargo, esto es un error por lo siguiente. De la proposición mencionada, se define una variable auxiliar η tal que $\min\{a_i\lambda + b_i : i \in I_1^*\} \geq \eta \geq \max\{a_i\lambda + b_i : i \in I_2^*\}$ y las condiciones $\min\{a_i\lambda + b_i : i \in I_1^*\} > 0$, y $\min\{a_i\lambda + b_i : i \in I_3^*\} \geq 0$ sean satisfechas. Al fijarse en el valor mínimo que puede alcanzar η se tiene que es $\max\{a_i\lambda + b_i : i \in I_2^*\}$, no obstante, como se menciona al principio de esa sección el conjunto I_2^* puede ser vacío, lo que puede resultar en un problema no acotado y la obtención de falsos óptimos. Al considerar el valor máximo que puede tomar η, $\min\{a_i\lambda + b_i : i \in I_1^*\}$, la misma proposición asegura que si este valor es positivo entonces es el semiplano inferior el de interés.

Finalmente se debe mencionar que, aunque la construcción teórica del algoritmo para bajas dimensiones resulta elegante e ingeniosa, la implementación puede ser bastante larga e intrincada, por lo que si no se tiene especial cuidado la complejidad del algoritmo puede aumentar.

Capítulo 6

Conclusiones

El propósito de este trabajo es realizar un estudio detallado del algoritmo lineal de Megiddo para resolver problemas de programación lineal en dimensiones fijas bajas y probar su eficiencia práctica. Para esto se decidió elegir un problema de relevancia actual como lo es el análisis de sujeciones robóticas.

El algoritmo de Megiddo para problemas de dimensión baja nació como una inquietud por tener un algoritmo lineal en el número de restricciones y concluyó como un algoritmo teóricamente elegante e intuitivo, utilizando herramientas básicas de geometría computacional. El ingenioso uso de la mediana para ir eliminando restricciones, y la manera de ir transformando un problema en varios problemas de dimensión menor, refuerzan la perspicacia con que fue ideado este algoritmo. Ésto, aunado al seguimiento que se le dio por una corriente de seguidores, causó interés ya que tenía una interesante base teórica y motivaba su implementación práctica. Por su parte, la modelación de un problema de calidad de una sujeción robótica como un problema de programación lineal resulta de un refinado estudio de teoría de poliedros, resultando en un problema de dimensión fija y una cantidad de restricciones moderada, lo cual lo hace atractivo para probar el algoritmo de Megiddo.

Al iniciar el estudio del algoritmo se determinó que su construcción teórica y práctica para dimensiones mayores que tres era bastante elaborada y a pesar de ser lineal sobre el número de restricciones, el algoritmo es mucho más lento que distintos algoritmos ya existentes y por lo tanto, no era una labor que correspondiera al presente trabajo. Sin embargo, su implementación para problemas de dimensión dos y tres es bastante intuitiva y relativamente rápida, así que se decidió analizar el algoritmo de Megiddo aplicado a la resolución de pruebas cualitativas de sujeciones robóticas en el plano para los cuales su espacio de trabajo es $\mathbb{R}^3$.

Al progresar el análisis del artículo [30] donde se presentaba por primera vez el algoritmo de Megiddo se encontró que las ideas del algoritmo no estaban completamente desarrolladas y se requirió el estudio de artículos posteriores sobre el algoritmo tanto de Megiddo como de sus seguidores. Sin embargo, la

teoría detrás de éste es sumamente elegante y se puede procesar desde distintas perspectivas, por lo cual se entiende la curiosidad que generó. No obstante, se encontró que el contenido teórico muchas veces era faltante, por lo que ocurrían errores en la comprensión debido a ambigüedades o vaguedades sobre ciertas instancias. Más aún, se detectaron errores en la construcción de subproblemas y el manejo de algunos conjuntos de restricciones, errores a los que no se encontró mención en ningún artículo posterior, por lo que se pone en duda la existencia de alguna implementación práctica anterior del algoritmo.

A pesar de que las múltiples instancias que constituyen cada uno de los algoritmos es sencilla, el resultado es un código con muchísimas subrutinas, por lo que se debe tener sumo cuidado en mantener el tiempo de corrida lineal en cada una de ellas, arriesgando aumentar la complejidad del algoritmo. Una observación es que en el artículo [31] Megiddo menciona que el algoritmo se puede llevar a cabo sin la transformación inicial la formulación de Megiddo. A pesar de este comentario, se considera que esta transformación inicial es indispensable para el resto del desarrollo del algoritmo ya que todo está construido a partir de ésta. No obstante, a lo largo del algoritmo se realizan una cantidad considerable de transformaciones lineales al problema, lo que puede acarrear errores de redondeo y causar problemas desde el punto de vista de análisis numérico. Una posible mejora al algoritmo consistiría en eliminar algunas de estas transformaciones ya que a pesar de no perjudicar la linealidad del algoritmo, puede crear problemas de carácter numérico.

Por su parte, la caraterización de una sujeción *force-closure* como un problema de programación lineal es uno de los distintos enfoques que se han desarrollado para resolver este tipo de problemas, y la propuesta de Zheng y Qian se considera una teoría bastante robusta y concisa. La formulación y conceptos desarrollados por ellos, se pueden utilizar no solo para determinar la propiedad de *force-closure*, sino para optimizar las fuerzas de contacto que son aplicadas por una configuración *force-closure* al objeto, problema de gran importancia por el ahorro de energía que representa. En virtud de lo anterior, en este trabajo se considera muy interesante realizar investigaciones en esta área que incluyen la búsqueda de algoritmos específicos para este tipo de problemas donde la dimension es fija y baja (por ejemplo [35]).

Apéndice A

Topología de Espacios Afines

El contenido de este apéndice se basa en los capítulos 2 y 3 del libro de topología de James Munkres [38].

Se define una *bola abierta* centrada en $\mathbf{a}$ con radio r como todos los puntos cuya distancia a $\mathbf{a}$ es menor que r, i.e.,

$$B(\mathbf{a}, r) = \{\mathbf{x} \in \mathbb{R}^d \,:\, ||\mathbf{x} - \mathbf{a}|| < r\},$$

donde $\mathbf{a} \in \mathbb{R}^d, r \in \mathbb{R}^+$. Por comodidad se hará referencia a la bola abierta $B(\mathbf{a}, r)$ como bola. De forma semejante, se puede definir una *bola cerrada* con centro en $\mathbf{a}$ y radio r, $\overline{B(\mathbf{a}, r)}$, como todos los puntos cuya distancia a $\mathbf{a}$ no es mayor que r, o bien

$$\overline{B(\mathbf{a}, r)} = \{x \in \mathbb{R}^d \,:\, ||\mathbf{x} - \mathbf{a}|| \leq r\}.$$

Así, se puede definir un conjunto $A \subseteq \mathbb{R}^d$ *abierto* como aquel conjunto tal que para todo $\mathbf{x} \in A$ existe una bola $B(\mathbf{a}, r)$ que esté totalmente contenida en A. Por su parte, un subconjunto A de $\mathbb{R}^d$ se dice que es *cerrado* si su complemento es abierto. Una propiedad muy importante de estos conjuntos es que la intersección arbitraria de conjuntos es cerrado. Un conjunto es *acotado* si existe un valor $K \in \mathbb{R}$ tal que $||\mathbf{x}|| \leq K$ para todo $\mathbf{x}$ en el conjunto, es decir, si existe una bola $B(\mathbf{x}_k, K)$ tal que $S \subset B(\mathbf{x}_k, K)$ para algún $\mathbf{x}_k \in \mathbb{R}^d$. Para los propósitos de este trabajo, se define como conjunto *compacto* a un conjunto que sea cerrado y acotado. De lo anterior se concluye que una bola abierta es un conjunto abierto y una bola cerrada es un conjunto compacto.

El *interior* de un conjunto S, $int(S)$, se define como la unión de todas las bolas abiertas contenidas en S. Puesto que la unión de abiertos es abierto, el interior es un conjunto abierto, de hecho, el abierto más grande contenido en S. De manera complementaria, la *cerradura* de un conjunto S se define como la intersección de todos los conjuntos cerrados que contienen a S y se denota por $cl(S)$. La *frontera* de un conjunto S, $b(S)$ se puede expresar como $b(S) = cl(S) \backslash int(S)$.

En optimización el concepto de interior es muy importante, pero al trabajar en conjuntos convexos en $\mathbb{R}^d$ con dimensión menor que d, esta noción puede

perderse. Por lo tanto, es de interés estudiar esta propiedad en el “espacio más pequeño"que contenga al conjunto. Este espacio es la envoltura afín del conjunto. Así, dado $S \subseteq \mathbb{R}^d$ y $\mathbf{x} \in S$ se dice que $\mathbf{x}$ es un *punto interior relativo* si existe un radio $r > \mathbf{0}$ tal que

$$B(\mathbf{x}, r) \cap aff(S) \subseteq S.$$

Entonces, el *interior relativo* de S , $ri(S)$, es el conjunto de todos los puntos interiores de S. En otras palabras, el interior relativo de $S \subseteq \mathbb{R}^d$ es el interior de S que resulta de considerar S como subconjunto de su envoltura afín, es decir, es el interior de S en relación con $aff(S)$. Nótese que el concepto de interior está relacionado con la dimensión. Finalmente, se define la *frontera relativa* $rb(S)$ como los puntos de la cerradura de S que no están en el interior relativo, es decir, $rb(S) = cl(S) \backslash ri(S)$. Es importante observar que cuando $aff(S) \cong \mathbb{R}^d$ entonces $ri(S) = int(S)$, lo cual se demuestra a continuación.

Teorema A.0.1. *Dado $S \subseteq \mathbb{R}^n$, $\mathbf{0} \in int(S)$ si y solo si $dim(S) = n$ y $\mathbf{0} \in ri(S)$.*

Demostración. Si $\mathbf{0} \in int(S)$ entonces existe un radio $r > 0$ tal que $B(\mathbf{0}, r) \subseteq S$. Puesto que la dimensión de una bola es completa resulta que $dim(S) \geq n$. Sin embargo, $S \subseteq \mathbb{R}^n$ por lo que $dim(S) = n$. Por definición $dim(S) = dim(aff(S))$, así que la dimensión del espacio afín es n, lo que implica$aff(S) = \mathbb{R}^n$. Se sigue

$$B(\mathbf{0}, r) \cap aff(S) \subseteq B(\mathbf{0}, r) \subseteq S.$$

Luego, $\mathbf{0} \in ri(S)$ y $dim(S) = n$. Inversamente, si $dim(S) = dim(aff(S)) = n$ entonces $aff(S) = \mathbb{R}^n$ lo que implica $ri(S) = int(S)$. Así $\mathbf{0} \in ri(S)$ implica $\mathbf{0} \in int(S)$. □

Apéndice B

Algoritmo mediana de medianas

El algoritmo mediana de medianas está basado en el algoritmo *quickselect* [1]. El algoritmo *quickselect* es un algoritmo tipo *divide and conquer* (divide y vencerás) el cual encuentra el k-ésimo elemento más pequeño de una lista; este algoritmo tiene una complejidad computacional promedio de $O(n)$ pudiendo empeorar hasta ser cuadrático si el pivote con el que trabaja no está bien seleccionado. *Quickselect* funciona eligiendo un elemento pivote que divida L en dos conjuntos tal que uno contenga los elementos menores al pivote y el otro los elementos mayores; posteriormente compara la posición del pivote con k para proceder de manera recursiva en alguna de las sublistas o determinar si el pivote es el elemento deseado. Lo que hace el algoritmo de mediana de medianas es elegir de manera inteligente este pivote de tal forma que las sublistas tengan tamaños aproximados, por lo tanto el algoritmo inicia descomponiendo el conjunto L en subconjuntos de a lo más 5 elementos y calcula la mediana de cada subconjunto; posteriormente aplica de manera recursiva el algoritmo para encontrar la mediana del conjunto de medianas y ese elemento lo utiliza como pivote para *quickselect* obteniendo un algoritmo de complejidad $O(n)$.

Bibliografía

[1] Aho, A.V. ,Hopcroft, J. E. y Ullman, J. D., *The Design and Analysis of Computer Algorithms*, Addison-Wesley, Reading, MA, 1974.

[2] Al-Gallaf, E.M., *A Learning Rule-Based Robotics Hand Optimal Force Closure*, Second International Conference on Computational Intelligence, Communication Systems and Networks, 2010.

[3] Asada, H., *Studies on Prehension and Handling by Robot Hands With Elastic Fingers*, Tesis doctoral, Universidad de Kyoto, 1979.

[4] Ball, R.S., *A treatise on the theory of screws*, Cambridge University Press, Cambridge,00.

[5] Bicchi, A., *On the Closure Properties of Robotic Grasping*, Int. Journal of Robotics Research, 1993.

[6] Bicchi, A. y Kumar, V., *Robotic Grasping and Contact: A Review*, IEEE Int.Conf. on Robotics and Automation, Vol 1, 2000 pp. 348-353.

[7] Bjerhammar, A., *Application of calculus of matrices to method of least squares; with special references to geodetic calculations*, Transactions of the Royal Institute of Technology, no 48, 1951.

[8] Boyd, S. P. y Wegbreit, B., *Fast Computation of Optimal Contact Forces*, IEEE Transaction on Robotics, Vol 23(6), 2007, pp 1117-1132.

[9] Clarkson, K.L., *Linear programming in* $O(n3^{d^2})$ *time*. Information Processing Letters., Vol 2(2), 1986, pp 21–24.

[10] Dahl, G., *Introduction to Convex Analysis* University of Olso, 2010.

[11] Dantzig, G. B y Thapa, M. N., *Linear Programming 1: Introduction*, Springer-Verlag New York, 1997.

[12] Ding, D.; Liu, Y-H. y Wang, S., *Automatic Selection of Fixturing Surfaces and Fixturing Points for Polyhedral Workpieces*, IEEE Transactions on Robotics and Automation, Vol 17(6), 2001, pp. 833–841.

[13] Ding, D.; Liu, Y-H.; Zhang, J. y Knoll, A., *Computational Fingertip Positions for a Form-Closure Grasp*, IEEE Int.Conf. on Robotics and Automatons Vol 3, 2001, pp. 2217 - 2222.

[14] Dyer, M. E. *Linear Time Algorithms for two and three-variable linear programs*, SIAM Journal on Computing, Vol 13, 1984, pp 31-45.

[15] Dyer, M.E. *A class of convex programs with applications to computational geometry*, Proceedings of the 8th Annual Symposium on Computational Geometry, 1992, pp 9–15.

[16] Dyer, M.E.; Megiddo, N y Welzl, E. *Linear Programming*, The Handbook of Discrete and Computational Geometry, CRC Press, 1997, pp 699–710.

[17] Dyer, M.E *On a Multidimensional Search Problem and its Application to the Euclidean One-Centre Problem*, SIAM Journal on Computing, Vol 15, 1986, pp 725–738.

[18] El-Khoury, S. y Sahbani, A., *On Computing Robust N-Finger Force-Closure Grasps of 3D Objects*, IEEE International Conference on Robotics and Automation, 2009, pp 2480-2486 .

[19] Ferrari, C. y Canny, J., *Planning optimal grasps*, IEEE Interantional Conference on Robotics and Automation, Vol 3, 1992, pp. 2290–2295.

[20] Han, L., Trinkle, J. C. y Li, Z., *Grasp analysis as linear matrix inequality problems*, IEEE Transactions on Robotics and Automation, Vol 16(6), 2000, pp. 663–674.

[21] Hurtado, F., Sacristán, V. y Toussaint, G., *Prunning can solve from facility location to visibility problems*, Jornades Francoespanyoles de Geometria Algorítmica, 1997.

[22] S., Lang. *Algebra*, Springer, Yale University, 2002 .

[23] Kao, I., Lynch, K. y Burdick J.W., *Contact Modeling and Manipulation*, Springer Handbook of Robotics, Springer, 2008, pp 647-666.

[24] Khachian L.,G . *A polynomial algorithm in linear programming*, Soviet Math. Dokl. 20, pp 191-194, 1979.

[25] Karmarkar N. *A New Polynomial-Time Algorithm for Linear Programming*, Combinatorica 4, 373-395, 1984.

[26] León, B., Morales, A. y Sancho.Bru, J., *From Robot to Human Grasping Simulation*, Springer, 2014.

[27] Liu, Y-H., *Computing n-finger force-closure grasps on polygonal objects*, IEEE International Conferences on Robotics and Automation, Vol 3, 1998, pp 2734 - 2739.

[28] Liu, Y-H., Lam, M-L. y Ding, D., *A Complete and Efficient Algorithm for Searching* 3*D Form-Closure Grasps in the Discrete Domain*, IEEE Transactions on Robitcs and Automation, Vol 17(4), 2004, pp. 805–816.

[29] Liu, Y-H., y Wang, M., *Qualitative Test and Force Optimization of 3D Frictional Force-Closure Grasps Using Linear Programming*, Proceedings of the 1998 IEEE International Conference on Robotics and Automation, 1998.

[30] Megiddo, N., *Linear Time Algorithms for Linear Programming in* $\Re^3$ *and Related Problems*. SIAM J. Comput., Vol 12, pp. 759–776, 1983.

[31] Megiddo, N., *Linear Programming in Linear Time When Dimension is Fixed*. J. Assoc. Comput. Mach., Vol 31, pp. 114–127, 1984.

[32] Megiddo, N., *On the ball spanned by balls*. Discrete Comput. Geom., Vol4, pp. 605–610, 1989.

[33] Mishra, B., Schwartz, J.T. y Sharir, M., *On the Existence and Synthesis of Multifinger positive grips*, Algorithmica, Vol.2, No.4, 1987, pp. 541-558.

[34] Moore, E. H., *On the reciprocal of the general algebraic matrix*, Bulletin of the American Mathematical Society, Vol 26 (9), 1920, pp 394–395.

[35] Mosso-Vázquez, A., Juárez-Romero, D., Cruz-Chávez, M.A. y Sucar L.E., *A KKT Simplex Method for Efficiently Solving Linear Programs for Grasp Analysis Based on the Identification of Nonbinding Constraints*, Comp. y Sist. vol.18 no.2 México abr./jun. 2014

[36] Muller, D. E. y Preparata, F. P., *Finding the Intersection of Two Convex Polyhedra*, Theort. Comput. Sci., Vol 7, ,1978, pp. 217-236.

[37] Mulmuley, K., *Computational Geometry*, Prentice Hall, 1994.

[38] Munkres, *Topology*

[39] Murray, R.M., Li, Z y Sastry, S.S., *A Mathematical Introduction to Robotic Manipulation*, primera edición, CRC Press, 1994.

[40] Nakamura, Y., Nagai, K. y Yoshikawa, T., *Dynamics and Stability in Coordination of Multiple Robotic Systems*, The International Journal of Robotics Research, Vol 8(2), 1989, pp 44-61.

[41] Niparnan, N. y Sudsang, A., *Positive Span of Force and Torque Components of Four-Fingered Three-Dimensional Force-Closure Grasps*, IEEE International Conference on Robotics and Automation, 2007, pp 4701 - 4706.

[42] Ppapadimitriou, C. H., *Combinatorial Optimization*, Dover Publications, 1998.

[43] Penrose, R., *A generalized inverse for matrices*, Mathematical Proceedings of the Cambridge Philosophical Society, Vol 51(3), 1955, pp 406-413.

[44] Phoka, T., Niparnan, N. y Sudsang, A., *Planning Optimal Force-Closure Grasps for Curved Objects by Genetic Algorithm*, IEEE Conference on Robotics, Automation and Mechatronics, 2006, pp 1-6.

[45] Phoka, T., Niparnan, N. y Sudsang, A., *Planning Optimal Force-Closure Grasps for Curved Objects*, Proceedings of the IEEE International Conference on Robotics and Biomimetics, 2006, pp 721-726.

[46] Popov, V.L., *Contact Mechanis and Friction: Physycal Principles and Applications*, primera edición, Springer, 2010.

[47] Pratichizzo, D. y Trinkle, J. C., *Grasping Handbook of Robotics*, Springer, 2008, pp 671-700.

[48] Reuleaux, F., *The Kinematics of Machinery*, New York Macmillan, 1876.

[49] Rockafellar, R.T., *Convex Analysis*, Princeton University Press, 1970.

[50] Sahbani, A.; El-Khoury, S. y Bidaud, P. *A overview of 3D object grasp synthesis algorithms*, Robotics and Autonomus Systems, Vol 60(3), 2011.

[51] Salisbury, J. K. y Roth, B., *Kinematic and force analysis of articulated hands*, ASMEJournal of Mechanism, Transmissions, Automation, Design, Vol. 105, pp. 33-41, 1982.

[52] Schrijver, A., *Theory of Linear and Integer Programming*, Wiley, 1998.

[53] Shafarevich, I. R. y Remizov, A., *Linear Algebra and Geometry*, Springer Science and Business Media, 2012.

[54] Shamos, M.I *Computational Geometry*, PhD thesis, Yale University, 1978.

[55] Siciliano, B., Khatib, O., *Springer Handbook of Robotics*, Springer, 2oo8.

[56] Trinkle, J. C., *A quantitative Test for Form Closure Grasps*, Proceedings of the 1992 lEEE/RSJ International Conference on Intelligent Robots and Systems, Vol 3, 1992, pp 1670-1677.

[57] Trinkle, J. C., *On the stability and instantaneous velocity of grasped frictionless objects*, IEEE Transactions on Robotics and Automation, 1992, pp 560-572.

[58] van Oort, G. y Stramigioli, S. *Geometric interpretation of the Zero-Moment Point*, IEEE Int. Conference on Robotics and Automation, 2011, pp 575 - 580.

[59] Xiong, C.; Ding, H. y Xong, Y., *Fundamentals of Grasping and Fixturing*, World Scientific Publishing Company, 2007.

[60] Zheng, Y y Qian, W-H., *An Enhanced Ray-shooting approach to force-closure problems*, Journal of Manufacturing Science and Engineering, 2006.

[61] Zheng, Y y Chew, C-M., *Distance Between a Point and a Convex Cone in n-Dimensional Space: Computation and Applications*, IEEE Transactions on Robotics, Vol 25(6), 2009, pp 1397 - 1412.

[62] Zhu, X. y Ding, H., *Computation of Force-Closure Grasps: An Iterative Algorithm*, IEEE Transaccions on Robotics, Vol 22(1), 2006, pp 172-179 .

[63] Zhu, X. y Wang, J., *Synthesis of force-closure grasps on $3D$ objects based on the q distance*, IEEE Transactions on Robotics and Automation, Vol. 19(4), 2003, pp. 669–679.

[64] Zuo, Bing-Ran y Qian, Wen-Han, *On the Equivalence of Internal and Interaction Forces in Multifingered Grasps*, IEEE Trans. on Robotics and Automatons, Vol 15(5), 1999, pps. 934-941.