



**UNIVERSIDAD MICHOACANA DE SAN  
NICOLÁS DE HIDALGO**

**FACULTAD DE INGENIERÍA ELÉCTRICA**

**DIVISIÓN DE ESTUDIOS DE POSGRADO**

**MODELADO BIDIMENSIONAL ROBUSTO DE ENTORNOS  
INTERIORES PARA UN ROBOT MÓVIL EQUIPADO CON  
TELÉMETRO LÁSER**

**TESIS**

**QUE PARA OBTENER EL GRADO DE:**

**DOCTOR EN CIENCIAS EN INGENIERÍA ELÉCTRICA**

**PRESENTA**

**CARLOS ALBERTO LARA ÁLVAREZ**

**DIRECTOR DE TESIS**

**DR. LEONARDO ROMERO MUÑOZ**

**MORELIA MICHOACÁN, FEBRERO 2011.**





**MODELADO BIDIMENSIONAL ROBUSTO DE  
ENTORNOS INTERIORES PARA UN ROBOT MÓVIL  
EQUIPADO CON TELÉMETRO LÁSER**

**TESIS**

Que para obtener el grado de  
**DOCTOR EN CIENCIAS EN INGENIERÍA ELÉCTRICA**

presenta

**Carlos Alberto Lara Alvarez**

**Dr. Leonardo Romero Muñoz**

**Director de Tesis**

Universidad Michoacana de San Nicolás de Hidalgo

*A la memoria de mi padre*



# Resumen

El problema de generación eficiente de mapas con un robot autónomo móvil es un reto y ha atraído la atención de muchas investigaciones de diversos grupos especializados. Al problema de generar el mapa y al mismo tiempo localizarse en él se le conoce como SLAM (del inglés *Simultaneous Localization And Mapping*); a pesar del progreso logrado en el área, muchos de los algoritmos actuales funcionan muy bien en entornos simples pero su desempeño se reduce drásticamente en entornos complejos<sup>1</sup>.

El SLAM estocástico es el enfoque probabilista fundamental para generar mapas de forma autónoma y a partir de él se han propuesto numerosos enfoques; usualmente, el SLAM estocástico usa marcas para representar el entorno. En esta tesis se usa un telémetro láser para sensar ambientes interiores desde diferentes posiciones; las mediciones se usan para crear un mapa de líneas del entorno y para localizar al robot.

En su mayoría, los algoritmos de detección de marcas son heurísticas *ad hoc* y por ende los resultados obtenidos funcionan sólo en condiciones específicas. El enfoque Bayesiano es útil en este caso, ya que ofrece una formulación matemática simple pero poderosa. La primera contribución teórica de este trabajo es la aplicación de la formulación Bayesiana para detectar el mejor número de líneas a partir de mediciones láser tomadas desde una posición fija.

Esta tesis describe las ideas básicas del SLAM estocástico a través de la implementación del algoritmo para obtener un mapa de líneas. Las principales debilidades del algoritmo de SLAM estocástico son el error inducido por linealización y la complejidad temporal en que se incurre cuando crece el número de marcas en el mapa. En su mayoría, los enfoques que usan marcas también deben resolver el problema conocido como *asociación*, este problema es uno de los más difíciles de resolver. La segunda contribución teórica propone mejorar las técnicas actuales para asociación de líneas con el uso de la información cualitativa del entorno.

Los resultados obtenidos son superiores a otros esquemas reportados en la literatura, comprobando la construcción exitosa de mapas bajo distintas condiciones experimentales; estos mapas son muy útiles para tareas posteriores de navegación de los robots en los entornos explorados.

---

<sup>1</sup>Entornos dinámicos con muchos objetos pequeños



# Abstract

Efficient and reliable generation of maps is an essential element for any autonomous mobile robot. This challenging problem has attracted the attention of many specialized research groups. Despite significant progress in this area, many of the existing algorithms work well for simple environments but their performance is dramatically reduced in complex environments<sup>2</sup>.

The stochastic SLAM is the foundational probabilistic approach to determine maps; usually, this approach uses features to represent the map. This thesis uses a laser rangefinder to sense indoor environments from different poses; the measurements are then used to create a line map and to localize the robot.

Most of the feature extraction algorithms are *ad hoc* heuristics and therefore, their results are valid for particular conditions. The Bayesian approach is useful to solve the problem because it offers a simple but powerful mathematical formulation. The first theoretical contribution of this thesis is the application of the Bayesian formulation to extract the best number of lines from laser measurements taken from a fixed position.

This thesis describes the basic ideas of the stochastic SLAM through its implementation to determine line maps. The main weaknesses of the stochastic SLAM algorithm are the error induced by linearization and the time complexity for large scale problems. Most feature-based approaches must also solve the *association problem*, this problem is one of the most difficult to solve. The second theoretical contribution improves the commonly used association techniques by using qualitative information from the environment.

The results are superior to other schemes reported in the literature, proving the successful construction of maps under different experimental conditions, these maps are useful for future navigation tasks in the explored environments.

---

<sup>2</sup>dynamic environments with many small objects



# Contenido

Lista de Símbolos . . . . .	XII
1. Introducción . . . . .	1
1.1. El problema de SLAM . . . . .	2
1.1.1. Definición probabilista del SLAM . . . . .	3
1.1.2. Taxonomía del SLAM . . . . .	5
1.2. Enfoque de la tesis . . . . .	6
1.3. Hipótesis . . . . .	8
1.4. Retos del problema de SLAM . . . . .	9
1.5. Objetivo de la tesis . . . . .	9
1.6. Motivación . . . . .	10
1.7. Principales contribuciones de esta tesis . . . . .	10
1.8. Organización del documento . . . . .	11
2. Detección de marcas . . . . .	13
2.1. Definición del problema . . . . .	15
2.2. RANSAC: un enfoque frecuentista . . . . .	17
2.3. Enfoque Bayesiano . . . . .	18
2.4. Algoritmo propuesto . . . . .	21
2.5. Resultados experimentales . . . . .	25
2.6. Discusión . . . . .	27
3. SLAM estocástico . . . . .	29
3.1. Notación . . . . .	30
3.2. Descripción del algoritmo . . . . .	31
3.2.1. Inicialización . . . . .	33
3.2.2. Predicción . . . . .	33
3.2.3. Agregación . . . . .	35
3.2.4. Actualización . . . . .	37
3.3. Resultados experimentales . . . . .	38
3.4. Discusión . . . . .	42

---

4. Asociación entre marcas	47
4.1. Asociación basada en relaciones espaciales	48
4.1.1. Vecino más cercano compatible	48
4.1.2. Compatibilidad conjunta	50
4.1.3. Asociación basada en RANSAC	53
4.1.4. Dificultades de los esquemas de asociación	54
4.2. Asociación geométrica propuesta	55
4.3. Resultados experimentales	59
4.3.1. Prueba en entorno sintético	59
4.3.2. Prueba en entorno real	60
4.4. Discusión	62
5. Conclusiones	65
5.1. Contribuciones	65
5.1.1. Razón de verosimilitud para detección de marcas geométricas	65
5.1.2. Asociación de líneas usando secuencias de etiquetas	66
5.2. Trabajos futuros	67
<b>Apéndices</b>	<b>67</b>
A. Cálculo de integrales	69
A.1. Integral de funciones gaussianas	69
A.2. Cálculo de la evidencia	70
B. Estimación y Filtrado	71
B.1. Cálculo de covarianzas	71
B.2. El Filtro de Kalman	75
B.3. El Filtro de Kalman Extendido	77
C. Publicaciones del autor	79
Referencias	83



# Lista de Símbolos

$\mathbf{x} = [x, y, \theta]^T$	Posición del robot en el plano, donde: $x, y$ son las coordenadas de la posición y $\theta$ , la orientación del robot
$X$	Ruta del robot
$\mathbf{x}_i$	$i$ -ésima posición del robot; $\mathbf{x}_i \in X$
$U$	Secuencia de órdenes de control a lo largo de la ruta del robot
$\mathbf{u}_i$	Orden de control para cambiar de la posición $\mathbf{x}_{i-1}$ a $\mathbf{x}_i$ ; $\mathbf{u}_i \in U$
$Z$	Conjunto de todas las marcas obtenidas a lo largo de la trayectoria
$\mathbf{z}$	Parámetros de una marca (línea) con referencia local obtenidos de un proceso de medición, $\mathbf{z} \in Z$
$\mathbf{Y}$	Conjunto de marcas (líneas) obtenidas a partir de una secuencia de mediciones obtenidas desde una posición fija
$\mathbf{l}$	Parámetros de una línea en referencia global
$[r, \alpha]^T$	Parámetros de una línea en forma polar
$\mathbf{m}$	Mapa de líneas del entorno
$\mathbf{L}$	Subconjunto de líneas, $\mathbf{L} \subseteq \mathbf{m}$
$\mathbf{p}$	Medición tomada por el telémetro láser
$[\rho, \phi]^T$	Parámetros de un punto en forma polar
$S$	Secuencia de mediciones (puntos) tomadas con el telémetro láser desde una posición fija
$W$	Variable aleatoria
$\mathbf{C}_W$	Covarianza de la variable aleatoria $W$
$D_a$	Conjunto de puntos que siguen una distribución gaussiana respecto de la línea $\mathbf{l}_a$
$\tau$	Estado del sistema en el SLAM estocástico
$\hat{w}$	Valor estimado de la variable $y$

## Capítulo 1

# Introducción

En la actualidad, los robots se usan de manera extensa en la industria bajo condiciones controladas. Estos robots son indispensables en una gran parte de los procesos de manufactura. Sin embargo, muchas otras aplicaciones requieren *robots móviles* que se puedan desenvolver en ambientes dinámicos. Las tareas donde un robot móvil es útil son innumerables; algunas de estas tareas son simples y únicamente requieren una respuesta reactiva a los estímulos del entorno para completarlas; por ejemplo, en robots aspiradoras disponibles en el mercado. Otras tareas son más complejas pero más relevantes: toma de decisiones donde la integridad humana corre riesgos [Washington99], respaldo importante para personas con capacidades diferentes [Simpson05], guías de turistas [Thrun99], etc. En estas aplicaciones el robot móvil debe tener la capacidad de responder dos preguntas fundamentales: ¿dónde estoy? ¿a dónde voy?; estas dos preguntas son los dos problemas fundamentales en el área de robótica móvil conocidos como *Localización* y *Planificación de rutas*.

Para responder a la primera de las preguntas –¿dónde estoy?– el robot requiere una representación del entorno en el que va a realizar alguna tarea. A la representación del entorno se le conoce como *mapa* [Siegwart04]; una opción es considerar al mapa como requerimiento *a priori* para que el robot pueda localizarse y posteriormente planificar y ejecutar acciones de manera eficiente. Sin embargo, un robot autónomo debe generar su propia representación del entorno; es decir, debe generar un mapa a partir de las mediciones

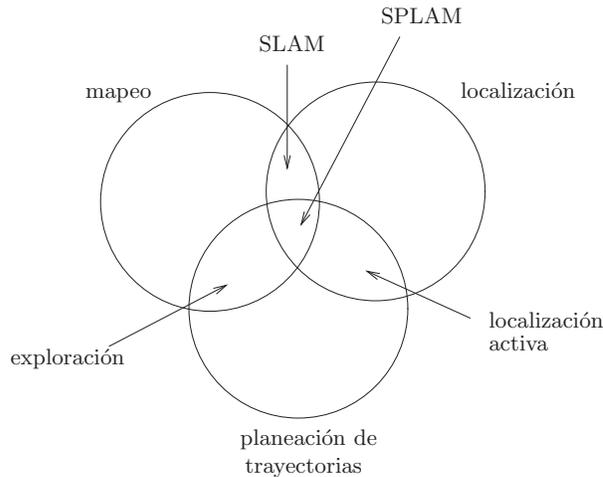


Figura 1.1: Actividades principales que debe realizar un robot móvil [Stachniss09].

obtenidas por sus sensores. Debido a las limitaciones sensoriales, el robot únicamente percibe una parte del entorno en que realizará sus tareas; por ende, tiene la necesidad de moverse a través del entorno para construir el mapa asociado. Al problema de generar el mapa y al mismo tiempo localizarse se conoce como SLAM (del inglés *Simultaneous Localization And Mapping*<sup>1</sup>) [Leonard91]. La Fig. 1.1 ilustra la relación de las diferentes actividades que debe realizar un robot móvil. El centro del área integra las actividades de Generación de mapas, Localización y planeación de rutas; a este enfoque integrado se le conoce como SPLAM (del inglés *Simultaneous Planning Localization and Mapping*) [Feder99].

## 1.1. El problema de SLAM

La forma más simple de localización se le conoce como *cálculo sin precisión*<sup>2</sup> y se obtiene integrando los estimados de los movimientos relativos (*v.g.* obtenidos con mediciones odométricas). El problema de este enfoque es que cada movimiento incluye errores que se acumulan cuando se integran; de esta manera, la incertidumbre de la posición se incrementa sin acotamiento.

<sup>1</sup>En lo sucesivo este problema será referido como “*el problema de SLAM*” o simplemente “*el SLAM*”. Su traducción al español es “*el problema de generación de mapas y localización concurrente*”.

<sup>2</sup>En inglés se le conoce como “*dead reckoning*”.

Si se resuelve el problema de SLAM entonces se evita el requerimiento de tener un mapa *a priori*. La idea general es que si hay marcas estáticas en el entorno, entonces la incertidumbre se puede mantener reducida. A pesar de ello, el SLAM no es un problema trivial debido a que el error en la posición del robot afecta la estimación del mapa y a su vez, el ruido del mapa afecta la estimación de la posición del robot. Encontrar una buena solución al problema del SLAM es crucial para lograr autonomía de robots en entornos desconocidos.

### 1.1.1. Definición probabilista del SLAM

Formalmente el problema de SLAM se puede definir más fácilmente de forma probabilista como se describe a continuación [Thrun05]. Sea  $X$  la ruta del robot

$$X = [\mathbf{x}_0, \mathbf{x}_1, \dots, \mathbf{x}_M], \quad (1.1)$$

donde  $\mathbf{x}_i$  es la  $i$ -ésima posición del robot; por ejemplo, para robots que se mueven en un plano<sup>3</sup> la posición del robot se puede describir por  $\mathbf{x}_i = [x_i, y_i, \theta_i]^\top$ .

Para desplazarse de una posición a la siguiente, el robot usa sus actuadores; una primera aproximación del cambio entre las posiciones se puede obtener del control del robot o del odómetro. La información de todas las acciones de control se puede recopilar como

$$U = [\mathbf{u}_1, \mathbf{u}_2, \dots, \mathbf{u}_M], \quad (1.2)$$

donde  $\mathbf{u}_i$  es un vector aproximado del cambio de posición<sup>4</sup> entre las posiciones adyacentes  $\mathbf{x}_{i-1}$  y  $\mathbf{x}_i$ . Usualmente, la  $i$ -ésima aproximación  $\mathbf{u}_i$  se obtiene con los sensores intrínsecos del robot (*v.g.* el odómetro). Si las lecturas odométricas no tuvieran errores, entonces  $U$  sería suficiente para conocer  $X$ ; desafortunadamente, la posición final calculada únicamente con  $U$  diverge sustancialmente de la posición final real debido a la acumulación de errores.

Finalmente, el robot obtiene mediciones sensoriales del entorno; la secuencia de mediciones obtenidas por el robot están dadas por

$$Z = [\mathbf{Z}_1, \mathbf{Z}_2, \dots, \mathbf{Z}_M], \quad (1.3)$$

<sup>3</sup>Usualmente se considera la hipótesis de piso plano.

<sup>4</sup>Note que el cambio de posición se expresa en coordenadas locales del robot.

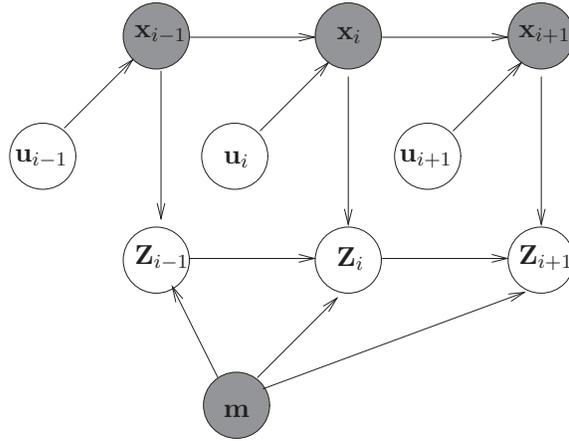


Figura 1.2: Modelo gráfico del SLAM [Thrun05]. Las flechas indican relaciones causales, los nodos sin relleno son variables observables por el robot. El problema consiste en encontrar las variables ocultas (los nodos sombreados).

donde  $\mathbf{Z}_i = \{z_{i_1}, z_{i_2}, \dots, z_{i_n}\}$  es un conjunto de mediciones o marcas obtenidas desde la  $i$ -ésima posición. La Fig. 1.2 muestra el diagrama de las diversas variables involucradas en el problema de SLAM. A tal diagrama se le conoce como *modelo gráfico* y muestra las diferentes dependencias entre las variables involucradas en el problema de SLAM.

El problema del SLAM consiste en recuperar el mapa del entorno y las posiciones del robot a partir de las mediciones sensoriales y odométricas; en lo sucesivo, se representa con  $\mathbf{m}$  el mapa del entorno<sup>5</sup>. La literatura distingue dos modos del problema de SLAM, ambos de gran importancia práctica. El primero, conocido como *SLAM total* [Thrun05], involucra estimar la probabilidad posterior del mapa y la ruta del robot;

$$P(\mathbf{X}, \mathbf{m} \mid Z, U). \quad (1.4)$$

El segundo, conocido como *SLAM en línea* [Thrun05], estima la probabilidad posterior del mapa y la posición final del robot  $\mathbf{x}_M$ ; es decir,

$$P(\mathbf{x}_M, \mathbf{m} \mid Z, U). \quad (1.5)$$

<sup>5</sup>La representación del mapa es un factor muy importante y modifica sustancialmente el tipo de algoritmo usado.

Esta tesis se orienta a mejorar el desempeño de los algoritmos actuales para obtener mapas en entornos interiores de mediana escala donde hay una gran cantidad de objetos (tales como sillas, mesas, etc). Aunque se estudia principalmente el SLAM en línea, las técnicas propuestas son útiles también para el caso del SLAM total.

### 1.1.2. Taxonomía del SLAM

Como se describe en la sección anterior, el problema de SLAM se puede dividir como SLAM total y en línea; otras clasificaciones que se usan en la literatura son:

**Volumétrico vs. basado en características.** Los mapas volumétricos usan rejillas discretas para representar el mapa [Elfes89, Romero01]. Las rejillas pueden llegar a ser de alta resolución de tal manera que pueden permitir reconstrucción fotorealista del entorno. Una variante de este tipo de mapas es el uso de datos sin procesar [Nieto05, Romero05, Lara08]. El mapa volumétrico  $\mathbf{m}$  es usualmente de alta dimensionalidad, con las complicaciones que esto implica; una alternativa que evita la complejidad espacial es la representación en el espacio de características. Los enfoques basados en características (o marcas) son más eficientes pero los resultados son usualmente de inferior calidad debido a la pérdida de información provocada por la abstracción.

**Topológico vs. métrico.** Los mapas topológicos describen de forma cualitativa el entorno [Kuipers91]; un mapa topológico se define como un conjunto de lugares y un conjunto de relaciones entre dichos lugares. En otras palabras, en un mapa topológico la posición del robot se define por el estado (lugar) en el que se encuentra mientras que los mapas métricos definen la posición del robot de forma cuantitativa (v.g.  $[x, y, \theta]^T$  para un robot en 2D). Un inconveniente de los mapas topológicos es que se requiere un mecanismo de distinción sensorial entre diferentes lugares, esto dificulta la representación de zonas abiertas o habitaciones grandes; por otra parte, los mapas métricos son vulnerables a imprecisiones; una extensión conocida como mapas híbridos [Tomatis08] trata de obtener ventajas de combinar ambos enfoques.

**Correspondencias conocidas vs. desconocidas.** En general, existe la necesidad de relacionar mediciones sensoriales con otras mediciones obtenidas previamente. El proble-

ma de establecer correspondencias se le conoce como DAP (del inglés *Data Association Problem*) y es uno de los problemas más difíciles de resolver en robótica. Muchos autores no consideran el problema de asociación sobre todo cuando abordan algoritmos probabilistas desde un punto de vista abstracto [Thrun05].

**Estático vs. dinámico.** Los algoritmos de SLAM estático consideran que el entorno no cambia a través del tiempo; en contraste, los algoritmos de SLAM dinámico están interesados en obtener un mapa a pesar que haya cambios en el entorno. En robótica es una práctica común permitir que el robot obtenga un mapa en condiciones ideales del entorno (sin personas, con iluminación adecuada, etc.) para que posteriormente se pueda desempeñar en condiciones operacionales; un ejemplo muy conocido de esta situación es el robot Minerva [Thrun00].

**Grado de incertidumbre.** Los algoritmos de SLAM también se pueden dividir en aquellos que son capaces de manejar pequeños errores, mientras que otros son capaces de resolver el problema del SLAM a pesar de que exista una incertidumbre muy elevada. Cuando el grado de incertidumbre es muy elevado se dificulta el cerrado de ciclos<sup>6</sup> ya que se acumulan muchos errores a lo largo de la ruta.

**Robot único vs. multirobot.** Muchos de los algoritmos de SLAM se han diseñado pensando en un único robot. Sin embargo, se han popularizado soluciones que involucran más de un robot. En estos enfoques es importante la comunicación entre los robots.

## 1.2. Enfoque de la tesis

Usando la taxonomía que se introduce en la sección anterior, el enfoque SLAM usado en esta tesis se clasifica como:

**SLAM en línea/SLAM total.** Aunque esta tesis estudia esencialmente el problema de SLAM en línea –a través de la implementación de la técnica conocida como SLAM

---

<sup>6</sup>Un ciclo se define como una posición a la que se puede llegar desde varias rutas, los bucles son difíciles de detectar sobre todo para rutas largas.

estocástico; también se ha implementado una técnica de suavizado que resuelve el SLAM total.

**Basado en características.** Las soluciones que se presentan en esta tesis usan características. Es decir, el mapa  $\mathbf{m}$  es un conjunto de marcas

$$\mathbf{m} = \{\mathbf{l}_j \mid j = 1, \dots, N\}, \quad (1.6)$$

donde,  $\mathbf{l}_j$  son los parámetros de una marca. Específicamente, se han escogido líneas como marcas. De tal manera que  $\mathbf{l}_j$  representa los parámetros de una línea. Cabe destacar que las líneas son inherentes a entornos interiores por lo se que evita la necesidad de infraestructura adicional.

**Métrico.** Se desea obtener un mapa métrico que sea fácil de interpretar por cualquier persona.

**Correspondencias desconocidas.** Se desea que el algoritmo sea capaz de identificar correspondencias entre las diferentes marcas.

**Estático.** Para obtener el mapa, se considera que el único objeto que se mueve en el entorno es el robot.

**Robot único.** El mapa será generado por un solo robot.

### Proceso general

El proceso para generar mapas de líneas del entorno a partir de  $M$  secuencias de mediciones se muestra en la Fig. 1.3; a continuación se describen los pasos relevantes:

1. Obtener una secuencia  $S_i$  de mediciones láser del entorno (se supone que el robot no se mueve mientras se toman las mediciones).
2. Extraer un conjunto de líneas  $\mathbf{Z}_i$  que mejor represente a  $S_i$ . Debido a que en los entornos interiores usualmente existen muchos objetos planos, se usan líneas como marcas.

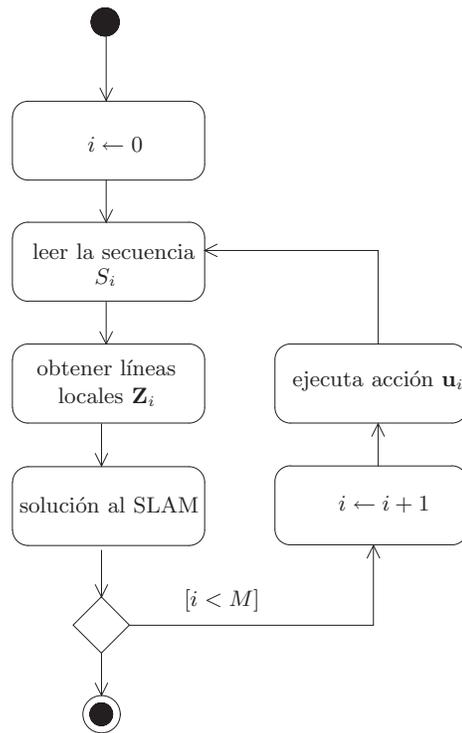


Figura 1.3: Proceso general para la obtención de un mapa desde  $M$  posiciones

3. Integrar las líneas locales al mapa y estimar la posición del robot; es decir, resolver el problema de SLAM.
4. Si el recorrido no ha terminado, el robot se desplaza usando la orden  $\mathbf{u}_i$  y se regresa al paso 1.

### 1.3. Hipótesis

Existe un método bayesiano que permite la detección del conjunto de líneas que mejor representan a una secuencia de mediciones obtenidas con un telémetro láser bidimensional. Este método es capaz de descubrir cuál es la cantidad de líneas presentes en el conjunto de mediciones.

Los conjuntos de líneas obtenidos por el método bayesiano propuesto son más precisos que aquellos obtenidos por otras técnicas reportadas en la literatura. De igual

manera, los algoritmos de SLAM obtienen mejores mapas de líneas de entornos interiores complejos cuando se usa el método bayesiano propuesto.

## 1.4. Retos del problema de SLAM

Los retos más importantes en el problema del SLAM son dos[Martinelli04]: la convergencia del mapa, y los requerimientos computacionales (tanto en procesamiento como en memoria). Estos dos retos, son críticos en aplicaciones en tiempo real o en entornos reales.

Para lograr convergencia, los enfoques basados en características requieren obtener marcas confiables y resolver el *problema de asociación*. En ocasiones, la extracción de marcas confiables no es una tarea trivial ya que se deben inferir a partir de información ruidosa. Por ejemplo, para un telémetro láser es difícil obtener mediciones de entornos con vidrios y superficies oscuras. Además del ruido, la existencia de muchos objetos pequeños (patas de sillas, mesas, computadoras, etc) dificulta la extracción de marcas. Por otra parte, el problema de asociación consiste en determinar cual es la marca del mapa que corresponde a una marca que se ha medido recientemente; este problema es crítico, pues una mala asociación puede ser suficiente para obtener un mapa que sea sustancialmente diferente al real. Los problemas de extracción y asociación de marcas son diferentes dependiendo del tipo de sensor usado y de las características del entorno.

Desde un punto de vista más general, cada enfoque tiene diferentes dificultades de convergencia y requerimientos computacionales. Por ejemplo, el SLAM estocástico tiene problemas de convergencia ya que aproxima funciones no lineales y problemas de requerimientos computacionales que aumentan drásticamente con el incremento del número de marcas.

## 1.5. Objetivo de la tesis

El objetivo principal de esta tesis consiste en generar mapas bidimensionales de entornos interiores con un robot móvil; el mapa consistirá de líneas y segmentos de líneas. Los únicos sensores que se usarán son el telémetro láser y la información odométrica del

robot. El robot se desenvuelve en un ambiente tipo oficina real donde abundan sillas, mesas, vidrios, puertas oscuras, etc; el entorno no debe sufrir ninguna modificación y se puede considerar la hipótesis de piso plano. Finalmente, el método propuesto debe ser útil en aplicaciones de tiempo real además de ser robusto.

## 1.6. Motivación

La importancia de resolver el SLAM nunca ha sido exagerada: este problema ha sido considerado como fundamental si se desea obtener un robot móvil que sea verdaderamente autónomo.

En la literatura se han propuesto muchas técnicas para resolver el SLAM. Entre otros factores la calidad de los resultados depende directamente del tipo de sensor usado, en la actualidad se ha difundido el uso del láser y las cámaras. Muchos algoritmos se basan en el uso de marcas presentes en el entorno; por lo que la detección de marcas a partir de las lecturas sensoriales es un proceso de vital importancia. Después de su detección, los algoritmos de SLAM deben manejar las marcas; es decir, se deben identificar marcas previamente observadas y eliminar aquellas que no existen en el entorno. La correcta detección y manejo de las marcas da como resultado mapas confiables que son útiles para otras actividades del robot; es por ello que esta tesis se enfoca en estos aspectos.

El autor considera que muchos de estos problemas se pueden resolver de forma elegante usando la teoría Bayesiana. Finalmente, el ámbito de aplicación de las técnicas que surgan de este trabajo se pueden aplicar en otras áreas de la ciencia y la tecnología, sobre todo cuando se requiere obtener más de un modelo a partir de un conjunto de datos.

## 1.7. Principales contribuciones de esta tesis

La primera contribución teórica de este trabajo es la aplicación de la formulación Bayesiana para extraer el mejor número de líneas a partir de una secuencia de mediciones láser tomadas desde una posición fija. La segunda contribución consiste en mejorar las técnicas para asociación de líneas con el uso de una representación cualitativa del en-

torno. Los resultados obtenidos son superiores a otros esquemas reportados en la literatura hasta la fecha, comprobando la construcción exitosa de mapas bajo distintas condiciones experimentales.

## 1.8. Organización del documento

El resto del presente documento se organiza como se describe a continuación:

- El Capítulo 2 estudia el problema de extracción de líneas a partir de una secuencia de mediciones con el telémetro láser; este Capítulo presenta la contribución más importante de esta tesis.
- El Capítulo 3 explica los detalles de la implementación del algoritmo clásico conocido como SLAM estocástico para obtener mapas de líneas. El caso de estudio propuesto consiste en generar un mapa local en un entorno interior real.
- El Capítulo 4 estudia los algoritmos de asociación basados en relaciones espaciales y propone la técnica de asociación de líneas usando secuencias de medición.
- El Capítulo 5 presenta las conclusiones derivadas de esta investigación así como las líneas de trabajo futuro.
- El Apéndice A define como calcular integrales de funciones gaussianas en varias dimensiones, y se deducen las ecuaciones para calcular las integrales involucradas en este trabajo.
- El Apéndice B describe como calcular covarianzas para variables aleatorias gaussianas y el filtro de Kalman.
- El Apéndice C describe la producción científica del autor derivada de este trabajo.



## Capítulo 2

# Detección de marcas

En robótica móvil, las marcas son importantes en muchas tareas de alto nivel tales como localización, cartografía, reconocimiento de lugares, o navegación. Las *marcas o características* son estructuras reconocibles en el entorno [Siegwart04]. Dada una secuencia de mediciones del entorno, el problema de detección de marcas consiste en encontrar la cantidad de marcas y los parámetros de cada marca. El tipo de marcas que se usa para representar el entorno depende del sensor usado. Thrun [Thrun98] distingue entre marcas artificiales, que se han dispuesto estratégicamente en el entorno; y marcas naturales, propias de cada entorno.

Las marcas geométricas como líneas, círculos, planos o cónicas son convenientes para representar mediciones obtenidas con el telémetro láser. Entre las técnicas que se usan para detección de marcas se encuentran: técnicas de detección ascendente [Duda73, Borges00, Siegwart04], técnicas probabilistas [Thrun03, Han04], RANSAC (del inglés *Random Sample Consensus*) [Bolles81, Schnabel07] y transformada de Hough [Leavers93], entre otras. Nguyen *et al.* [Nguyen07] comparan algoritmos de detección de marcas geométricas a partir de mediciones de láser.

Detectar la cantidad y el tipo de modelos que mejor representan a un conjunto de puntos es un reto. El *enfoque secuencial* es una solución simple que detecta una marca a la vez; es decir, la mejor marca se extrae en cada iteración y los datos que pertenecen a dicha marca se eliminan del conjunto de datos. Para que los resultados obtenidos sean aceptables,

el enfoque secuencial usa técnicas de *regresión robusta*. Las técnicas de regresión robusta pueden obtener un modelo que sea representativo a pesar que una proporción elevada de mediciones no pertenezcan al modelo buscado. El algoritmo RANSAC [Bolles81] es una técnica de regresión robusta muy difundida gracias a que es tolerante a ruido. El enfoque secuencial es voraz y, por ende, no considera la relación entre diferentes marcas del entorno; como consecuencia, el resultado suele ser impreciso para casos no triviales (cuando existen líneas paralelas muy próximas entre sí y muchos objetos pequeños).

La solución conocida como *enfoque de detección ascendente* o *agrupamiento aglomerativo* se usa comúnmente en tareas de reconocimiento de patrones en el área de visión. Inicialmente se detectan marcas locales representadas por un grupo de puntos y después los grupos se fusionan entre sí con el objetivo de obtener las marcas reales. El criterio más simple de fusión es el siguiente: si la distancia entre las marcas más próximas es menor a un determinado umbral las marcas se fusionan; de lo contrario el proceso termina [Siegwart04]. En este caso, el resultado depende fuertemente de la distancia (típicamente se usa la distancia Euclidiana o de Mahalanobis) y del umbral seleccionado.

En la literatura se han propuesto algunos algoritmos probabilistas para resolver el problema de detección de marcas. Thrun *et al.* [Thrun03] proponen una variante del algoritmo EM (del inglés *Expectation Maximization*) que penaliza el número de modelos usando una función de distribución exponencial; esta distribución se introduce al modelo como conocimiento *a priori*. Para encontrar el mejor número de marcas se implementa una fase de búsqueda que se entrelaza con el algoritmo EM; la fase de búsqueda incrementa o decrementa el número de marcas en intervalos regulares.

Han *et al.* [Han04] formulan el problema en un contexto Bayesiano, aunque también usan una distribución *a priori* para penalizar el número de marcas. El algoritmo simula cadenas de Markov con saltos reversibles y difusión estocástica. Debido a que este algoritmo trata de seleccionar el tipo de marca que mejor se ajusta a los datos, los saltos reversibles se usan para cambiar la cantidad y tipo de marcas.

Este capítulo estudia el problema de detección de marcas en entornos interiores; específicamente, se aborda el problema de detección de líneas a partir de mediciones tomadas con el láser. Se seleccionaron los segmentos lineales como marca geométrica porque en

entornos interiores abundan los objetos planos.

El resto del capítulo se organiza como sigue: la sección 2.1 define el problema de detección de segmentos lineales a partir de mediciones del telémetro láser, la sección 2.2 introduce el enfoque conocido como RANSAC, la sección 2.3 presenta la principal contribución de esta tesis: un modelo Bayesiano para la fusión de segmentos lineales, la sección 2.4 integra el modelo propuesto en un algoritmo que es útil para la detección de múltiples líneas, la sección 2.5 muestra experimentos que se realizaron con datos simulados; finalmente, la sección 2.6 presenta las conclusiones de este capítulo.

## 2.1. Definición del problema

Sea  $S = \{\mathbf{p}_1, \dots, \mathbf{p}_N\}$  una secuencia de mediciones obtenidas por un telémetro láser bidimensional. Cada medición  $\mathbf{p}_i \in S$  se representa por un punto en coordenadas polares;

$$\mathbf{p}_i = [\rho_i, \phi_i]^\top, \quad (2.1)$$

donde  $\rho_i$  es la distancia del objeto más próximo al sensor en el ángulo de disparo  $\phi_i$ . El punto  $\mathbf{p}_i$  también puede ser expresado en sus coordenadas rectangulares como

$$\mathbf{p}_i = [x_i, y_i]^\top = [\rho_i \cos \phi_i, \rho_i \sin \phi_i]^\top \quad (2.2)$$

Se desea encontrar el conjunto de líneas que mejor representan a  $S$ ; es decir

$$\mathbf{Z} = \{\mathbf{z}_1, \dots, \mathbf{z}_\eta\}. \quad (2.3)$$

Para obtener  $\mathbf{Z}$  es necesario descubrir el número de líneas  $\eta$ , y los parámetros de cada línea. La  $j$ -ésima línea se representa en su forma polar como

$$\mathbf{z}_j = [r_j, \alpha_j]^\top, \quad (2.4)$$

donde  $r_j$  y  $\alpha_j$  son la longitud y el ángulo de inclinación de la normal, respectivamente. Como se ilustra en la Fig. 2.1, la normal se define como el segmento ortogonal más corto entre la línea y el origen.

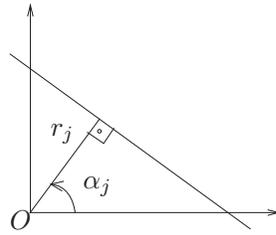


Figura 2.1: Parámetros de una línea en coordenadas polares.

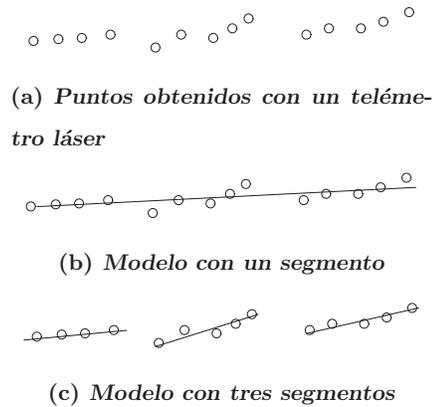


Figura 2.2: Detección de segmentos lineales.

El reto para cualquier algoritmo de detección de marcas es encontrar una representación realista del entorno; un ejemplo donde las líneas extraídas puede ser diferente a las existentes en el entorno real se muestra en la Fig. 2.2. Dado un conjunto de puntos como el mostrado en la Fig. 2.2a, se desea obtener el mejor conjunto de segmentos lineales. El resultado dependerá del número de segmentos lineales, en la Fig. 2.2b se muestra el resultado obtenido con un segmento lineal mientras que en la Fig. 2.2c se muestra el resultado usando tres segmentos lineales.

Las hipótesis del número de segmentos pueden ser tan diversas que en el caso extremo cada par de puntos puede representar una línea; por ello surge la necesidad de obtener una forma de evaluar qué cantidad de líneas es más conveniente. A continuación se muestran alternativas para encontrar el número de segmentos.

## 2.2. RANSAC: un enfoque frecuentista

Los métodos convencionales de ajuste de curvas tal como el ajuste de mínimos cuadrados consideran que las mediciones son variables aleatorias que siguen una distribución gaussiana. Esta suposición no siempre es correcta: uno o varios datos experimentales pueden no cumplir esta hipótesis; este tipo de datos son conocidos como *datos atípicos* o *valores extremos*. Los datos atípicos pueden afectar drásticamente el resultado [Rousseeuw87].

El algoritmo conocido como RANSAC (del inglés *Random Sample Consensus*) es un estimador robusto propuesto por Fischler y Bolles [Bolles81]. El algoritmo RANSAC es muy popular para ajustar un modelo único a un conjunto de datos experimentales. Las razones de su popularidad son su tolerancia a valores atípicos y que es fácil de implementar. La filosofía de RANSAC es opuesta a muchas técnicas robustas convencionales: en lugar de tomar tantos datos como sea posible para obtener soluciones iniciales y tratar de eliminar datos atípicos, RANSAC usa la mínima cantidad de datos para obtener un modelo inicial e incrementa dicho conjunto con datos consistentes cuando sea posible.

Al menor número de datos requerido para definir un modelo se le conoce como conjunto mínimo: dos puntos para definir una línea, tres para un plano, etc. Cuando se usa RANSAC, se seleccionan conjuntos mínimos; cada conjunto genera una hipótesis del mejor modelo; los modelos son luego comparados entre sí con algún criterio estadístico que considera la totalidad de los datos medidos.

Se han realizado muchos esfuerzos para mejorar el desempeño de RANSAC: reemplazo de la función estadística de costo, incorporar un estimador robusto tal como los estimadores-M, mejorar la selección del conjunto mínimo. Para una revisión general de las variantes puede consultar [Lara06]. RANSAC ha demostrado ser muy efectivo para encontrar un único modelo y por ello sus aplicaciones son muy diversas.

Se han adaptado variantes de RANSAC para encontrar múltiples modelos [Lara06]. La técnica más simple consiste en aplicar el algoritmo de forma secuencial: en cada iteración se encuentra un modelo, los puntos asociados a dicho modelo se remueven del conjunto de datos. Para obtener resultados relativamente buenos, el algoritmo RANSAC secuencial requiere iterar una gran cantidad de veces; y a pesar de ello, los resultados no son los sufi-

cientemente confiables [Lara-Alvarez09]. En lugar de encontrar un modelo a la vez, el autor propone encontrar un conjunto de modelos al mismo tiempo [Lara-Alvarez09]. Una de las principales desventajas de este enfoque es que usa umbrales para su correcto funcionamiento: el umbral para considerar un dato como atípico, la proporción de puntos atípicos en la muestra, la confiabilidad [Hartley00]. La cantidad de parámetros puede aumentar considerablemente cuando se busca más de un modelo: cantidad de puntos mínimos para considerar que un modelo es válido, límites de región de búsqueda inicial, etc. Por ende, los resultados dependerán mucho de la experiencia del diseñador para encontrar los parámetros con los que funciona mejor el algoritmo en la situación particular.

### 2.3. Enfoque Bayesiano

El algoritmo RANSAC es fácil de implementar pero requiere ajustar algunos parámetros para su correcto funcionamiento. En esta sección se propone una alternativa que se basa en el enfoque Bayesiano para análisis de datos [Sivia96]. La propuesta tiene relación con los algoritmos de detección ascendente [Siegwart04, Duda73, Borges00] en el sentido de que primero se obtienen marcas locales con su correspondiente grupo de mediciones y después se fusionan los grupos similares para obtener el resultado. Aquí se usa una prueba probabilista en lugar de una distancia para tomar decisiones de fusión. El enfoque propuesto es similar al propuesto por Heller y Ghahramani [Heller05] para agrupamientos generales. A continuación se describe el modelo probabilista que se usará para decidir si dos agrupamientos se deben de fusionar o no. De aquí en adelante, el conocimiento previo del problema se denotará por  $I$ ; información típica que se incluye en  $I$  es la hipótesis de ruido gaussiano, el tipo de entorno, etc.

### Prueba de razón de verosimilitud (LRT) para líneas

Sean  $D_a$  y  $D_b$ , dos agrupamientos de puntos obtenidos a partir de objetos lineales con  $N_a$  y  $N_b$  mediciones, respectivamente; deseamos conocer cual de las siguientes hipótesis es más probable:

$H_1$ : los dos conjuntos son mediciones del mismo objeto plano; por tanto, el conjunto

$$D = D_a \cup D_b \text{ se caracteriza por una línea con parámetros desconocidos } \mathbf{z}_* = [r_*, \alpha_*]^\top.$$

$H_2$ : existe un cambio significativo entre los conjuntos; por tanto, las mediciones se representan mejor por dos líneas diferentes con parámetros  $\mathbf{z}_{a*} = [r_{a*}, \alpha_{a*}]^\top$  y  $\mathbf{z}_{b*} = [r_{b*}, \alpha_{b*}]^\top$ , respectivamente.

La razón de verosimilitud entre las hipótesis  $H_1$  y  $H_2$  es útil para tomar una decisión. Si se considera la misma probabilidad *a priori* para las dos hipótesis; esto es, si  $P(H_1 | I) = P(H_2 | I)$  entonces aplicando el teorema de Bayes tenemos;

$$\begin{aligned} R(D_a, D_b) &= \frac{P(H_1 | D_a, D_b, I)}{P(H_2 | D_a, D_b, I)} = \frac{\frac{P(D_a, D_b | H_1, I)P(H_1, I)}{P(D_a, D_b | I)}}{\frac{P(D_a, D_b | H_2, I)P(H_2, I)}{P(D_a, D_b | I)}} \\ &= \frac{P(D_a, D_b | H_1, I)}{P(D_a, D_b | H_2, I)}. \end{aligned} \quad (2.5)$$

Si el valor de  $R(D_a, D_b)$  es mayor que uno, entonces se prefiere la hipótesis  $H_1$ ; si es menor que uno, entonces se prefiere  $H_2$ ; y si es de orden unitario, entonces las mediciones obtenidas son insuficientes para tomar un juicio informado.

Usando la regla de probabilidad total, el numerador de (2.5) se puede expresar como

$$P(D_a, D_b | H_1, I) = \int_0^{\alpha_m} \int_0^{r_m} P(D_a, D_b | r, \alpha, H_1, I) P(r, \alpha | H_1, I) dr d\alpha \quad (2.6)$$

donde  $r_m$  y  $\alpha_m$  son los valores máximos de  $r$  y  $\alpha$ , respectivamente. Para asignar la probabilidad *a priori*, se usa una distribución uniforme sobre los intervalos  $0 < r < r_m$ , y  $0 < \alpha < \alpha_m$ ,

$$P(r, \alpha | H_1, I) \propto \frac{1}{r_m \alpha_m}. \quad (2.7)$$

Considerando ruido gaussiano e independencia entre las mediciones, para cualquier línea  $\mathbf{z} = [r, \alpha]^\top$

$$P(D_a, D_b | r, \alpha, H_1, I) \propto \left(\sigma\sqrt{2\pi}\right)^{-N} \exp\left(-\frac{\chi^2}{2}\right), \quad (2.8)$$

donde  $N = N_a + N_b$ ; y

$$\chi^2 = \sum_{\mathbf{p} \in (D_a \cup D_b)} \frac{d_{\perp}^2(\mathbf{p}, \mathbf{z})}{\sigma^2}, \quad (2.9)$$

es la suma de los errores residuales cuadráticos; aquí,  $\sigma^2$  es la varianza de las mediciones y  $d_{\perp}(\mathbf{p}, \mathbf{z})$  es la distancia ortogonal del punto  $\mathbf{p} = [x, y]^T$  a la línea  $\mathbf{z} = [r, \alpha]^T$ ,

$$d_{\perp}(\mathbf{p}, \mathbf{z}) = r - x \cos \alpha - y \sin \alpha. \quad (2.10)$$

Para resolver las integrales de (2.6) se puede usar la expansión de Taylor alrededor de los parámetros de la línea estimada por el método de mínimos cuadrados. Sea  $\mathbf{z}_* = [r_*, \alpha_*]^T$ , la línea de mínimos cuadrados obtenida a partir de los puntos  $(D_a \cup D_b)$ ; y  $\chi_*^2$ , el error correspondiente. La expansión de Taylor alrededor de  $\chi_*^2$  es

$$\chi^2 \approx \chi_*^2 + \frac{1}{2}(\mathbf{z} - \mathbf{z}_*)^T \nabla \nabla \chi^2(\mathbf{z} - \mathbf{z}_*) + \dots, \quad (2.11)$$

donde  $\nabla \nabla \chi^2$  es el Hessiano evaluado en  $\mathbf{z}_*$  y el segundo término se ha omitido porque  $\chi_*^2$  es un mínimo. Para un determinado agrupamiento  $D$ , el valor de  $\nabla \nabla \chi^2$  se calcula como:

$$\nabla \nabla \chi^2 = \left[ \begin{array}{cc} \frac{\partial^2 \chi^2}{\partial^2 r} & \frac{\partial^2 \chi^2}{\partial r \partial \alpha} \\ \frac{\partial^2 \chi^2}{\partial r \partial \alpha} & \frac{\partial^2 \chi^2}{\partial^2 \alpha} \end{array} \right] \Bigg|_{\mathbf{z}=\mathbf{z}_*},$$

donde:

$$\begin{aligned} \frac{\partial^2 \chi^2}{\partial^2 r} &= \frac{2}{\sigma^2} N, \\ \frac{\partial^2 \chi^2}{\partial r \partial \alpha} &= \frac{2}{\sigma^2} \sum_{\mathbf{p} \in D} (x \sin \alpha - y \cos \alpha), \text{ y} \\ \frac{\partial^2 \chi^2}{\partial^2 \alpha} &= \frac{2}{\sigma^2} \sum_{\mathbf{p} \in D} (r - x \cos \alpha - y \sin \alpha)(x \sin \alpha + y \cos \alpha) + (x \sin \alpha - y \cos \alpha)^2. \end{aligned}$$

Sustituyendo (2.7), (2.8) y (2.11) en (2.6) y resolviendo la integral como se describe en el apéndice A, tenemos:

$$\begin{aligned} &P(D_a, D_b \mid H_1, I) \\ &\propto \frac{1}{r_m \alpha_m (\sigma \sqrt{2\pi})^N} \int_0^{\alpha_m} \int_0^{r_m} \exp\left(-\frac{\chi^2}{2}\right) dr d\alpha \\ &= \frac{1}{r_m \alpha_m (\sigma \sqrt{2\pi})^N} \frac{4\pi}{\sqrt{\det \nabla \nabla \chi^2}} \exp\left(\frac{-\chi_*^2}{2}\right). \end{aligned}$$

Si ahora se hace un análisis similar para los agrupamientos  $D_a$  y  $D_b$  bajo la hipótesis  $H_2$ ; es decir, considerando los agrupamientos por separado se obtiene

$$P(D_a, D_b | H_2, I) = \frac{1}{r_m \alpha_m (\sigma \sqrt{2\pi})^{N_a}} \frac{4\pi}{\sqrt{\det \nabla \nabla \chi_a^2}} \exp\left(\frac{-\chi_{a*}^2}{2}\right) \times \frac{1}{r_m \alpha_m (\sigma \sqrt{2\pi})^{N_b}} \frac{4\pi}{\sqrt{\det \nabla \nabla \chi_b^2}} \exp\left(\frac{-\chi_{b*}^2}{2}\right) \quad (2.12)$$

donde  $\chi_{a*}^2$  y  $\chi_{b*}^2$  son los errores cuadráticos alrededor de las líneas de mínimos cuadrados para cada uno de los agrupamientos.

Finalmente, la razón de verosimilitud (2.5) está dada por

$$\begin{aligned} R(D_a, D_b) &= \frac{P(D_a, D_b | H_1, I)}{P(D_a, D_b | H_2, I)} \\ &= \frac{\frac{1}{r_m \alpha_m (\sigma \sqrt{2\pi})^{N_a + N_b}} \frac{4\pi}{\sqrt{\det \nabla \nabla \chi^2}} \exp\left(\frac{-\chi_*^2}{2}\right)}{\frac{1}{r_m \alpha_m (\sigma \sqrt{2\pi})^{N_a}} \frac{4\pi}{\sqrt{\det \nabla \nabla \chi_a^2}} \exp\left(\frac{-\chi_{a*}^2}{2}\right) \frac{1}{r_m \alpha_m (\sigma \sqrt{2\pi})^{N_b}} \frac{4\pi}{\sqrt{\det \nabla \nabla \chi_b^2}} \exp\left(\frac{-\chi_{b*}^2}{2}\right)} \\ &= \frac{r_m \alpha_m}{4\pi} \sqrt{\frac{\det(\nabla \nabla \chi_a^2) \det(\nabla \nabla \chi_b^2)}{\det(\nabla \nabla \chi^2)}} \exp\left(\frac{J}{2}\right); \end{aligned} \quad (2.13)$$

donde  $J = \chi_{a*}^2 + \chi_{b*}^2 - \chi_*^2$ .

La ecuación (2.13) se puede usar para decidir cuales agrupamientos se deben de fusionar. En lo sucesivo, nos referimos a la razón expresada por (2.13) como LRT (del inglés *Likelihood Ratio Test*). Los factores que multiplican al término exponencial penalizan modelos con más marcas; de tal manera que (2.13) sigue el principio conocido como el rastrillo de Ockham. En palabras simples, este principio establece que “la explicación más simple para un fenómeno es preferible sobre explicaciones más complejas”.

## 2.4. Algoritmo propuesto

Inicialmente, el algoritmo obtiene un conjunto de agrupamientos de tamaño  $\eta$ ,  $D = \{D_1, \dots, D_\eta\}$ ; cada agrupamiento  $D_i \in D$  sigue un modelo lineal. En cada iteración el par de agrupamientos  $D_a, D_b \in D \mid a \neq b$  con valor más alto de  $R(D_a, D_b)$  se mezclan para reducir el número de agrupamientos a  $\eta - 1$ . El procedimiento continúa hasta que no haya

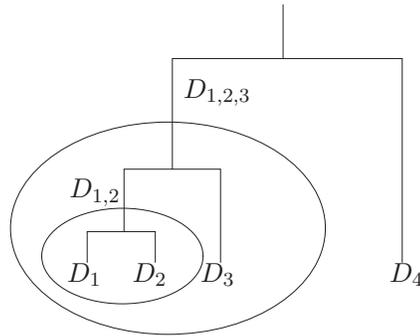


Figura 2.3: Ilustración de un árbol donde  $D_1$  y  $D_2$  se mezclan en  $D_{1,2}$  y después  $D_{1,2}$  se mezcla con  $D_3$  para obtener el agrupamiento  $D_{1,2,3}$ .

agrupamientos que se puedan mezclar ya sea porque se detectó un único agrupamiento o porque la razón de verosimilitud más alta es menor que uno.

Los agrupamientos iniciales se pueden obtener con cualquier técnica de búsqueda local; por ejemplo, la ventana deslizante [Siegwart04], el método IEPF (del inglés *Iterative End Point Fit*) [Duda73], o el método LT (del inglés *Line Tracking*) [Duda73].

Para ilustrar este proceso, la Fig. 2.3 muestra cuatro agrupamientos detectados por la fase inicial  $D = \{D_1, D_2, D_3, D_4\}$ . En la ilustración se considera que valor más alto de LRT es  $R(D_1, D_2) > 1$  por lo que se obtiene el nuevo agrupamiento  $D_{1,2} = D_1 \cup D_2$ . De igual manera la ilustración considera que en la siguiente iteración el valor más alto de LRT es  $R(D_{1,2}, D_3) > 1$  y por tanto  $D_{1,2,3} = D_{1,2} \cup D_3$ .

El algoritmo 2.1 sintetiza lo anteriormente dicho; en el paso de inicialización (línea 1), un conjunto inicial de agrupamientos locales  $D$  se obtiene por cualquier técnica de segmentación convencional. La línea 4 selecciona el par de agrupamientos con la razón de verosimilitud  $R$  mayor de acuerdo con (2.13); si dicha razón es mayor que uno<sup>1</sup> la fusión se realiza (líneas 6 a 10). Este proceso se itera hasta obtener el mejor número de líneas (agrupamientos).

**Ejemplo 2.1** *Detección de líneas a partir de una secuencia sintética de mediciones de rango.* La Fig. 2.4(a) muestra un entorno sintético<sup>2</sup> con siete segmentos; cuatro

<sup>1</sup>lo que significa que es mejor fusionar los agrupamientos que mantenerlos separados

<sup>2</sup>Los entornos sintéticos son modelos matemáticos generados por computadora que son muy útiles porque podemos contrastar los resultados de un algoritmo con los parámetros con los que se originó el modelo

**Algoritmo 2.1:** Algoritmo Propuesto**Entrada:** Una secuencia de mediciones  $S = \{\mathbf{p}_1, \dots, \mathbf{p}_N\}$ **Salida:** Un conjunto de líneas  $\mathbf{Z} = \{\mathbf{z}_1, \dots, \mathbf{z}_\eta\}$  y el agrupamiento de mediciones asociadas a cada línea  $D = \{D_1, \dots, D_\eta\}$ 

1 Encontrar un conjunto de agrupamientos locales

 $D = \{D_1, \dots, D_\eta\} \mid (\forall i \neq j, D_i \cap D_j = \{\}) \wedge (\bigcup_{i=1 \dots \eta} D_i \subseteq S)$  con alguna técnica de detección convencional (tal como IEPF o LT)2 **repetir**3     **si**  $\eta > 1$  **entonces**4         Identificar los agrupamientos  $D_a, D_b \in D, D_a \neq D_b$  con la razón de verosimilitud  $R(D_a, D_b)$  más alta5          $R^* \leftarrow R(D_a, D_b)$ 6         **si**  $R^* > 1$  **entonces**7              $D_a \leftarrow D_a \cup D_b$ 8              $D \leftarrow D \setminus \{D_b\}$ 9              $\eta \leftarrow \eta - 1$ 10         **finSi**11     **finSi**12 **mientras**  $(R^* > 1) \wedge (\eta > 1)$ 13  $\mathbf{Z} \leftarrow \{\mathbf{z}_{j^*} \mid j = 1, \dots, \eta\}$  donde  $\mathbf{z}_{j^*} = [r_{j^*}, \alpha_{j^*}]^\top$  es la mejor línea para el grupo  $D_j$  en términos de mínimos cuadrados14 **regresar**  $\mathbf{Z}, D$ 

segmentos interesantes están marcados con flechas: los segmentos  $s_1$  y  $s_2$  se han originado en la misma línea mientras que los segmentos marcados como  $s_3$  y  $s_4$  se originaron de dos líneas paralelas (separadas 50mm. entre sí). A partir de la Fig. 2.4(a) se obtienen las mediciones sintéticas mostradas en la Fig. 2.4(b). Las mediciones sintéticas se han contaminado con 0.1 de las mediciones son valores atípicos (que no siguen ninguna línea); mientras que el resto de las mediciones se les agrega ruido gaussiano con varianza  $\sigma = 1[\text{cm}]$ .

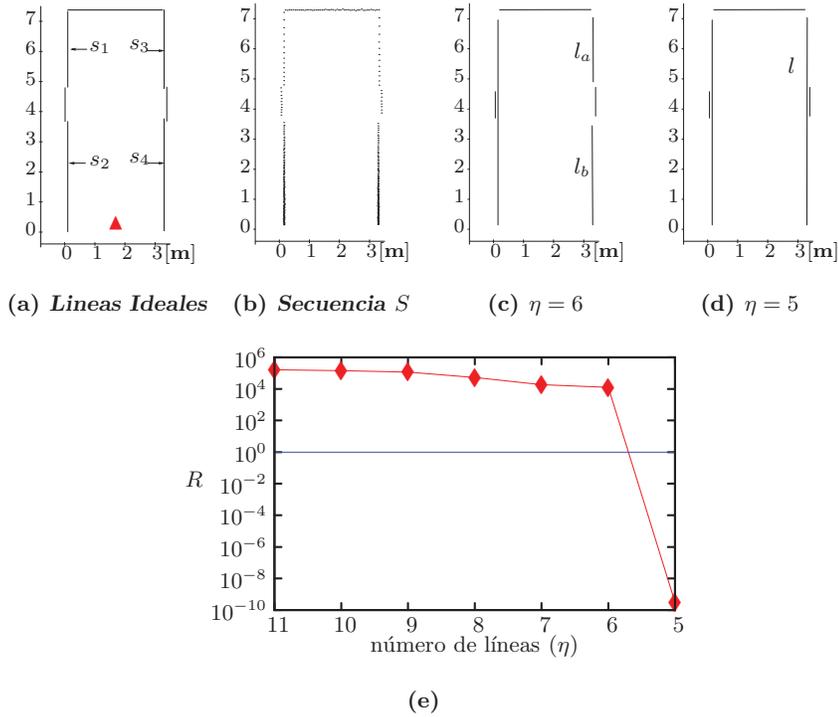


Figura 2.4: Detección del número adecuado de líneas a partir de un ambiente simulado. (a) Ambiente simulado; el triángulo representa la posición del telémetro láser. (b) las mediciones obtenidas por un telémetro simulado con un ruido gaussiano de  $\sigma = 10$  mm). (c y d) las líneas resultantes mediante el uso de LRT, la fusión de los puntos de  $l_a$  y  $l_b$  ( $\eta = 6$ ) en un agrupamiento único ( $\eta = 5$ ) no es favorable ya que  $R = 3.106 \times 10^{-10}$ . (e) El diagrama de  $R$  respecto del número de líneas  $\eta$ .

Solución. Después de llevar a cabo el análisis con el algoritmo 2.1, se obtienen los valores de  $R$  representados por rombos en la Fig. 2.4(e). Cada rombo en la Fig. 2.4(e) representa el mejor valor de  $R$  para un determinado valor de  $\eta$ . Un cambio considerable se detecta<sup>3</sup> desde  $\eta = 6$  a  $\eta = 5$ , y el valor de  $R$  para  $\eta = 5$  es menor que 1 (lo que significa que hay  $\eta = 6$  líneas en el ambiente). Note que el entorno original, efectivamente contiene seis líneas: dos de ellas son paralelas y están muy próximas entre sí (50 [mm]) como se muestra en la Fig. 2.4(a). Otro experimento realizado se muestra en la Tabla 2.1; aquí se ha variado el valor<sup>4</sup> de  $\sigma$  desde el valor nominal de 10 [mm] hasta 30 [mm]. Se observa que para  $\sigma = 16$  aún se

<sup>3</sup> $\eta = 6$  significa que se ha contrastado el modelo de 7 líneas contra el modelo de 6 líneas; si  $R \geq 1$  se ha optado por el modelo con menos líneas.

<sup>4</sup>Aunque el valor de  $\sigma$  se conoce *a priori* este experimento se realizó para saber en qué grado el método propuesto es tolerante ante la variación de este parámetro

Tabla 2.1: Resultado de la prueba LRT para diferentes valores de  $\sigma$ .

$\sigma$	Num. Lín. detectadas	$R$		
		de 7 a 6	de 6 a 5	de 5 a 4
10	6	12551.40	$3.106 \times 10^{-10}$	*
12	6	9730.02	$1.154 \times 10^{-5}$	*
14	6	7638.94	$6.018 \times 10^{-3}$	*
16	6	6105.91	0.327	*
18	5	4968.97	4.789	$3.888 \times 10^{-56}$
20	5	4110.76	31.347	$9.260 \times 10^{-45}$
30	5	1964.51	1920.756	$5.976 \times 10^{-18}$

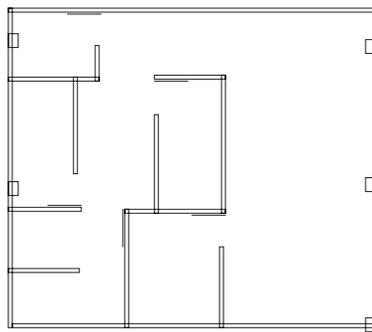


Figura 2.5: Entorno simulado

detectan 6 líneas y es hasta  $\sigma = 18$  que con nuestro criterio se encuentran 5 líneas.  $\square$

## 2.5. Resultados experimentales

Para probar las ideas propuestas en este capítulo se generaron secuencias de mediciones a partir del entorno sintético de la Fig. 2.5 con la configuración que se muestra en la Tabla 2.2.

Se seleccionaron tres algoritmos: el SR (del inglés *Sequential RANSAC*) con generación local de hipótesis [Schnabel07], El LT (del inglés *Line Tracking*) [Duda73] y SM (del inglés *Split and Merge*) [Duda73]. Además del criterio convencional de fusión basada en distancia, los algoritmos SM y LT se probaron con la razón de verosimilitud propuesta; las versiones que usan esta razón están marcadas en la Tabla 2.3 como SM+LRT y LT+LRT, respectivamente.

Tabla 2.2: Parámetros para generar datos sintéticos

núm. líneas	42
distancia mínima entre líneas paralelas próximas	70[mm]
resolución angular del láser	0.5°
modelo de mediciones	0.2 ruido uniforme + 0.8 gaussiano con desviación 1[cm] (direccional)
número de posiciones del robot (secuencias de medición láser)	400

Tabla 2.3: Comparación de algoritmos de detección de líneas a partir de datos sintéticos

	VP %	ND %	$\mu_{err_r}$ [mm]	$\mu_{err_\alpha}$ [rad]	velocidad [ms]
SR	91.29	21.32	6.56	0.0116	5.088
SM	75.86	24.13	6.68	0.0143	3.051
SM + LRT	95.38	12.70	4.37	0.0062	31.682
LT	89.97	18.22	4.92	0.0077	16.770
LT+LRT	96.82	13.20	3.95	0.0055	58.207

La Tabla 2.3 muestra los resultados experimentales; la primera columna muestra la proporción de verdaderos positivos (VP), es decir, la proporción de líneas detectadas que corresponden a líneas en el entorno. La segunda columna es la proporción de líneas no detectadas (ND), es decir, líneas donde se obtuvieron mediciones sintéticas pero que no fueron detectadas por el algoritmo. La proporción de líneas no detectadas es alta en general debido a la presencia de líneas pequeñas en el entorno (Fig. 2.5).

Las columnas marcadas con  $\mu_{err_r}$  y  $\mu_{err_\alpha}$  muestran las medias de los errores de  $r$  y  $\alpha$ , respectivamente; note que las versiones que utilizan la prueba de razón de verosimilitud propuesta (LRT) obtienen mejoras significativas. La última columna de la Tabla 2.3 muestra la velocidad de los algoritmos<sup>5</sup>; como era de esperarse, la técnica propuesta es más costosa computacionalmente.

<sup>5</sup>En una notebook con procesador Intel Core 2 Duo P8600 de 2.4 GHz.

## 2.6. Discusión

Este capítulo introduce un nuevo método para encontrar los mejores agrupamientos lineales a partir de las mediciones del telémetro láser. El algoritmo busca agrupamientos locales y luego usa un criterio probabilista – el LRT– para decidir cuando fusionar los agrupamientos. El criterio se establece como una razón entre verosimilitudes de dos hipótesis; con lo que se decide qué modelos se deben fusionar y como consecuencia se obtiene el número de marcas presentes en el entorno.

El planteamiento presentado invierte mayor tiempo de procesamiento que otras técnicas; sin embargo, este tiempo es insignificante en comparación con la complejidad temporal en que se puede incurrir al resolver el problema del SLAM cuando el número de marcas es muy grande. A cambio de esto, la propuesta tiene las siguientes ventajas sobre otros esquemas:

1. En contraste con otros criterios, evita usar umbrales innecesarios, ya que únicamente usa el modelo de medición (parámetro  $\sigma$ ); por ende, la implementación del algoritmo es directa y no hay necesidad de *afinar* parámetros del algoritmo.
2. La propuesta establece cuándo fusionar dos agrupamientos o no, esto se logra comparando las hipótesis  $H_1$  y  $H_2$ . Esto contrasta con la distancia de Mahalanobis que especifica una probabilidad de aceptación entre dos marcas pero no la probabilidad de rechazo.
3. Aunque en el presente capítulo no se ha estudiado, cuando  $R \approx 1$  la decisión entre las hipótesis  $H_1$  y  $H_2$  es incierta. Esta situación se debe estudiar de manera más detenida ya que puede ayudar a manejar de forma más adecuada las marcas para algoritmos de SLAM; y
4. Los resultados experimentales muestran que el enfoque propuesto es confiable ya que encuentra correctamente el número de líneas presentes en el entorno y además el número de falsos positivos se mantiene bajo; por lo que este enfoque es preferible sobre el enfoque de mezclado usando una distancia entre los segmentos. Como se verá más adelante este algoritmo ayuda a extraer líneas muy confiables a pesar de

encontrarse en entornos con muchos objetos e incluso se puede generar el mapa de líneas aún en la presencia de personas.

La aplicación de la técnica LRT a la detección de otros modelos en dos y tres dimensiones es directa; sin embargo, surgen preguntas de investigación interesantes: i) ¿Es posible aplicar esta técnica para detectar diferentes tipos de modelos al mismo tiempo (*v.g.* encontrar círculos, líneas y agrupamientos a partir de una secuencia de mediciones)?, ii) ¿Qué resultados se obtendrán si se usa otro tipo de sensores?, iii) ¿Es posible reemplazar el proceso preliminar de agrupamiento por una técnica probabilista?; y, iv) ¿Conviene desarrollar otros modelos que no consideren ruido gaussiano en las mediciones?

El siguiente capítulo estudia el algoritmo de SLAM estocástico para generar mapas de líneas de entornos interiores; el Algoritmo 2.1 se usa para detectar líneas locales de forma robusta y mejorar el desempeño general del algoritmo de SLAM.

## Capítulo 3

# SLAM estocástico

Como se describe en el Capítulo 1, el problema de SLAM en línea consiste en calcular la posición más reciente del robot y el mapa del entorno. Una de las soluciones más populares a este problema se conoce como SLAM estocástico; algunos de los precursores de este algoritmo son Smith y Chesseman [Smith87], y Leonard y Durrant–Whyte [Leonard91].

El SLAM estocástico usa marcas para representar el entorno y se basa en las ecuaciones del EKF (del inglés *Extended Kalman Filter*). Las ecuaciones del KF (Apéndice B) tratan de predecir el estado de un sistema dinámico lineal. Análogamente, las ecuaciones del EKF predicen el estado de un sistema dinámico no lineal usando una aproximación de Taylor.

El SLAM estocástico formaliza algunas ideas intuitivas: cuando el robot se mueve de una posición a otra, se usa la información odométrica para estimar la nueva posición (fase de predicción); cuando el robot obtiene una medición de una marca que no está en el mapa, entonces la marca se agrega al mapa (fase de agregación); por último, si el robot obtiene una medición de una marca que ya está representada en el mapa, la nueva medición se usa para obtener una mejor estimación de los parámetros de la marca encontrada y la posición del robot (fase de actualización). Para mantener las cosas simples, las fases de agregación y actualización operan con una marca a la vez. Esto significa que si desde una posición estática el robot encuentra más de una marca, cada marca se procesa de forma individual.

Este capítulo se aborda desde una perspectiva práctica: se implementa el SLAM estocástico para obtener un mapa de líneas. Se ha elegido la representación de líneas como caso de estudio porque es útil para aplicaciones en entornos interiores.

El resto del capítulo se organiza como sigue: la notación usada se describe en la Sección 3.1; el algoritmo de SLAM estocástico para obtener mapas de líneas se describe en la Sección 3.2; los resultados experimentales del caso de estudio se muestran en la Sección 3.3; finalmente, la Sección 3.4 discute algunas dificultades del algoritmo de SLAM estocástico y contrasta el algoritmo de SLAM estocástico con otros algoritmos probabilistas.

### 3.1. Notación

En la literatura especializada se usan diferentes notaciones, a continuación se explica la usada en este capítulo.

Las acciones de control del robot para el tiempo  $i = \{0, \dots, M\}$  se representan por

$$\mathbf{u}_i = \begin{bmatrix} u_{i_x} \\ u_{i_y} \\ u_{i_\theta} \end{bmatrix}. \quad (3.1)$$

El estado del sistema en el tiempo  $i = \{0, \dots, M\}$  se representa como

$$\tau_i = \begin{bmatrix} \mathbf{x}_i \\ \mathbf{m}_i \end{bmatrix}, \quad (3.2)$$

donde  $\mathbf{x}_i$  indica la posición del robot en el tiempo, y  $\mathbf{m}_i$  es el mapa del entorno. Para representar la posición del robot se requiere una cantidad fija de variables así que podemos expresar que  $\forall i \in \{1 \dots M\} |\mathbf{x}_i| = |\mathbf{x}_{i-1}|$ . El número de variables depende de los grados de libertad del robot, para el caso de un robot que se mueve en un plano

$$\mathbf{x}_i = \begin{bmatrix} x_i \\ y_i \\ \theta_i \end{bmatrix} \quad (3.3)$$

donde  $(x_i, y_i)$  son las coordenadas del robot y  $\theta_i$  es su orientación (ángulo respecto del eje  $x$ ). En cambio, el tamaño del mapa  $\mathbf{m}_i$  se incrementará cuando se descubran marcas nuevas en el entorno; es decir,  $\forall i \in \{1 \dots M\} |\mathbf{m}_i| \geq |\mathbf{m}_{i-1}|$ . Un mapa de líneas en el plano se representa como

$$\mathbf{m}_i = [r_1, \alpha_1, \dots, r_j, \alpha_j, \dots, r_{N_i}, \alpha_{N_i}]^T, \quad (3.4)$$

donde  $[r_j, \alpha_j]^T$  son los parámetros de la  $j$ -ésima línea en su forma polar y  $N_i$  es el número de marcas detectadas hasta el  $i$ -ésimo momento. El filtro de Kalman [Kalman60, Kalman61] asume que las variables aleatorias siguen una distribución gaussiana; y las variables de interés se representan por su media y covarianza. En lo sucesivo, la media y covarianza del  $i$ -ésimo estado  $\tau_i$  se denotan como

$$\hat{\tau}_i = \begin{bmatrix} \hat{\mathbf{x}}_i \\ \hat{\mathbf{m}}_i \end{bmatrix} \quad (3.5)$$

$$\mathbf{C}_i = \begin{bmatrix} \mathbf{C}_{[\mathbf{x}]_i} & \mathbf{C}_{[\mathbf{xm}]_i} \\ \mathbf{C}_{[\mathbf{xm}]_i}^T & \mathbf{C}_{[\mathbf{m}]_i} \end{bmatrix} \quad (3.6)$$

donde  $\mathbf{C}_{[\mathbf{x}]_i}$  es la covarianza de la  $i$ -ésima posición del robot,  $\mathbf{C}_{[\mathbf{m}]_i}$  es la covarianza del mapa y  $\mathbf{C}_{[\mathbf{xm}]_i}$  es la covarianza del mapa con respecto de la posición del robot.

## 3.2. Descripción del algoritmo

El SLAM estocástico [Smith87, Leonard91, Bailey02] se describe en el Algoritmo 3.1, en la  $i$ -ésima iteración (líneas 2 a 18) se procesa la acción de control  $\mathbf{u}_i$  y el conjunto de líneas  $\mathbf{Z}_i$  (este conjunto de líneas se puede obtener con el algoritmo que se discute en el Capítulo 2); al finalizar cada iteración se obtiene el nuevo estimado del sistema  $\hat{\tau}$  (que incluye la nueva posición del robot y el mapa). Al principio, se ejecuta la inicialización de variables (línea 4); la posición inicial  $\hat{\mathbf{x}}_0 = [\mathbf{0}]$  significa que se considera la primera posición como origen del sistema global. Si no es la primera posición, entonces se predice la nueva posición con la acción de control  $\mathbf{u}_i$  (línea 6).

La fase de asociación (línea 8) relaciona las nuevas líneas  $\mathbf{Z}_i$  con las ya descubiertas en el mapa; este problema se estudia en el Capítulo 4. Finalmente, de acuerdo con la

**Algoritmo 3.1:** SLAM estocástico

**Entrada:** Las acciones de control  $U = [\mathbf{u}_1, \mathbf{u}_2, \dots, \mathbf{u}_M]$  y las marcas descubiertas  $Z = [\mathbf{Z}_0, \mathbf{Z}_1, \dots, \mathbf{Z}_M]$  en cada posición  $0, 1, \dots, M$

**Salida:** El mapa del entorno y la última posición del robot

```

1   $i \leftarrow 0$ 
2  repetir
3      si  $i = 0$  entonces
4           $\hat{\tau}_i^- \leftarrow \begin{bmatrix} \mathbf{0} \\ \epsilon \end{bmatrix}, \mathbf{C}_i^- \leftarrow \begin{bmatrix} \mathbf{0} \end{bmatrix}$  // inicialización (Sec. 3.2.1)
5      sino
6           $\hat{\tau}_i^- \leftarrow \mathbf{g}(\hat{\tau}_{i-1}, \mathbf{u}_i)$ , calcula  $\mathbf{C}_i^-$  // predicción (Sec. 3.2.2)
7      finSi
8      Encontrar la asociación  $A \leftarrow \{(k, f_a(k)) \mid k = 1, \dots, |\mathbf{Z}_i|\}$  // (Cap.4)
9      para cada  $\mathbf{z}_k \in \mathbf{Z}_i$  hacer
10         si esNueva( $\mathbf{z}_k$ ) entonces
11              $\hat{\tau}_i^+ \leftarrow \mathbf{g}_a(\hat{\tau}_i, \mathbf{z}_k)$  // agregación (Sec.3.2.3)
12         sino
13              $\hat{\tau}_i^+ \leftarrow \mathbf{h}_u(\hat{\tau}_i, \mathbf{z}_k, (k, f_a(k)))$  // actualización (Sec. 3.2.4)
14         finSi
15         Calcula  $\mathbf{C}_i^+$ 
16     finPara
17      $i \leftarrow i + 1$ 
18 mientras  $i \leq M$ 
19 regresar  $\hat{\tau}_i$ 

```

asociación  $A$  cada línea o se agrega al mapa (línea 11) o actualiza una línea previamente encontrada (línea 13).

Las siguientes secciones describen cada fase del algoritmo; note que se usa  $\hat{\tau}^-$  para indicar un estimado del estado en la fase de predicción y  $\hat{\tau}^+$  para indicar un estimado en la fase de actualización.

### 3.2.1. Inicialización

Si se considera la primera posición del robot como referencia para el resto del mapa entonces  $\mathbf{x}_0 = \mathbf{0}$ . Note que inicialmente no se ha detectado ninguna marca, por lo que el estado inicial y su covarianza son

$$\hat{\tau}_0 = \begin{bmatrix} \hat{\mathbf{x}}_0 \\ \epsilon \end{bmatrix} = \begin{bmatrix} \mathbf{0} \\ \epsilon \end{bmatrix}, \text{ y } \mathbf{C}_0 = \begin{bmatrix} \mathbf{0} \end{bmatrix}, \quad (3.7)$$

respectivamente; aquí,  $\epsilon$  indica que aún no se introducido ninguna marca al mapa.

### 3.2.2. Predicción

Esta fase usa el modelo de movimiento para estimar la siguiente posición del robot. El modelo de movimiento depende de la arquitectura del robot y tiene la forma

$$\mathbf{x}_i = \mathbf{f}(\mathbf{x}_{i-1}, \mathbf{u}_i), \quad (3.8)$$

donde  $\mathbf{x}_{i-1}$  es la posición anterior,  $\mathbf{x}_i$  es la posición nueva, y  $\mathbf{u}_i$  es la información odométrica.

La Fig. 3.1 ilustra el caso de un robot que se mueve en un plano, de aquí se deriva que (3.8) se puede expresar como

$$\mathbf{x}_i = \mathbf{f}(\mathbf{x}_{i-1}, \mathbf{u}_i) = \begin{bmatrix} x_{i-1} + u_{ix} \cos \theta_{i-1} - u_{iy} \sin \theta_{i-1} \\ y_{i-1} + u_{ix} \sin \theta_{i-1} + u_{iy} \cos \theta_{i-1} \\ \theta_{i-1} + u_{i\theta} \end{bmatrix}. \quad (3.9)$$

La fase de predicción usa (3.9) para calcular una primera estimación<sup>1</sup> del  $i$ -ésimo estado denotado por  $\hat{\tau}_i^-$ ,

$$\hat{\tau}_i^- = \mathbf{g}(\hat{\tau}_{i-1}, \mathbf{u}_i) = \begin{bmatrix} \mathbf{f}(\hat{\mathbf{x}}_{i-1}, \mathbf{u}_i) \\ \hat{\mathbf{m}} \end{bmatrix} \quad (3.10)$$

$$= \begin{bmatrix} \hat{x}_{i-1} + u_{ix} \cos \hat{\theta}_{i-1} - u_{iy} \sin \hat{\theta}_{i-1} \\ \hat{y}_{i-1} + u_{ix} \sin \hat{\theta}_{i-1} + u_{iy} \cos \hat{\theta}_{i-1} \\ \hat{\theta}_{i-1} + u_{i\theta} \\ \hat{\mathbf{m}} \end{bmatrix} \quad (3.11)$$

<sup>1</sup>Se logran mejores estimaciones al refinar el valor de  $\hat{\tau}^-$  en futuras fases de actualización.

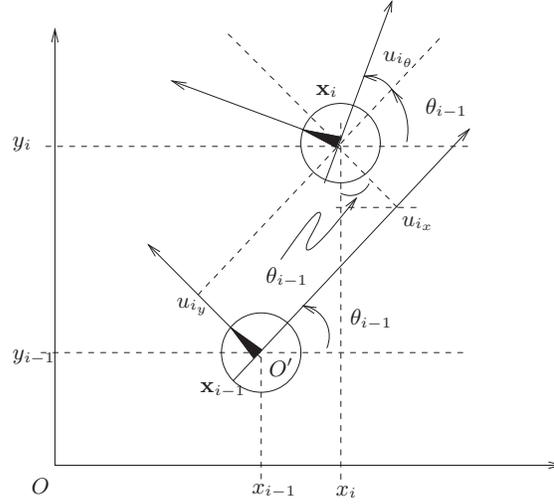


Figura 3.1: Modelo de movimiento para un robot en un entorno bidimensional.

Dado que  $\hat{\tau}_i^-$  es una función no lineal de  $\hat{\mathbf{x}}_i$  y  $\mathbf{u}_i$ , la covarianza en la fase de predicción se calcula<sup>2</sup> como:

$$\mathbf{C}_i^- = \mathbf{G}_x \mathbf{C}_{i-1} \mathbf{G}_x^T + \mathbf{G}_u \mathbf{Q} \mathbf{G}_u^T, \quad (3.12)$$

donde  $\mathbf{Q}$  es la covarianza del movimiento  $\mathbf{u}_i$ ,

$$\mathbf{G}_x = \left. \frac{\partial \mathbf{g}}{\partial \mathbf{x}} \right|_{(\hat{\tau}_{i-1}, \mathbf{u}_i)} = \begin{bmatrix} \mathbf{F}_x & \mathbf{0}_{[xm]} \\ \mathbf{0}_{[xm]}^T & \mathbf{I} \end{bmatrix} \quad (3.13)$$

$$\mathbf{G}_u = \left. \frac{\partial \mathbf{g}}{\partial \mathbf{u}} \right|_{(\hat{\tau}_{i-1}, \mathbf{u}_i)} = \begin{bmatrix} \mathbf{F}_u \\ \mathbf{0}_{[xm]}^T \end{bmatrix}, \quad (3.14)$$

y

$$\mathbf{F}_x = \left. \frac{\partial \mathbf{f}}{\partial \mathbf{x}} \right|_{(\hat{\mathbf{x}}_{i-1}, \mathbf{u}_i)} = \begin{bmatrix} 1 & 0 & -u_{ix} \sin \hat{\theta}_{i-1} - u_{iy} \cos \hat{\theta}_{i-1} \\ 0 & 1 & u_{ix} \cos \hat{\theta}_{i-1} - u_{iy} \sin \hat{\theta}_{i-1} \\ 0 & 0 & 1 \end{bmatrix} \quad (3.15)$$

$$\mathbf{F}_u = \left. \frac{\partial \mathbf{f}}{\partial \mathbf{u}} \right|_{(\hat{\mathbf{x}}_{i-1}, \mathbf{u}_i)} = \begin{bmatrix} \cos \hat{\theta}_{i-1} & -\sin \hat{\theta}_{i-1} & 0 \\ \sin \hat{\theta}_{i-1} & \cos \hat{\theta}_{i-1} & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (3.16)$$

<sup>2</sup>Ver Apéndice B.

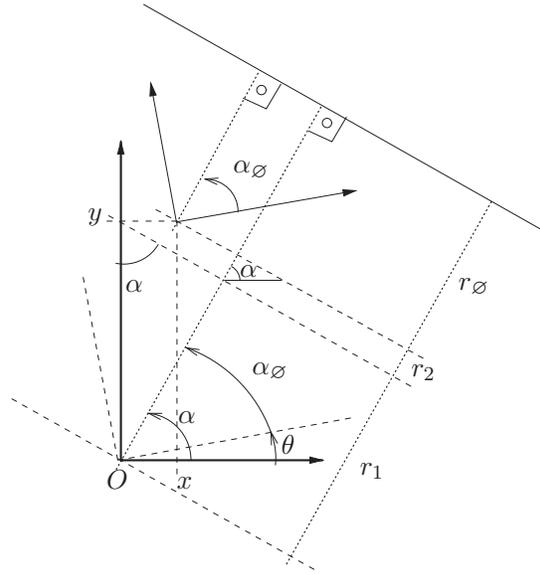


Figura 3.2: Transformación de los parámetros de una línea de la referencia del robot a la referencia global.

La ecuación 3.12 se puede implementar de forma eficiente como

$$\mathbf{C}_i^- = \begin{bmatrix} \mathbf{G}_x \mathbf{C}_{[x]_{i-1}} \mathbf{G}_x^T + \mathbf{G}_u \mathbf{Q} \mathbf{G}_u^T & \mathbf{G}_x \mathbf{C}_{[xm]_{i-1}} \\ (\mathbf{G}_x \mathbf{C}_{[xm]_{i-1}})^T & \mathbf{C}_{[m]_{i-1}} \end{bmatrix}. \quad (3.17)$$

### 3.2.3. Agregación

El robot explora su entorno para detectar marcas que no se pueden percibir desde una posición fija debido a oclusiones o a limitaciones sensoriales. Al explorar el entorno, el robot puede encontrar marcas inéditas (marcas que no se habían detectado antes). La fase de agregación incluye las nuevas marcas en el mapa.

Sea  $\mathbf{z}_\emptyset = [r_\emptyset, \alpha_\emptyset]^T$ , una nueva línea (que no tiene una línea asociada en el mapa). Note que las marcas almacenadas en el mapa tienen  $\mathbf{x}_0$  como referencia común, entonces  $\mathbf{z}_\emptyset$  se debe transformar en la misma referencia para almacenarse en la variable de estado.

La operación de composición,  $\mathbf{l} = \mathbf{x} \oplus \mathbf{z}_\emptyset$ ; transforma la nueva medición del sistema de coordenadas del robot, al sistema de coordenadas global. Como se puede observar en la

Fig. 3.2, la composición se obtiene como

$$\begin{aligned} \mathbf{l} &= \begin{bmatrix} r \\ \alpha \end{bmatrix} = \begin{bmatrix} r_{\emptyset} + r_1 + r_2 \\ \alpha_{\emptyset} + \theta \end{bmatrix} \\ &= \begin{bmatrix} r_{\emptyset} + y \sin(\alpha_{\emptyset} + \theta) + x \cos(\alpha_{\emptyset} + \theta) \\ \alpha_{\emptyset} + \theta \end{bmatrix}. \end{aligned} \quad (3.18)$$

Para integrar la medición  $\mathbf{z}_{\emptyset}$  en el estado  $\tau_i$  primero se genera un estado temporal  $\tau_i^{\text{tmp}}$ ; a continuación, el estado temporal se transforma con (3.18) para obtener el estado con la marca en la referencia adecuada [Bailey02]. Esta técnica permite calcular la covarianza del estado después de agregación. Procediendo de esa manera, el estado temporal es

$$\hat{\tau}_i^{\text{tmp}} = \begin{bmatrix} \hat{\tau}_i \\ \mathbf{l}_{\emptyset} \end{bmatrix} = \begin{bmatrix} \hat{\tau}_i \\ r_{\emptyset} \\ \alpha_{\emptyset} \end{bmatrix} \quad (3.19)$$

con,

$$\mathbf{C}_i^{\text{tmp}} = \begin{bmatrix} \mathbf{C}_{[\mathbf{x}]_i} & \mathbf{C}_{[\mathbf{xm}]_i} & \mathbf{0} \\ \mathbf{C}_{[\mathbf{mx}]_i} & \mathbf{C}_{[\mathbf{m}]_i} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{R} \end{bmatrix}, \quad (3.20)$$

donde  $\mathbf{R}$  es la covarianza de la medición  $\mathbf{z}_{\emptyset}$ . El estado después de la agregación se calcula como

$$\hat{\tau}_i^+ = \mathbf{g}_a(\hat{\mathbf{x}}, \mathbf{z}_{\emptyset}) \quad (3.21)$$

$$= \mathbf{g}_a(\hat{\tau}_i^{\text{tmp}}) = \begin{bmatrix} \hat{\tau}_i \\ r_{\emptyset} + y \sin(\alpha_{\emptyset} + \theta_i) + x \cos(\alpha_{\emptyset} + \theta_i) \\ \alpha_{\emptyset} + \theta_i \end{bmatrix} \quad (3.22)$$

$$\mathbf{C}_i^+ = \mathbf{G}_{\text{tmp}} \mathbf{C}_i^{\text{tmp}} \mathbf{G}_{\text{tmp}}^{\text{T}} \quad (3.23)$$

donde

$$\mathbf{G}_{\text{tmp}} = \left. \frac{\partial \mathbf{g}}{\partial \tau_i^{\text{tmp}}} \right|_{\hat{\tau}_i^{\text{tmp}}} = \begin{bmatrix} \mathbf{I}_x & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{I}_m & \mathbf{0} \\ \mathbf{G}_1 & \mathbf{0} & \mathbf{G}_2 \end{bmatrix} \quad (3.24)$$

y,

$$\mathbf{G}_1 = \begin{bmatrix} \cos(\alpha_\emptyset + \theta_i) & \sin(\alpha_\emptyset + \theta_i) & y_i \cos(\alpha_\emptyset + \theta_i) - x_i \sin(\alpha_\emptyset + \theta_i) \\ 0 & 0 & 1 \end{bmatrix}, y \quad (3.25)$$

$$\mathbf{G}_2 = \begin{bmatrix} 1 & y_i \cos(\alpha_\emptyset + \theta_i) - x_i \sin(\alpha_\emptyset + \theta_i) \\ 0 & 1 \end{bmatrix}. \quad (3.26)$$

Una forma simplificada de resolver (3.23) es

$$\mathbf{C}_i^+ = \begin{bmatrix} \mathbf{C}_{[x]_i} & \mathbf{C}_{[xm]_i} & \mathbf{C}_{[x]_i} \mathbf{G}_1^\top \\ \mathbf{C}_{[xm]_i}^\top & \mathbf{C}_{[m]_i} & \mathbf{C}_{[xm]_i}^\top \mathbf{G}_1^\top \\ \mathbf{G}_1 \mathbf{C}_{[x]_i} & \mathbf{G}_1 \mathbf{C}_{[xm]_i} & \mathbf{G}_1 \mathbf{C}_{[x]_i} \mathbf{G}_1^\top + \mathbf{G}_2 \mathbf{R} \mathbf{G}_2^\top \end{bmatrix}. \quad (3.27)$$

### 3.2.4. Actualización

Suponga que el robot obtiene una medición de una marca y que se conoce que dicha marca ya existe en el mapa. Esto significa que se conoce que la  $k$ -ésima medición es la misma que la  $f_a(k)$ -ésima marca del mapa o en otras palabras se ha encontrado la asociación  $(k, f_a(k))$ . Intuitivamente, esta información es útil para obtener una nueva estimación de los valores de la marca y corregir errores de la posición actual.

En general, la medición y la marca en el mapa tienen coordenadas de referencia distintas: la medición tiene como referencia el robot mientras que la marca tiene referencia global. Por tanto, se desea transformar sus parámetros a un sistema común. Esta transformación se describe matemáticamente en el *modelo de medición*. Los parámetros de cómo se debería percibir una marca almacenada estarán en función de la posición del robot. Dichos parámetros se derivan de (3.18), y son

$$\begin{aligned} \mathbf{h}(\mathbf{x}_i, \mathbf{l}_j) &= \mathbf{l}_\emptyset = \begin{bmatrix} r_\emptyset \\ \alpha_\emptyset \end{bmatrix} \\ &= \begin{bmatrix} r_j - y_i \sin \alpha_j - x_i \cos \alpha_j \\ \alpha_j - \theta_i \end{bmatrix} \end{aligned} \quad (3.28)$$

donde  $\mathbf{x}_i = [x_i, y_i, \theta_i]^\top$ , es la posición del robot; y  $\mathbf{l}_j = [r_j, \alpha_j]^\top$ , es una marca almacenada en el mapa.

Sea  $\mathbf{z}$  una medición obtenida desde la posición  $\mathbf{x}_i$ , y  $\mathbf{l}_j$  su marca correspondiente en el mapa. La innovación se calcula como

$$\mathbf{v} = \mathbf{z} - \mathbf{h}(\mathbf{x}_i, \mathbf{l}_j). \quad (3.29)$$

La ganancia de Kalman (Apéndice B) es,

$$\mathbf{S} = \mathbf{H}_\tau \mathbf{C}_i^- \mathbf{H}_\tau^\top + \mathbf{R} \quad (3.30)$$

$$\mathbf{W} = \mathbf{C}_i^- \mathbf{H}_\tau^\top \mathbf{S}^{-1} \quad (3.31)$$

donde  $\mathbf{R}$  es la covarianza de  $\mathbf{z}$ , y

$$\mathbf{H}_\tau = \left. \frac{\partial \mathbf{h}}{\partial \tau} \right|_{\hat{\tau}_i^-} = \begin{bmatrix} \mathbf{H}_x & \mathbf{0} & \mathbf{H}_l & \mathbf{0} \end{bmatrix} \quad (3.32)$$

$$\mathbf{H}_x = \left. \frac{\partial \mathbf{h}}{\partial \mathbf{x}_x} \right|_{\hat{\tau}_i^-} = \begin{bmatrix} -\cos \hat{\alpha}_j & -\sin \hat{\alpha}_j & 0 \\ 0 & 0 & -1 \end{bmatrix}$$

$$\mathbf{H}_l = \left. \frac{\partial \mathbf{h}}{\partial \mathbf{l}_j} \right|_{\hat{\tau}_i^-} = \begin{bmatrix} 1 & \hat{x}_i \sin \hat{\alpha}_j - \hat{y}_i \cos \hat{\alpha}_j \\ 0 & 1 \end{bmatrix}.$$

La estimación *a posteriori* del estado es

$$\hat{\tau}_i^+ = \mathbf{h}_u(\hat{\tau}_i^-, \mathbf{z}, (k, f_a(k))) \quad (3.33)$$

$$= \hat{\tau}_i^- + \mathbf{W} \mathbf{v} \quad (3.34)$$

$$\mathbf{C}_i^+ = \mathbf{C}_i^- - \mathbf{W} \mathbf{S} \mathbf{W}^\top \quad (3.35)$$

En la siguiente sección se muestran los resultados experimentales del SLAM estocástico para generar mapas de líneas.

### 3.3. Resultados experimentales

El SLAM estocástico presentado en este capítulo se implementó en C++; un resumen de la configuración de la prueba en entorno real se muestra en la Tabla 3.1. Para resolver el problema de asociación se ha usado el algoritmo JCT (del inglés *Joint Compatibility Test*) [Neira01]. Además, se considera que una línea es válida si la suma total de

Tabla 3.1: Configuración usada en la prueba en entorno real

Sensor	telémetro láser SICK LMS-200
Configuración de cada secuencia de medición	apertura de 180° con resolución angular de 0.5° y alcance máximo de 30[m]
número de secuencias de medición	215 (77615 puntos)
modelo del robot	construido por la UMSNH
configuración del robot	dos ruedas con movimiento diferencial y con tercer punto de contacto
odometría	disponible
distancia recorrida	approx. 67.88 metros
movimiento odométrico robot	avance 50[cm], rotación 25.07°
descripción del entorno	laboratorios del posgrado con muchos objetos pequeños, sillas, mesas, cajas, etc.

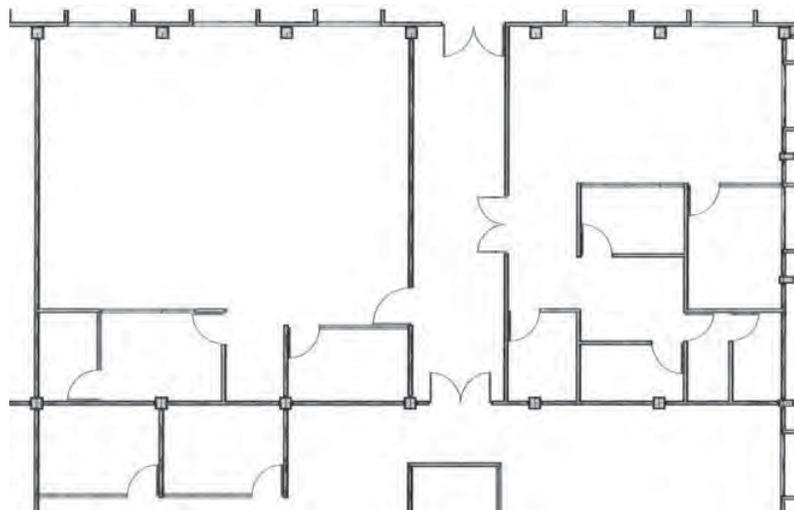
las longitudes de sus segmentos es mayor que 30 cm (esta restricción no es estrictamente necesaria y su único objetivo es mejorar el desempeño del algoritmo).

El mapa ideal del entorno se muestra en la Fig. 3.3a; en la Fig. 3.3b se muestra el registro de las imágenes de rango con una variante de la técnica ICP (del inglés *Iterative Closest Point*) [Romero05]. Esta técnica es muy costosa computacionalmente debido al uso de datos crudos; además, para obtener el mapa total se resuelve el problema encontrando el registro de la secuencia de mediciones actual contra todas las secuencias anteriores. En la Fig. 3.3b se puede apreciar que el entorno es muy complejo porque contiene muchos objetos que no se pueden representar con líneas (*v.g.* sillas, mesas, cables, etc); además tiene lecturas espurias debido a los vidrios y puertas negras<sup>3</sup>.

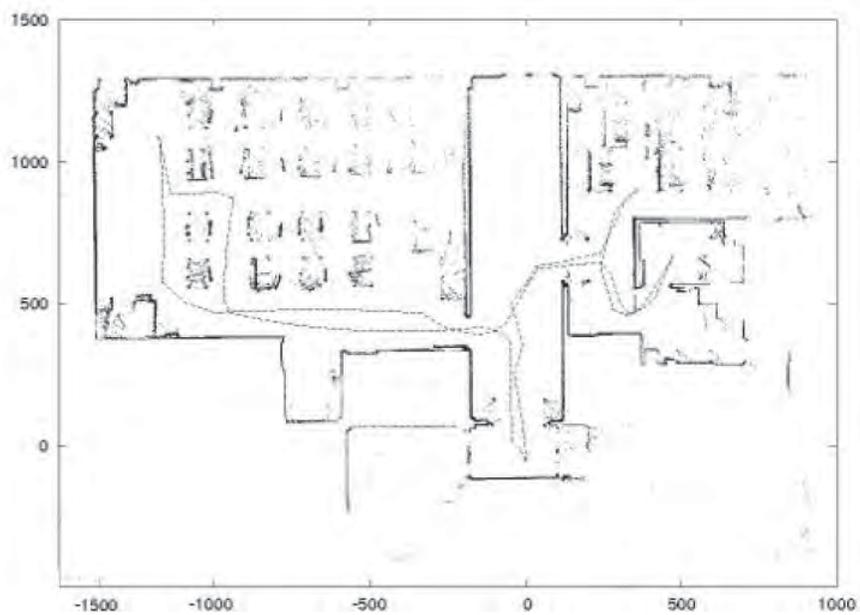
Además de probar la implementación del SLAM estocástico, aquí se desea contrastar el desempeño de los algoritmos de extracción local de líneas que se discuten en el Capítulo 2. La Fig. 3.4 muestra el resultado del SLAM estocástico usando RANSAC secuencial (SR). La Fig. 3.5 muestra dos resultados obtenidos por el algoritmo de división y mezcla: en la Fig. 3.5(a) se usa la distancia euclideana (SM) mientras que en la Fig. 3.5b se usa la prueba LRT (SM+LRT). El resultado de SM + LRT es cualitativamente mejor que el resultado de SR; note por ejemplo que se han detectado correctamente las líneas paralelas correspondientes a paredes.

Análogamente, la Fig. 3.6 presenta los resultados obtenidos usando el algoritmo de

<sup>3</sup>Los objetos oscuros pueden absorber la energía que emite el sensor



(a) *Mapa Ideal*



(b) *Datos crudos registrados con ICP + Lorenziano [Romero05] y la ruta del robot (línea punteada), la escala es en cm.*

Figura 3.3: Mapa del entorno real.

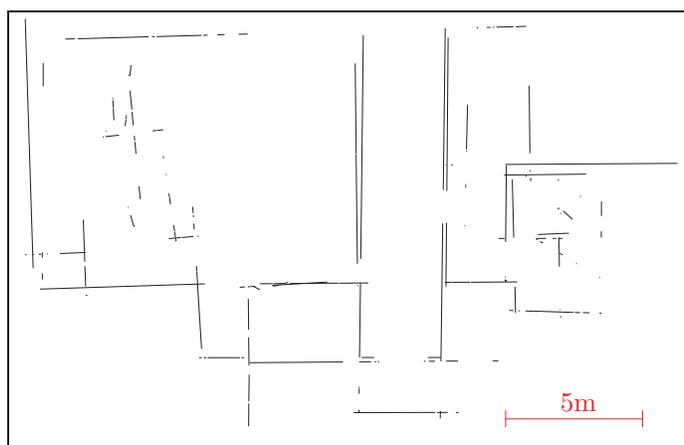


Figura 3.4: Resultados con el algoritmo SR y SLAM estocástico aplicado a una trayectoria de 215 secuencias de mediciones de telémetro obtenidas de nuestro laboratorio.

seguimiento de líneas (LT). En la Fig. 3.6b se observa que se ha obtenido un mejor resultado del algoritmo LT+LRT respecto del algoritmo LT que usa distancia euclídeana (Fig. 3.6a). En la Fig. 3.6a es tal el error inducido que el mapa final tiene líneas duplicadas; este efecto es causado por un error de extracción local que luego impacta en la asociación de marcas.

La Fig. 3.7 contrasta el tiempo<sup>4</sup> que invierte el algoritmo de SLAM estocástico usando dos técnicas de extracción de líneas: SM y SM+LRT. La implementación que usa LRT es más costosa computacionalmente; afortunadamente, la extracción de características únicamente usa secuencias de mediciones locales. Por tanto, no se incrementa el costo con el aumento de marcas del entorno; y en entornos mucho más grandes, el mayor costo se invierte en el manejo del mapa. La implementación actual de la técnica LRT es útil en entornos de mediana escala; ya que en el peor de los casos de esta prueba, el proceso duró 77.0345[ms] con lo que el robot puede generar el mapa en tiempo real si se desplaza a una velocidad inferior a 6.5[m/s].

Note como en las Figuras 3.4, 3.5 y 3.6, las líneas paralelas (vg. puerta – pared) no se han detectado satisfactoriamente; este efecto se debe a una mala asociación; el siguiente capítulo propone un algoritmo que resuelve esta situación.

---

<sup>4</sup>En una notebook con procesador Intel Core 2 Duo P8600 de 2.4 GHz.

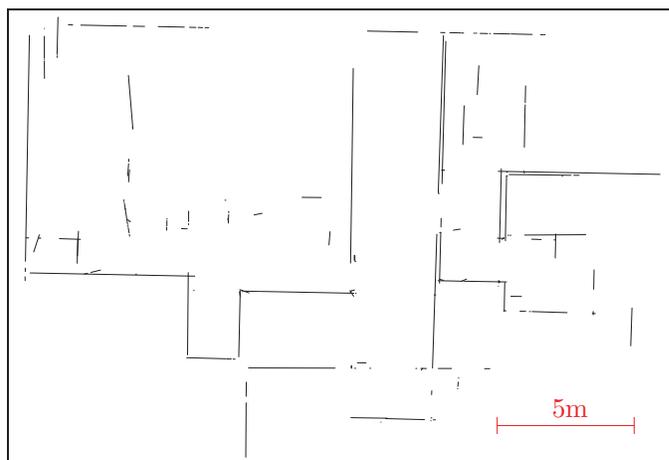
(a) *Split and Merge (SM)*(b) *Split and Merge +Likelihood Ratio Test (SM+LRT)*

Figura 3.5: Comparación entre los algoritmos SM y SM+LRT. Resultados del SLAM estocástico aplicado a una trayectoria de 215 imágenes de rango obtenidas de nuestro laboratorio.

### 3.4. Discusión

El algoritmo de SLAM estocástico es aplicable en problemas en tiempo real con entornos de mediana escala como el mostrado en la Sección 3.3. Desafortunadamente, el algoritmo de SLAM estocástico tiene dos dificultades cuando trata de manejar problemas de gran escala: la no-linealidad y la inversión de matrices. El problema de la no-linealidad ocasiona la acumulación de errores derivados de la aproximación; por otro lado, la necesidad

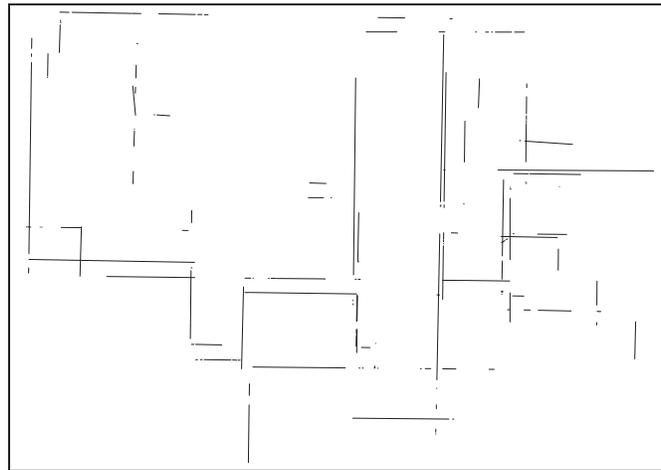
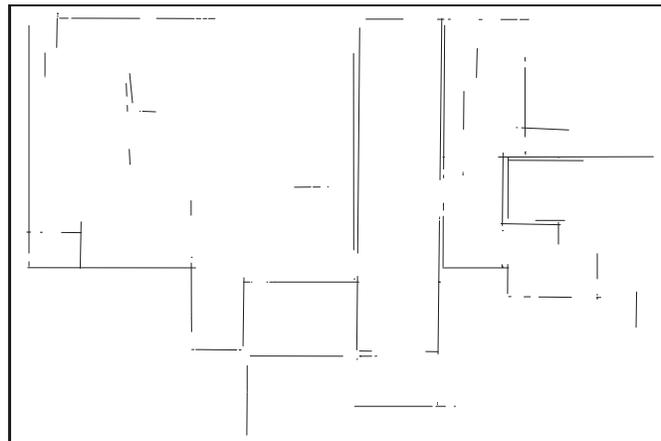
(a) *Line Tracking (LT)*(b) *Line Tracking + Likelihood Ratio Test (LT+LRT)*

Figura 3.6: Comparación entre los algoritmos LT y LT+LRT. Resultados del SLAM estocástico aplicado a una trayectoria de 215 imágenes de rango obtenidas de nuestro laboratorio.

de inversión de matrices ocasiona que se incremente la complejidad temporal a medida que el mapa tiene más marcas.

Estos problemas son más evidentes cuando se tratan de cerrar ciclos de gran longitud. Para resolver estas dificultades se han propuesto diferentes variantes del SLAM estocástico. Por ejemplo se pueden crear mapas locales y después integrarlos en mapas cada vez mas grandes [Tardós02] o usar mapas híbridos [Tomatis08].

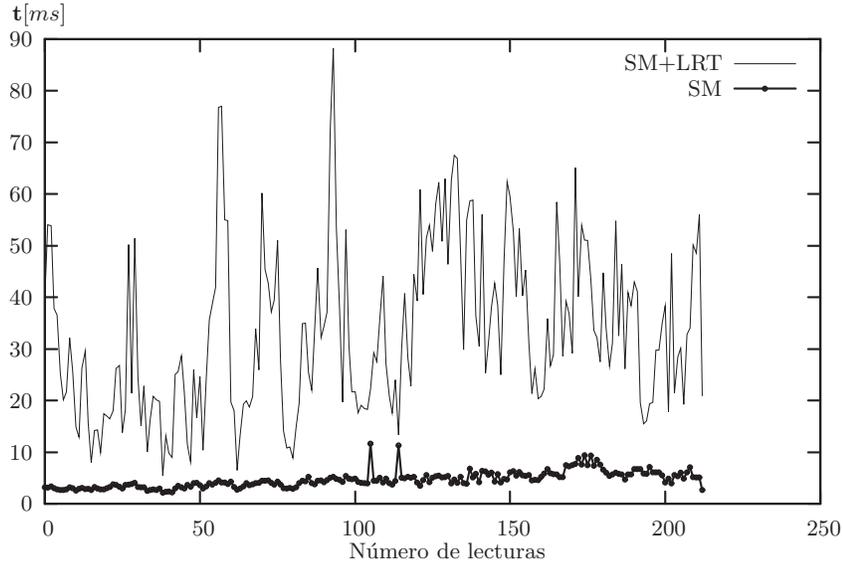


Figura 3.7: Comparación del tiempo total (extracción de características + SLAM estocástico). La complejidad del algoritmo LRT sólo depende del número de agrupamientos locales y por ende esta complejidad está acotada.

Las alternativas no probabilistas usualmente se desempeñan bien en entornos de pequeña o mediana escala. Entre las alternativas probabilistas del SLAM estocástico destacan el filtro de partículas y el filtro de información. El IF (del inglés *Information filter*) es el dual del KF que tiene como parámetros la matriz de información (inversa de la matriz de covarianza) y el vector de información. En el IF cada iteración es mucho menos compleja computacionalmente que en el KF; aunque las ecuaciones de predicción son más complejas, éstas son independientes de las observaciones y son fáciles de desacoplar y centralizar [Thrun05]; de tal manera que se simplifica el cálculo.

Los filtros de partículas aproximan la distribución *a posteriori* mediante un conjunto discreto de muestras. La mejor técnica de filtro de partículas es conocida como FastSLAM [Montemerlo05], donde las partículas representan la posición del robot. Montemerlo muestra que las marcas se pueden recuperar fácilmente si se conoce la trayectoria (las marcas son condicionalmente independientes de la trayectoria). El principal reto de esta técnica consiste en aproximar de forma adecuada la probabilidad *a posteriori* cuando hay mucha incertidumbre; ya que para obtener una buena aproximación se requiere un número muy

grande de muestras. Un número elevado de muestras implica mayor complejidad, y reducirlas puede afectar la capacidad de representación. La única ventaja relevante de los filtros de partículas es que la asociación de marcas se simplifica: si cada partícula representa una ruta y la posición de las marcas es independiente dada la trayectoria, entonces la asociación entre marcas se puede realizar con la regla del vecino más cercano; esto implica que cada partícula puede representar diferentes decisiones de asociación.



## Capítulo 4

# Asociación entre marcas

La implementación del SLAM estocástico que se estudia en el capítulo 3 usa marcas geométricas indistinguibles; es decir, las marcas obtenidas en el proceso de medición no se pueden identificar de forma única ya que una marca encontrada puede corresponder a cualquiera de las otras marcas previamente observadas. Al proceso de etiquetar correctamente una medición se le conoce como *asociación de datos* y consiste en relacionar las mediciones con las marcas almacenadas en el mapa.

**Asociación.** Sea  $\mathbf{Z} = \{\mathbf{z}_1, \dots, \mathbf{z}_\eta\}$ , un conjunto de mediciones; y  $\mathbf{m} = \{\mathbf{l}_1, \dots, \mathbf{l}_N\}$ , el mapa total del entorno. Una *asociación* es una función inyectiva  $f_a : \mathbb{N} \mapsto \mathbb{N}$  y se puede representar por el conjunto de pares ordenados  $A = \{(k, f_a(k)) \mid 1 \leq k \leq \eta\}$ , donde el par  $(k, f_a(k))$  significa que la medición  $\mathbf{z}_k$  se asocia con la marca  $\mathbf{l}_{f_a(k)}$ .  $\square$

**Ejemplo 4.1** Sea  $\mathbf{m}' = \{\mathbf{l}_1, \mathbf{l}_2\}$ , un mapa parcial;  $\mathbf{m} = \{\mathbf{l}_1, \mathbf{l}_2, \mathbf{l}_3\}$ , el mapa total del entorno, tal que  $\mathbf{m}' \subset \mathbf{m}$ ; y  $\mathbf{Z} = \{\mathbf{z}_1, \mathbf{z}_2, \mathbf{z}_3\}$  un conjunto de mediciones. La asociación  $A = \{(1, 3), (2, 1), (3, 2)\}$  indica que la marca  $\mathbf{z}_1$  es nueva; es decir,  $\mathbf{l}_3 \notin \mathbf{m}'$ , las marcas  $\mathbf{z}_2$  y  $\mathbf{z}_3$  corresponden a  $\mathbf{l}_1$  y  $\mathbf{l}_2$ , respectivamente.  $\square$

En general, existen dos enfoques de asociación [C. Cadena09]:

- a) **Asociación basada en relaciones espaciales.** Es útil cuando las marcas son indistinguibles entre sí, por lo que se aprovecha la relación espacial entre las diferentes marcas,
- b) **Asociación con información de apariencia.** Se usa cuando las marcas tienen un

descriptor que facilita la asociación.

Para mediciones obtenidas con un telémetro láser existen algunos tipos de marcas que incluyen información de apariencia; por ejemplo, Tipaldi y Arras [Tipaldi10] han propuesto las marcas llamadas FLIRT (del inglés *Fast Laser Interest Region Transform*), sin embargo, es más común el uso de marcas geométricas. Las marcas geométricas son indistinguibles y para resolver el problema de localización se debe usar asociación basada en relaciones espaciales. En el caso de estudio se usan líneas como marcas, por lo que estamos interesados en la asociación basada en relaciones espaciales.

## 4.1. Asociación basada en relaciones espaciales

Los criterios comunes de asociación basada en relaciones espaciales son: vecino más cercano compatible, compatibilidad conjunta, y métodos de selección aleatoria. En el primer caso, el criterio de asociación se basa en la comparación directa entre dos marcas. En los casos de las pruebas de compatibilidad conjunta y los métodos de selección aleatoria se buscan relaciones de un conjunto de marcas con un subconjunto de las marcas que se encuentran almacenadas en el mapa. A continuación se describe cada una de estas técnicas.

### 4.1.1. Vecino más cercano compatible

La técnica más simple de asociación selecciona al vecino más cercano de acuerdo a la distancia conocida como NIS<sup>1</sup> (del inglés *Normalized Innovation Squared*). Dada la innovación  $\mathbf{v}$  con covarianza  $\mathbf{C}_v$ . La distancia NIS se define como

$$M = \mathbf{v}^T \mathbf{C}_v^{-1} \mathbf{v} \quad (4.1)$$

Si  $\mathbf{v}$  sigue una distribución de probabilidad gaussiana, la distancia  $M$  sigue una distribución  $\chi^2$ . La distribución  $\chi^2$  depende de la dimensión de  $\mathbf{v}$ ; de aquí en adelante se denotará como  $\chi_\kappa^2$  a la distribución con  $\kappa$  grados de libertad.

---

<sup>1</sup>también conocida como distancia de Mahalanobis

$\kappa$	$\gamma_\kappa$
1	3.841
2	5.991
3	7.815
4	9.488
5	11.07
6	12.59

$\kappa$	$\gamma_\kappa$
7	14.07
8	15.51
9	16.92
10	18.31
11	19.68
12	21.03

Tabla 4.1: Tabla inversa de la distribución  $\chi^2$  para  $p = 0.95$ 

**Criterio de Compatibilidad.** Sea  $p$  la proporción esperada de marcas correctamente aceptadas, y  $\mathbf{l}_i$  y  $\mathbf{l}_j$  dos marcas. Se dice que  $\mathbf{l}_i$  y  $\mathbf{l}_j$  son *compatibles* si

$$M = \mathbf{v}^T \mathbf{C}^{-1} \mathbf{v} < \gamma_\kappa; \quad (4.2)$$

donde  $\mathbf{v} = \mathbf{l}_i - \mathbf{l}_j$  es la innovación entre las marcas,  $\mathbf{C}$  es la covarianza de la innovación, y  $\gamma_\kappa$  es un valor tal que  $P(\chi_\kappa^2 < \gamma_\kappa) = p$ .  $\square$

Un valor típico de  $p$  es 0.95, la Tabla 4.1 muestra los valores de  $\gamma_\kappa$  para diferentes grados de libertad  $\kappa$ .

**Ejemplo 4.2 Distancia entre dos líneas.** Calcular la distancia NIS entre una línea  $\mathbf{z} = [r_z, \alpha_z]^T$  con referencia local y con covarianza  $\mathbf{R}$ , y la  $j$ -ésima línea del mapa  $\mathbf{l}_j = [r_j, \alpha_j]^T$ .

Solución. Para transformar la línea  $\mathbf{l}_j$  al sistema de referencia local se usa (3.28):

$$\mathbf{l}'_j = \mathbf{h}(\hat{\tau}) = \mathbf{h}(\hat{\mathbf{x}}_i, \hat{\mathbf{l}}_j) = \begin{bmatrix} r_j - y \sin \alpha_j - x \cos \alpha_j \\ \alpha_j - \theta \end{bmatrix}$$

con lo que se puede calcular la innovación y su covarianza

$$\mathbf{v} = \mathbf{z} - \mathbf{l}'_j = \mathbf{z} - \mathbf{h}(\hat{\tau})$$

$$\mathbf{C}_v = \mathbf{R} + \mathbf{H}_\tau \mathbf{P} \mathbf{H}_\tau^T$$

donde  $\mathbf{P}$  es la covarianza del mapa y  $\mathbf{H}_\tau$  está dado por (3.32).  $\square$

El criterio del vecino más cercano no es el más adecuado porque no considera la relación entre los diferentes pares de marcas. Un ejemplo típico se ilustra en la Fig. 4.1, la técnica del vecino mas cercano compatible puede ocasionar la asociación errónea  $\mathbf{l}_2 \mapsto \mathbf{z}_2$ . En el ejemplo,  $\mathbf{z}_2$  es una medición falsa (la marca correspondiente no existe en el entorno).

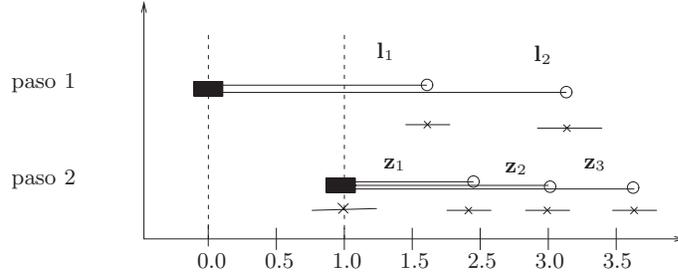


Figura 4.1: Robot en un entorno unidimensional con dos características [Neira01]. Paso 1: creación de un mapa con dos características obtenidas desde la posición inicial. Paso 2: Problema de asociación después de un recorrido de 1m del robot. En este ejemplo se considera que la medición  $\mathbf{z}_2$  no existe en el entorno. Las flechas debajo de las marcas ilustran su covarianza.

#### 4.1.2. Compatibilidad conjunta

Neira y Tardós [Neira01] sugieren que al aumentar el número de pares que cumplan *el criterio de compatibilidad conjunta* se reduce el riesgo de asociar marcas con mediciones falsas.

**Criterio de Compatibilidad Conjunta.** Sea  $\mathbf{m} = \{l_1, l_2, \dots, l_N\}$  el conjunto de las marcas del mapa final; y  $\mathbf{Z} = \{z_1, z_2, \dots, z_\eta\}$  un conjunto de mediciones obtenidas desde la posición fija  $\hat{\mathbf{x}}$ , tal que  $\eta \leq N$ . Se desea evaluar la compatibilidad conjunta de acuerdo a la asociación  $A = \{(k, f_a(k)) \mid k = \{1, \dots, \eta\}\}$ . El *vector de innovación en conjunto* es

$$\mathbf{v} = \begin{bmatrix} \mathbf{z}_1 - \mathbf{h}(\hat{\mathbf{x}}, l_{f_a(1)}) \\ \vdots \\ \mathbf{z}_\eta - \mathbf{h}(\hat{\mathbf{x}}, l_{f_a(\eta)}) \end{bmatrix}, \quad (4.3)$$

donde  $\mathbf{h}$  es el modelo de medición. La covarianza de innovación conjunta es

$$\mathbf{S}_v = \mathbf{H}_x \mathbf{C} \mathbf{H}_x^\top + \mathbf{R} \quad (4.4)$$

donde  $\mathbf{H}_x = \left. \frac{\partial \mathbf{h}}{\partial \mathbf{x}} \right|_{\hat{\mathbf{x}}}$  con  $\underline{\mathbf{h}} = [\mathbf{h}(\hat{\mathbf{x}}, l_{f_a(1)}), \dots, \mathbf{h}(\hat{\mathbf{x}}, l_{f_a(\eta)})]^\top$ ,  $\mathbf{C}$  es la covarianza de las marcas del mapa, y  $\mathbf{R}$  es la covarianza conjunta. Se dice que la asociación  $A$  cumple el criterio de compatibilidad conjunta para  $\mathbf{m}$  y  $\mathbf{Z}_i$ ; si

$$\mathbf{v}^\top \mathbf{S}_v^{-1} \mathbf{v} < \gamma_{\kappa}. \quad (4.5)$$

donde  $\kappa$  es el número de parámetros en total.  $\square$

Note que en la definición anterior  $\mathbf{m}$  se refiere al mapa final del entorno y no al mapa actual; en otras palabras, en las mediciones actuales  $\mathbf{Z}$  pueden existir marcas que no se habían observado previamente pero sí se incluirán en el mapa final.

**Ejemplo 4.3** *Compatibilidad en conjunto para un mapa de líneas.* Suponga que en el tiempo  $i$  un mapa de líneas tiene dos marcas, es decir  $\mathbf{m}_i = [r_1, \alpha_1, r_2, \alpha_2]^\top$  y su covarianza es  $\mathbf{C}_{[\mathbf{m}]_i}$ . Aplique el criterio de compatibilidad conjunta para la asociación  $A = \{(1, 1), (2, 2)\}$ ; si desde la posición actual se obtienen las mediciones  $\mathbf{Z} = \{\mathbf{z}_1, \mathbf{z}_2\}$  con covarianzas  $\mathbf{R}_1$  y  $\mathbf{R}_2$ , respectivamente.

Solución. Dado que se quiere asociar dos marcas cada una representada por dos variables, entonces  $\kappa = 4$ . Si se considera que las mediciones son independientes entre sí, entonces

$$\mathbf{R} = \begin{bmatrix} \mathbf{R}_1 & \mathbf{0} \\ \mathbf{0} & \mathbf{R}_2 \end{bmatrix},$$

$$\underline{\mathbf{h}} = \begin{bmatrix} r_1 - y \sin \alpha_1 - x \cos \alpha_1 \\ \alpha_1 - \theta \\ r_2 - y \sin \alpha_2 - x \cos \alpha_2 \\ \alpha_2 - \theta \end{bmatrix},$$

y la covarianza de innovación conjunta se calcula con (4.4). En la Tabla 4.1 se encuentra que el valor de  $\gamma_4 = 9.488$ . Si se cumple (4.5) entonces los conjuntos son compatibles.  $\square$

### Algoritmo JCBB

La técnica de *ramificación y poda* (del inglés *Branch and Bound*) crea particiones de forma recursiva a partir del conjunto de posibles soluciones; cada partición se puede representar como un nodo de un árbol de búsqueda. Al proceso de crear una nueva partición a partir de un subconjunto de soluciones se le conoce como *ramificación*. El proceso de *poda* evita continuar la exploración de ciertos nodos ya sea porque no son factibles (no cumplen algún criterio) o porque se prevee que las soluciones que se obtengan de dicho nodo serán de menor calidad que la mejor solución disponible en el momento. Este procedimiento

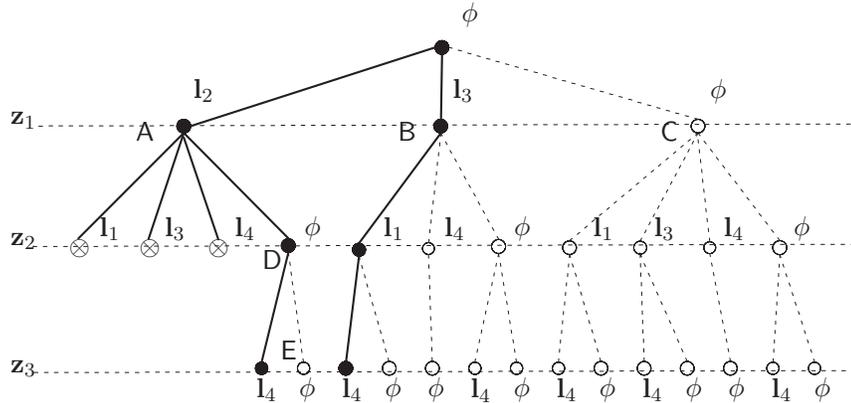


Figura 4.2: Árbol de búsqueda para el conjunto de pares compatibles:  $\{(z_1, l_2), (z_1, l_3), (z_2, l_1), (z_2, l_3), (z_2, l_4), (z_3, l_4)\}$ . Cada nivel del árbol define un conjunto de asociaciones y la ramificación representa la inclusión de una nueva asociación. Los nodos marcados con una cruz representan asociaciones no compatibles, los negros son asociaciones compatibles y los vacíos son aquellos que no hubo necesidad de explorar.

es muy común en muchos problemas de búsqueda y optimización combinatoria (*v.g.* en [Lara09] proponemos una forma eficiente de encontrar la intersección de un hiperplano y un hiperplano).

En esta sección estamos interesados en encontrar el mejor conjunto de asociaciones. El algoritmo JCBB (del inglés *Joint Compatibility Branch and Bound*) [Neira01] genera conjuntos tentativos de asociaciones y busca el conjunto con la mayor cantidad de marcas que cumpla con el criterio de compatibilidad conjunta. Si dos conjuntos tienen la misma cantidad de pares compatibles escoge aquel de máxima verosimilitud<sup>2</sup>. El algoritmo JCBB realiza la búsqueda con la técnica de ramificación y poda; en este caso cada nivel del árbol define un conjunto de posibles asociaciones y la ramificación representa la inclusión de una nueva asociación de tal manera que el conjunto de asociaciones forme una función inyectiva. Si al incluir una nueva asociación se cumple el criterio de compatibilidad conjunta entonces se continúa la ramificación. La poda consiste en considerar la cantidad máxima posible de asociaciones pendientes de un nodo; si dicha cantidad es inferior a la mejor asociación actual entonces se detiene la búsqueda.

La Fig. 4.2 ilustra con un ejemplo el algoritmo JCBB. En el primer nivel se con-

<sup>2</sup>El algoritmo JCBB propuesto en [Neira01] prefiere el conjunto con menor distancia NIS.

sideran las marcas compatibles con  $\mathbf{z}_1$ ; es decir,  $(\mathbf{z}_1, \mathbf{l}_2)$ ,  $(\mathbf{z}_1, \mathbf{l}_3)$  y  $(\mathbf{z}_1, \phi)$ , donde la última asociación  $(\mathbf{z}_1, \phi)$  significa que  $\mathbf{z}_1$  se incluiría como marca nueva. Los nodos etiquetados con las letras A, B y C se explorarán en profundidad. Al ramificar el nodo A con las asociaciones compatibles de  $\mathbf{z}_2$  se obtienen conjuntos de asociaciones que no cumplen el criterio de compatibilidad conjunta usando  $(\mathbf{z}_2, \mathbf{l}_1)$ ,  $(\mathbf{z}_2, \mathbf{l}_3)$  y  $(\mathbf{z}_2, \mathbf{l}_4)$ . También se observa que al explorar el nodo marcado como D se obtiene el conjunto  $\{(\mathbf{z}_1, \mathbf{l}_2), (\mathbf{z}_3, \mathbf{l}_4)\}$  que es compatible y tiene dos asociaciones; por tal situación se poda el nodo E pues no hay forma de obtener conjuntos con al menos dos asociaciones. Al explorar en profundidad el nodo marcado con la letra B se obtiene la asociación  $\{(\mathbf{z}_1, \mathbf{l}_3), (\mathbf{z}_2, \mathbf{l}_1), (\mathbf{z}_3, \mathbf{l}_4)\}$  con cardinalidad de tres; por tal motivo se podan los nodos restantes porque no es posible encontrar conjuntos con al menos tres asociaciones y con esto finaliza el proceso.

### 4.1.3. Asociación basada en RANSAC

La idea de los algoritmos de consenso de muestra aleatoria (Cap. 2 Sec. 2.2) se puede aplicar directamente al problema de registro. Como se describe en el Algoritmo 4.1 en cada iteración la técnica de asociación basada en RANSAC: genera hipótesis aleatorias mínimas (líneas 4 y 5), calcula una posible transformación (línea 6) y luego compara con la totalidad de los datos disponibles con alguna función estadística  $f_r(\cdot)$ . Después de iterar un cierto número de veces, se encuentra la mejor asociación (línea 10). El número de iteraciones (línea 1) se puede calcular con la proporción esperada de asociaciones erróneas [Hartley00].

En [Lara07] proponemos usar *esquinas virtuales* para resolver el problema de registro de dos secuencias de mediciones de rango consecutivas (*Scan Matching*). Este algoritmo funciona bien para resolver el problema de registro (encontrar la transformación entre mediciones obtenidas en dos posiciones consecutivas), pero es costoso computacionalmente cuando se usa para resolver el problema de asociación entre un conjunto de líneas locales con el mapa; esto se debe a que la cantidad de asociaciones posibles es mayor y además se incrementa la probabilidad de elegir una asociación errónea y por tanto se aumenta el número de iteraciones.

**Algoritmo 4.1:** Algoritmo de asociación basado en RANSAC

**Entrada:** El conjunto de marcas sensadas de la posición actual  $\mathbf{Z}$  y los mejores estimados del mapa de líneas  $\mathbf{m}$ .

**Salida:** La mejor asociación  $A$

- 1 Calcular el número de iteraciones  $n_{it}$
- 2  $i \leftarrow 0$
- 3 **repetir**
- 4     Seleccionar un conjunto mínimo de mediciones,  $\mathbf{Z}_h \subseteq \mathbf{Z}$
- 5     Seleccionar un conjunto mínimo de marcas del mapa,  $\mathbf{m}_h \subseteq \mathbf{m}$  de tal manera que las marcas  $\mathbf{m}_h$  sean compatibles uno a uno con las mediciones  $\mathbf{Z}_h$
- 6     Con  $\mathbf{Z}_h$  y  $\mathbf{m}_h$ , estimar la transformación rígida  $T$
- 7     **si**  $(f_r(T) > f_r(T^*))$  ó  $(T^*$  no inicializado) **entonces**  $T^* \leftarrow T$
- 8      $i \leftarrow i + 1$
- 9 **mientras**  $i \leq n_{it}$
- 10 Calcular la mejor asociación  $A = \{(k, f_a(k)) \mid k \in \{1, \dots, \eta\}\}$
- 11 **regresar**  $A$

#### 4.1.4. Dificultades de los esquemas de asociación

Los algoritmos JCBB y RANSAC tratan de maximizar el número de marcas asociadas; este criterio es suficiente cuando en una sólo medición existen muchas marcas, pero es limitado cuando el número de marcas por medición es reducido.

La Fig. 4.3 muestra una dificultad típica de este criterio de asociación para un mapa de líneas (en entornos interiores usualmente se detectan pocas líneas a la vez). En el mapa ideal de la Fig. 4.3a se pueden observar tres líneas, dos de ellas son puertas y la más larga y gruesa es una pared. Desde la posición  $\mathbf{x}_A$  se inicializa el mapa, aquí el robot sólo ha podido percibir la puerta más próxima y la pared (la puerta más alejada a él no se ha detectado). Desde la posición  $\mathbf{x}_B$  el robot toma mediciones locales con lo que se detectan las líneas  $\mathbf{Z} = \{\mathbf{z}_1, \mathbf{z}_2, \mathbf{z}_3\}$ . La asociación  $A = \{(1, 1), (2, 2), (3, 3)\}$  maximiza la cantidad de

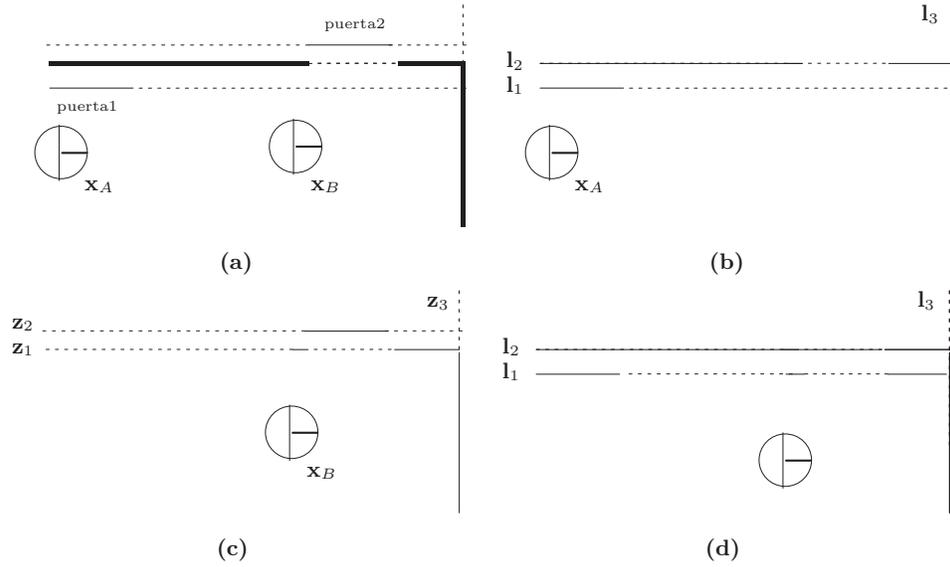


Figura 4.3: Dificultades del algoritmo JCBB en mapas de líneas. (a) mapa ideal (b) Mapa  $\mathbf{m} = \{l_1, l_2, l_3\}$  obtenido hasta la posición  $x_A$  (c) El conjunto de mediciones  $\mathbf{Z} = \{z_1, z_2, z_3\}$  obtenidas desde  $x_B$  (d) Mapa que se obtiene después de JCBB, este mapa es diferente al esperado en el mapa ideal mostrado en (a).

marcas compatibles que se asocian; pero el mapa de la Fig. 4.3d es diferente en el número de marcas que el mapa ideal de la Fig. 4.3a. El problema central es la función objetivo que usa: no siempre asociar la mayor cantidad de marcas compatibles es la mejor opción.

Para reducir este tipo de situaciones la siguiente sección propone un esquema de asociación basado en los segmentos de cada línea.

## 4.2. Asociación geométrica propuesta

La idea principal del esquema de asociación que se propone en este capítulo se ilustra en la Fig. 4.4. Si además de las líneas al infinito, el mapa contiene información de segmentos<sup>3</sup> dirigidos entonces es posible obtener una *secuencia de predicción*; esta secuencia indica que líneas se esperan leer desde una posición determinada; para la Fig. 4.4(a), la secuencia de predicción es

$$\Lambda_{\mathbf{m}} = \langle 22\varepsilon\varepsilon\varepsilon\varepsilon 233333 \rangle$$

<sup>3</sup>Note que en esta representación, una línea puede tener asociado más de un segmento.

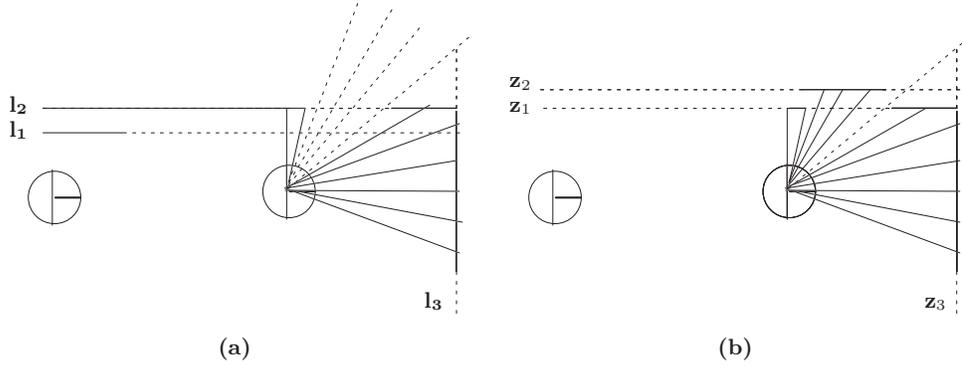


Figura 4.4: Generación de secuencias de comparación: (a) **Secuencia de predicción.** A partir de los segmentos del mapa y la posición del robot se predice la secuencia  $\langle 22\varepsilon\varepsilon\varepsilon\varepsilon 233333 \rangle$  (b) **Secuencia de líneas observadas.** La secuencia  $\langle 11222\varepsilon 333333 \rangle$  se obtiene a partir de la secuencia de mediciones de rango  $S$ .

donde  $\varepsilon$  significa que el rayo de medición no intersecta con ningún segmento.

Por otro lado, de una lectura de rango  $S$  se puede obtener la *secuencia de líneas observadas*; esta secuencia indica a que línea pertenece cada medición. Para el ejemplo de la Fig. 4.4(b), la secuencia de medición es

$$\Lambda_{\mathbf{z}} = \langle 11222\varepsilon 133333 \rangle$$

**Sustitución de una secuencia.** Sea  $\Lambda_{\mathbf{z}} = \langle c_1, \dots, c_j, \dots, c_J \rangle$  una secuencia de líneas observadas y  $A = \{(k, f_a(k)) \mid k \in \{1, \dots, \eta\}\}$  una asociación. La sustitución de  $\Lambda_{\mathbf{z}}$  usando  $A$ ,  $\mathbf{sust}(\Lambda_{\mathbf{z}}, A)$ , es una secuencia  $\langle b_1, \dots, b_j, \dots, b_J \rangle$  donde

$$b_j = \begin{cases} f_a(c_j) & \text{cuando } (c_j \neq \varepsilon), \text{ y} \\ \varepsilon & \text{en caso contrario.} \end{cases}$$

□

**Ejemplo 4.4** *Sustitución de una secuencia con diferentes asociaciones.*

En el ejemplo de la Fig. 4.4.  $\Lambda_{\mathbf{z}} = \langle 11222\varepsilon 133333 \rangle$ . Si  $A_1 = \{(1, 1), (2, 2), (3, 3)\}$  y  $A_2 = \{(1, 2)(2, 4)(3, 3)\}$ , entonces  $\mathbf{sust}(\Lambda_{\mathbf{z}}, A_1) = \langle 11222\varepsilon 133333 \rangle$  y  $\mathbf{sust}(\Lambda_{\mathbf{z}}, A_2) = \langle 22444\varepsilon 233333 \rangle$

□

En el ejemplo, los resultados de la sustitución se pueden contrastar con la secuencia de predicción:

$$\mathbf{sust}(\Lambda_{\mathbf{z}}, A_1) = \langle 11222\varepsilon 133333 \rangle$$

$$\mathbf{sust}(\Lambda_{\mathbf{z}}, A_2) = \langle 22444\varepsilon 233333 \rangle$$

$$\Lambda_{\mathbf{m}} = \langle 22\varepsilon\varepsilon\varepsilon\varepsilon 233333 \rangle$$

En este caso, la cadena de medición  $\Lambda_{\mathbf{m}}$  es más parecida a  $\mathbf{sust}(\Lambda_{\mathbf{z}}, A_2)$  que a  $\mathbf{sust}(\Lambda_{\mathbf{z}}, A_1)$ . Como las secuencias tienen el mismo tamaño, se puede calcular la distancia de Hamming (el número de símbolos que se tienen que cambiar para transformar una secuencia en otra). Así,  $d_{\text{ham}}(\Lambda_{\mathbf{m}}, \mathbf{sust}(\Lambda_{\mathbf{z}}, A_1)) = 3$  y que  $d_{\text{ham}}(\Lambda_{\mathbf{m}}, \mathbf{sust}(\Lambda_{\mathbf{z}}, A_2)) = 6$ ; de acuerdo a este criterio, se prefiere la asociación  $A_2$ . Para reducir el efecto del error en la posición se puede usar otra medida de similitud entre secuencias; por ejemplo, la T-WLCS (del inglés *Time-Warped Longest Common Subsequence*)[Guo04] es una medida de similitud que considera expansión y contracción de secuencias sin imponer restricciones de donde inicia y concluye la comparación.

**Similitud entre dos cadenas con T-WLCS [Guo04].** Sean  $\Lambda_{\mathbf{a}} = \{a_1, \dots, a_I\}$  y  $\Lambda_{\mathbf{b}} = \{b_1, \dots, b_J\}$  dos secuencias, la matriz auxiliar  $W = [w_{i,j}]_{(I+1) \times (J+1)}$  se define como

$$w_{i,j} = \begin{cases} 0 & (i=0) \vee (j=0), \\ w_{i-1,j-1} + 1 & (i>0) \& (j>0) \& (a_i = b_j), \\ \text{máx} \{w_{i-1,j}, w_{i,j-1}\} & (i>0) \& (j>0) \& (a_i \neq b_j); \end{cases}$$

y la similitud entre  $\Lambda_{\mathbf{a}}$  y  $\Lambda_{\mathbf{b}}$ , es

$$\mathbf{sim}_{\text{TWL}}(\Lambda_{\mathbf{a}}, \Lambda_{\mathbf{b}}) = w_{I,J} \quad (4.6)$$

□

**Ejemplo 4.5 Cálculo de similitud T-WLCS [Guo04].** Sean  $\Lambda_{\mathbf{a}} = \langle 4567 \rangle$  y  $\Lambda_{\mathbf{b}} = \langle 4151617 \rangle$  dos secuencias, calcular  $\mathbf{sim}_{\text{TWL}}(\Lambda_{\mathbf{a}}, \Lambda_{\mathbf{b}})$ .

Solución. La matriz auxiliar  $W$  es,

**Algoritmo 4.2:** Algoritmo de asociación propuesto**Entrada:** El conjunto de líneas sensadas de la posición actual

$\mathbf{Z} = \{\mathbf{z}_1, \dots, \mathbf{z}_\eta\}$  con los agrupamientos de mediciones asociados a cada línea  $D = \{D_1, \dots, D_\eta\}$ , y los mejores estimados del mapa de líneas  $\mathbf{m}$  y de la posición actual  $\hat{\mathbf{x}}$ .

**Salida:** La mejor asociación  $A$ 

- 1 Con  $\mathbf{Z}$  y  $D$ , obtener la secuencia de líneas observadas  $\Lambda_{\mathbf{z}}$
- 2 Con  $\mathbf{m}$  y  $\hat{\mathbf{x}}$ , obtener la secuencia de predicción  $\Lambda_{\mathbf{m}}$
- 3 Encontrar la asociación  $A = \{(k, f_a(k)) \mid k \in \{1, \dots, \eta\}\}$  que cumpla con el criterio de compatibilidad conjunta y que maximice  $\text{sim}_{\text{TWL}}(\Lambda_{\mathbf{m}}, \text{sust}(\Lambda_{\mathbf{z}}, A))$
- 4 **regresar**  $A = \{(k, f_a(k)) \mid k \in \{1, \dots, \eta\}\}$

		$\Lambda_{\mathbf{b}}$							
		4	1	5	1	6	1	7	1
		0	0	0	0	0	0	0	0
$\Lambda_{\mathbf{a}}$	4	0	1	1	1	1	1	1	1
	5	0	1	1	2	2	2	2	2
	6	0	1	1	2	2	3	3	3
	7	0	1	1	2	2	3	3	4

por tanto,  $\text{sim}_{\text{TWL}}(\Lambda_{\mathbf{a}}, \Lambda_{\mathbf{b}}) = 4$ . □

**Algoritmo propuesto**

El Algoritmo 4.2 describe la técnica de asociación propuesta: a partir de las líneas  $\mathbf{Z}$  y sus puntos asociados  $D$  se obtiene la secuencia de observación (línea 1); a partir del mapa  $\mathbf{m}$  y la posición estimada  $\hat{\mathbf{x}}$  se obtiene la secuencia de predicción (línea 2); finalmente, se encuentra la asociación conjunta que minimice la distanciasimilitud entre las secuencias (línea 3).

Tabla 4.2: Parámetros para generar datos sintéticos

núm. líneas	26
dimensiones del entorno	$9.5 \times 7$ [mts]
distancia mínima entre líneas paralelas próximas	70[mm]
movimiento promedio del robot	50[cm] translación, $40^\circ$ rotación
modelo de movimiento	gaussiano con desviaciones estándar: 10[cm] en dirección de translación, 10[cm] en dirección ortogonal a la translación, y $5^\circ$ en rotación.
apertura del láser	$180^\circ$
resolución angular del láser	$0.5^\circ$
modelo de mediciones	0.2 ruido uniforme + 0.8 gaussiano con desviación 1[cm] (direccional)
número de posiciones del robot (secuencias de medición láser)	76

### 4.3. Resultados experimentales

Para probar la técnica de asociación se han usado datos sintéticos y datos reales obtenidos de nuestro laboratorio.

#### 4.3.1. Prueba en entorno sintético

Para generar las mediciones sintéticas a partir del entorno de la Fig. 4.5 se usaron los parámetros que se muestran en la Tabla 4.2; mientras que para resolver el problema del SLAM se usó la configuración que se muestra en la Tabla 4.3.

La Fig. 4.6 muestra el error para cada una de las 76 posiciones; este error se calcula como la distancia euclídeana entre la posición sintética y la posición estimada.

El algoritmo JCBB incurre en errores de asociación, los cuales son más evidentes cuando se asocian líneas paralelas (Fig. 4.6). Estos errores se reducen en el esquema de asociación propuesto. Cuando se usa el algoritmo de asociación JCBB, los errores máximos en los que se incurre tienen origen en la distancia entre líneas paralelas próximas (70 mm).

Tabla 4.3: Configuración para la prueba sintética y para la primera prueba en entorno real

algoritmo de SLAM	SLAM Estocástico con líneas
extractor de marcas	Split & Merge + LRT
restricciones para considerar una línea válida	mínimo de 9 pts y 30 cm de largo

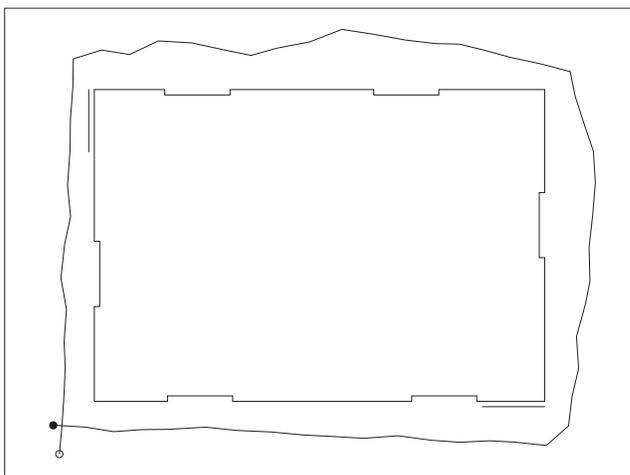


Figura 4.5: Entorno simulado: la ruta se representa por el polígono, la posición inicial con un círculo y la posición final con un círculo relleno.

### 4.3.2. Prueba en entorno real

Para las pruebas en entorno real se usaron dos algoritmos como se describe a continuación.

#### Prueba con SLAM estocástico

Para la primera prueba en el entorno real, se usó el algoritmo de SLAM estocástico con la configuración de la Tabla 4.3. La comparación de resultados se muestra en la Fig. 4.7; en la Fig. 4.7a se nota que la asociación con JCBB ha mezclado líneas que no pertenecen al mismo objeto en el entorno, lo que ocasiona errores tanto en el mapa como en la localización. Una mejor asociación se muestra en la Fig. 4.7b, donde se observa que se han detectado correctamente los segmentos aun cuando las líneas son paralelas entre sí.

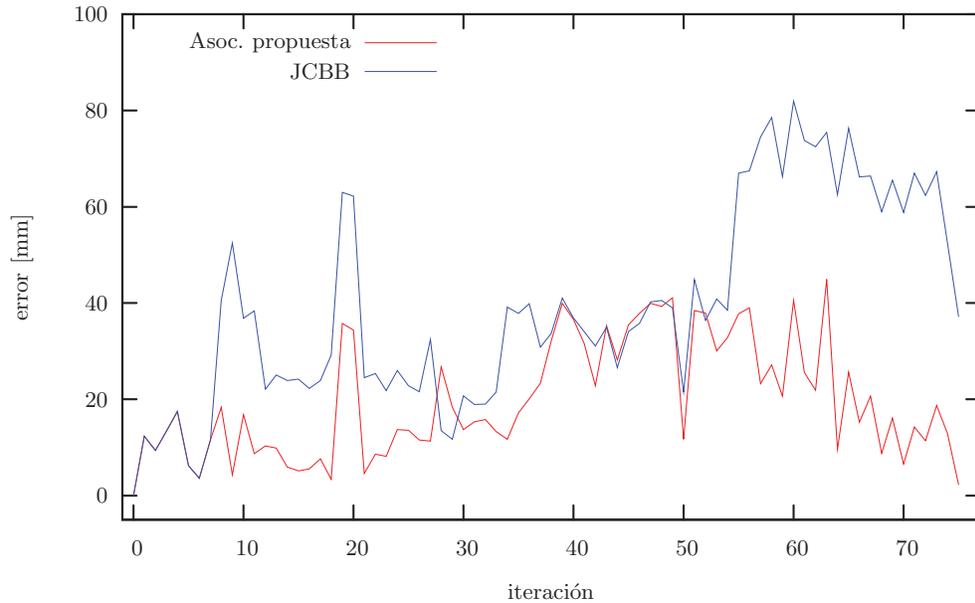


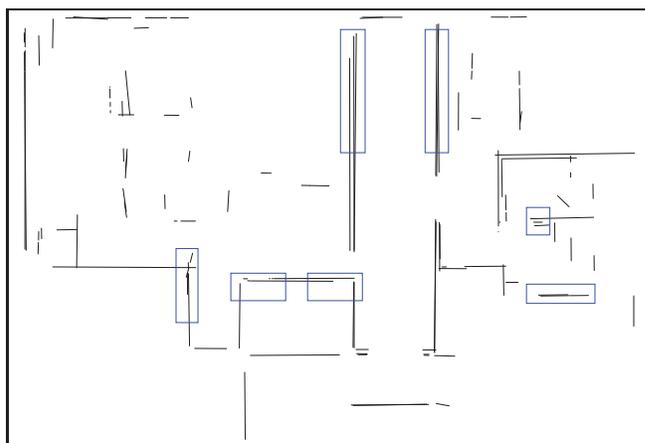
Figura 4.6: Error entre la posición sintética y la estimada; en general, se puede ver que la asociación propuesta (rojo) obtiene mejores resultados que la asociación con JCBB (azul).

Tabla 4.4: Configuración para la segunda prueba en entorno real

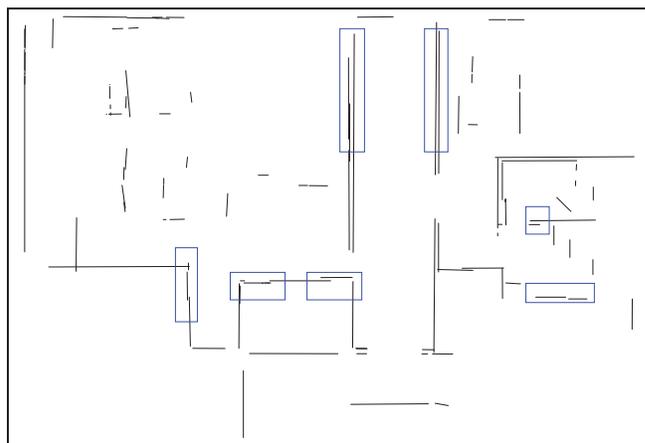
algoritmo de SLAM	Suavizado y generación de mapas [Kaess09]
extractor de marcas	Split & Merge + LRT
restricciones para considerar una línea válida	mínimo de 9 pts y 30 cm de largo

### Prueba con Suavizado y generación de mapas

En esta prueba se usó el algoritmo de *suavizado y generación de mapas* [Kaess09]; este algoritmo resuelve el problema de SLAM total, es decir, además del mapa del entorno también obtiene la ruta del robot. Este algoritmo se configura de acuerdo a los parámetros de la Tabla 4.4. Los mapas obtenidos con esta técnica se muestran en la Fig. 4.8; al igual que la prueba anterior, la técnica de asociación propuesta obtiene las líneas que realmente se encuentran en el entorno (Fig. 4.8b) mientras que la asociación con JCT no (Fig. 4.8b). Note que la ruta del robot es útil para diferentes aplicaciones; por ejemplo, el mapa bidimensional puede servir como base para la reconstrucción tridimensional.



(a)



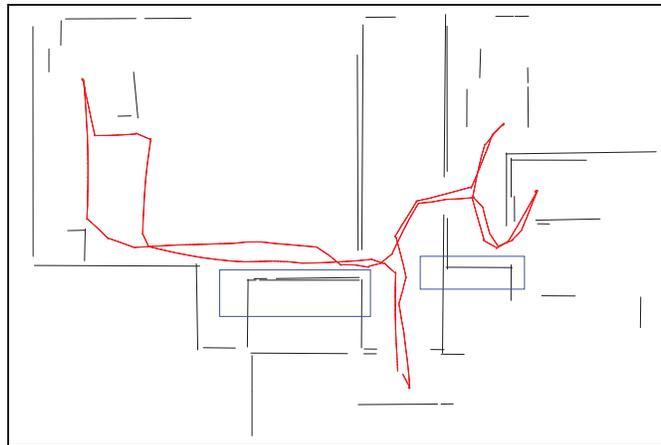
(b)

Figura 4.7: Mapas de líneas del entorno real con SLAM estocástico y con diferentes esquemas de asociación (a) JCBB y (b) algoritmo propuesto. Los rectángulos en azul muestran algunas zonas de interés.

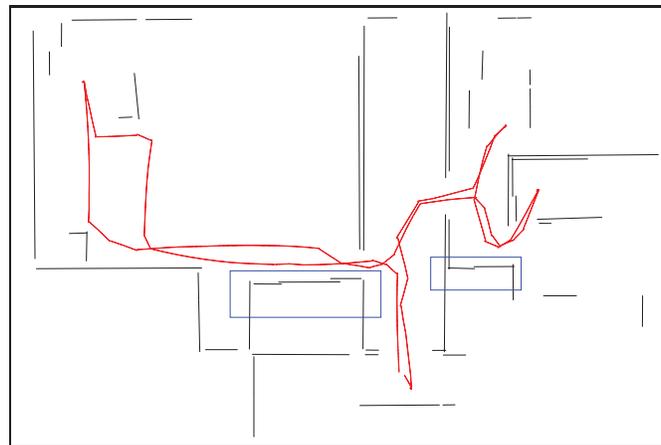
#### 4.4. Discusión

La asociación geométrica propuesta mejora los resultados respecto de los obtenidos con el algoritmo *Joint Compatibility Branch and Bound*. La principal desventaja del esquema de asociación propuesto es que depende de los límites de segmentos. Dado que los límites de los segmentos no se incluyen en la información estocástica del mapa, la estimación no es muy precisa pues depende de la posición del robot en el momento de la detección; esto se

puede notar sobre todo en las esquinas que forman las líneas de la Fig. 4.7b. En entornos interiores el enfoque de asociación basado en segmentos propuesto es adecuado porque la incertidumbre está acotada –y en todo caso se resuelve con el uso de la medida de similitud T-WLCS, sin embargo, esto puede representar dificultades mayores en entornos donde la



(a)



(b)

Figura 4.8: Mapa de nuestro laboratorio con la técnica de suavizado [Kaess09] se contrastan dos técnicas de asociación; el polígono rojo indica la trayectoria del robot y los rectángulos azules muestran líneas paralelas que en el entorno real son puertas y paredes. (a) Criterio de compatibilidad conjunta (JCT) y (b) Criterio de compatibilidad conjunta con el algoritmo propuesto. Como se puede apreciar la técnica propuesta obtiene los segmentos de línea que realmente existen en el entorno

incertidumbre aumente considerablemente –como en pasillos extremadamente largos (más largos que el alcance del sensor) donde se incrementa la incertidumbre en la dirección del movimiento. Una solución consiste en manejar los puntos extremos como información estocástica del mapa; pero esta solución incrementa sustancialmente el costo computacional.

El problema de detección de los límites de los segmentos mejora sustancialmente cuando se resuelve el problema desde la perspectiva del problema de SLAM total: cualquier mejora posterior en la estimación de la ruta impacta en la mejor estimación de los límites del segmento (Fig. 4.8). Aunque una mejor opción consiste en no depender de los límites de los segmentos, esta investigación se realizará en el futuro próximo.

## Capítulo 5

# Conclusiones

### 5.1. Contribuciones

En esta tesis se presentan dos contribuciones teóricas: 1) plantemiento de una razón de verosimilitud para la detección de marcas geométricas y 2) asociación de líneas usando secuencias de etiquetas; a continuación se describen estas contribuciones.

#### 5.1.1. Razón de verosimilitud para detección de marcas geométricas

Los enfoques basados en marcas requieren descubrir un conjunto de marcas a partir de una secuencia de mediciones. El enfoque aglomerativo primero encuentra marcas locales y luego mezcla aquellas que representen al mismo objeto. Esta tesis propone una razón de verosimilitud que guía el proceso de mezclado. La técnica propuesta estudia el caso de líneas; aunque se puede aplicar a otro tipo de marcas geométricas.

#### Contribuciones

- Se propone una razón de verosimilitud para decidir si los agrupamientos que representan a dos líneas se deben mezclar o no. Se contrastan dos hipótesis:  $H_1$  los agrupamientos son mediciones del mismo objeto plano y  $H_2$  existe un cambio significativo entre los agrupamientos. La razón entre las verosimilitudes de  $H_1$  y  $H_2$  que se propone ayuda a tomar la decisión.

- Este enfoque mejora sustancialmente la detección de marcas geométricas en comparación con otros criterios de mezclado, además sólo usa los parámetros del modelo de medición por lo que evita el uso de umbrales.

### 5.1.2. Asociación de líneas usando secuencias de etiquetas

En los enfoques basados en marcas es inevitable la solución al problema de asociación. Cuando se usan marcas indistinguibles, se deben usar algoritmos de asociación basados en relaciones espaciales. Es común que en el mapa se incluyan marcas tan abstractas que se pierda información relevante del entorno; y por ende, las asociaciones resultantes sean erróneas. Una opción consiste en usar mayor diversidad de marcas que reduzcan la pérdida de información; esta no es una solución factible porque eleva la complejidad temporal del algoritmo. La idea en la que se basa esta contribución es el uso de marcas auxiliares locales, cuya definición sea más tolerante a ruido y que permita guiar el proceso de asociación. Debido a la definición local (independiente de la posición del robot y con incertidumbre acotada); no es necesaria la inclusión de esta información con variables estocásticas; evitando así el aumento de la complejidad temporal.

### Contribuciones

- La información de los segmentos de las líneas puede mejorar el proceso de asociación. Con la información del conjunto de segmentos del mapa se puede predecir la secuencia de líneas (etiquetas) que se leerán desde una posición estimada; esta *secuencia de etiquetas de predicción* se puede contrastar con la secuencia obtenida a partir de las mediciones sensoriales. Las mejores asociaciones se obtienen cuando la distancia entre las dos secuencias es menor. Para reducir el impacto de la incertidumbre en la posición del robot se usa una distancia entre secuencias que considera la repetición y ausencia de etiquetas.
- La propuesta es una solución para problemas de asociación frágil, y es de gran utilidad práctica en algoritmos donde es costoso obtener la covarianza exacta de las variables estocásticas –como en el caso del enfoque de suavizado y generación de mapas.

- Esta solución permite manejar adecuadamente las líneas en el mapa; es decir cuando agregar, actualizar y eliminar líneas.

## 5.2. Trabajos futuros

El autor desea continuar la investigación en esta área, algunas de las ideas que se muestran a continuación ya tienen un avance significativo:

### Problema de detección de marcas

- Extensión de la técnica LRT para obtener planos y otros modelos geométricos a partir de mediciones tridimensionales de rango.
- Detección de diferentes tipos de modelos al mismo tiempo (*v.g.* encontrar círculos, líneas y agrupamientos) a partir de una secuencia de mediciones.

### Problema de asociación

- Robustecer el algoritmo de asociación geométrico presentado en el Capítulo 4 a través del uso de Modelos Ocultos de Markov. Este nuevo enfoque es independiente de la posición exacta de los segmentos de las líneas.

### Problema de SLAM

- Se están estudiando técnicas de SLAM activo que permitan obtener mejores mapas seleccionando las acciones inmediatas del robot.
- En la actualidad, se tiene implementado el algoritmo de suavizado y generación de mapas. Se buscará reducir el costo computacional del algoritmo de suavizado y generación a través de dos estrategias: detección de ciclos y asociación de marcas con el uso de covarianzas aproximadas y técnicas de asociación geométrica.
- Se desea incursionar en problemas de mayor escala y en entornos mixtos (interiores y exteriores).



## Apéndice A

# Cálculo de integrales

Un tópico que es importante en muchos cálculos probabilistas es resolver, al menos de forma aproximada, integrales de funciones gaussianas. El objetivo de este apéndice es mostrar resultados prácticos de dichas integrales y su aplicación a los problemas que se analizan en el capítulo 2.

### A.1. Integral de funciones gaussianas

Como lo demuestra Sivia en [Sivia96], la integral de una función gaussiana de una sola variable es,

$$\int_a^b \exp\left[-\frac{x^2}{2\sigma^2}\right] dx = \sigma\sqrt{\frac{\pi}{2}} \left[ \operatorname{erf}\left(\frac{b}{\sigma\sqrt{2}}\right) - \operatorname{erf}\left(\frac{a}{\sigma\sqrt{2}}\right) \right]. \quad (\text{A.1})$$

donde  $\operatorname{erf}(x) = \frac{2}{\pi} \int_x^\infty e^{-t^2} dt$ ; y la integral definida de una función gaussiana multidimensional es:

$$\int \cdots \int \exp\left[-\frac{1}{2}\mathbf{x}^T \mathbf{H} \mathbf{x}\right] dx_1 dx_2 \cdots dx_N = \frac{(2\pi)^{N/2}}{\sqrt{\det(\mathbf{H})}}, \quad (\text{A.2})$$

donde  $\mathbf{x} = [x_1 \ x_2 \ \dots \ x_N]^T$ .

## A.2. Cálculo de la evidencia

En esta sección estamos interesados en resolver las integrales de (2.6), mismas que tienen la forma

$$W = \int_0^{2\pi} \int_0^{r_{\max}} e^{-\frac{\chi_j^2}{2}} dr_j d\alpha_j \quad (\text{A.3})$$

donde

$$\chi_j^2 = \sum_{\mathbf{p}_i \in Z_j} \frac{d_{\perp}^2(\mathbf{p}_j, \mathbf{l}_j)}{\sigma^2}. \quad (\text{A.4})$$

El valor de  $\chi_j^2$  se pueden aproximar usando una expansión de series de Taylor. Sea  $\mathbf{l}_{j*} = [r_{j*}, \alpha_{j*}]^T$  los parámetros de la mejor línea para el agrupamiento  $Z_j$ . Definamos el error normalizado para la mejor línea como

$$\chi_{j*}^2 = \sum_{\mathbf{p}_i \in D_j} \frac{d_{\perp}^2(\mathbf{p}_j, \mathbf{l}_{j*})}{\sigma^2}. \quad (\text{A.5})$$

La expansión de Taylor alrededor de  $\chi_{j*}^2$  es

$$\chi_j^2 = \chi_{j*}^2 + \frac{1}{2}(\mathbf{1} - \mathbf{l}_{j*})^T \nabla \nabla \chi_j^2 (\mathbf{1} - \mathbf{l}_{j*}) + \dots, \quad (\text{A.6})$$

donde el segundo término se ha omitido porque  $\chi_{j*}^2$  es un mínimo. Sustituyendo los dos primeros términos de A.6 en A.3

$$\begin{aligned} W &= \int_0^{2\pi} \int_0^{r_{\max}} \exp \left[ -\frac{\chi_{j*}^2 + \frac{1}{2}(\mathbf{1} - \mathbf{l}_{j*})^T \nabla \nabla \chi_j^2 (\mathbf{1} - \mathbf{l}_{j*})}{2} \right] dr_j d\alpha_j \\ &= \exp \left[ -\frac{\chi_{j*}^2}{2} \right] \int_0^{2\pi} \int_0^{r_{\max}} \exp \left[ -\frac{1}{4}(\mathbf{1} - \mathbf{l}_{j*})^T \nabla \nabla \chi_j^2 (\mathbf{1} - \mathbf{l}_{j*}) \right] dr_j d\alpha_j \end{aligned} \quad (\text{A.7})$$

Asumiendo que la gran parte de la área de la función de distribución  $\exp(-\chi^2/2)$  está en la región  $0 < r_j < r_{\max}$  y  $0 < \alpha_j < 2\pi$  entonces se puede usar (A.2). De tal forma que (A.7) se puede resolver como

$$W = \int_0^{2\pi} \int_0^{r_{\max}} e^{-\frac{\chi_j^2}{2}} dr_j d\alpha_j = \frac{4\pi}{\sqrt{\det(\nabla \nabla \chi_j^2)}} \exp \left[ -\frac{\chi_{j*}^2}{2} \right], \quad (\text{A.8})$$

donde  $\det(\nabla \nabla \chi_j^2)$  es el determinante del Hessiano evaluado en  $\mathbf{l}_{j*}$ .

## Apéndice B

# Estimación y Filtrado

### B.1. Cálculo de covarianzas

La covarianza es una medida de cuanto dos variables cambian entre sí. Las covarianzas son importantes en los métodos de filtrado expuestos en este documento. En esta sección se estudian los métodos para el cálculo de covarianzas de forma determinista. El modelo clásico de predicción de errores se basa en la propagación de la covarianza de una variable aleatoria. Si la variable aleatoria  $X$  se transforma en la variable aleatoria  $Y$  usando la transformación lineal  $Y = FX$ , entonces

$$\hat{y} = F\hat{x}. \tag{B.1}$$

La covarianza se define como

$$\mathbf{C}_X = E[xx^\top] \tag{B.2}$$

donde  $E[\cdot]$  es el símbolo de esperanza matemática. Así la covarianza  $\mathbf{C}_Y$  se obtiene como

$$\begin{aligned} \mathbf{C}_Y &= E[yy^\top] = E[Fx(Fx)^\top] = E[Fxx^\top F^\top] = FE[xx^\top]F^\top \\ &= F\mathbf{C}_X F^\top \end{aligned} \tag{B.3}$$

La ecuación B.3 es muy importante, esta técnica también es conocida como propagación de errores. Análogamente, para la suma de dos variables aleatorias independientes

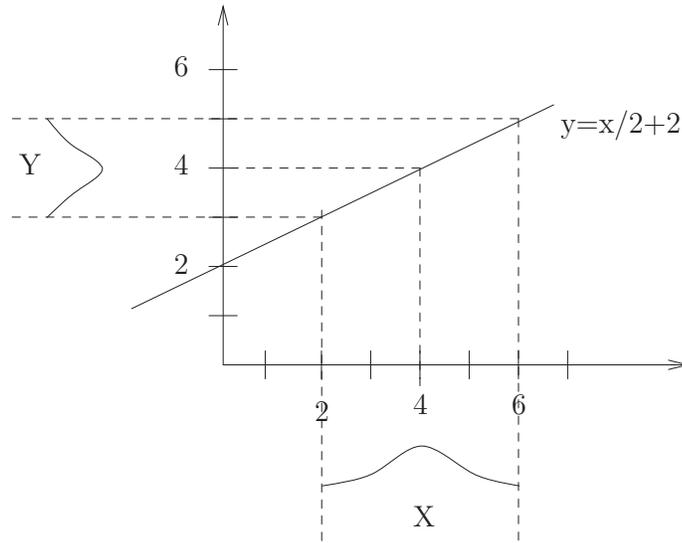


Figura B.1: Transformación Lineal de una variable aleatoria gaussiana

$X_3 = AX_1 + BX_2$  la matriz de covarianza se calcula

$$\mathbf{C}_3 = AC_1A^T + BC_2B^T \quad (\text{B.4})$$

**Ejemplo B.1** Sea  $X$  una variable aleatoria gaussiana con media  $\hat{x} = 4$  y desviación estándar  $\sigma_X = 1$ . Encontrar  $\hat{y}$  y  $\mathbf{C}_Y$ , si  $Y = \frac{X}{2} + 2$ .

Solución. La Fig. B.1 ilustra el problema. La ecuación  $Y = \frac{X}{2} + 2$  se puede escribir en su forma matricial como:

$$[Y] = [0.5 \ 2] \begin{bmatrix} X \\ 1 \end{bmatrix}$$

la matriz de covarianza asociada es

$$\mathbf{C}_X = \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix}$$

entonces, la covarianza  $\mathbf{C}_Y$  es

$$\mathbf{C}_Y = [0.5 \ 2] \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} 0.5 \\ 2 \end{bmatrix} = [0.25]$$

de tal manera que  $\hat{y} = 4$  y  $\sigma_Y = 0.5$ .  $\square$

Las ecuaciones B.3 y B.4 sólo se pueden aplicar para transformaciones lineales. La expansión de Taylor es útil en el caso de transformaciones no lineales. El método de la serie de Taylor simplifica el análisis a expensas de introducir errores por aproximación. Sean  $X = [X_1, \dots, X_p]^\top$  y  $Y = [Y_1, \dots, Y_q]^\top$  dos variables aleatorias, tal que:

$$f(X) = \begin{bmatrix} Y_1 \\ \vdots \\ Y_q \end{bmatrix} = \begin{bmatrix} f_1(X) \\ \vdots \\ f_q(X) \end{bmatrix} \quad (\text{B.5})$$

$Y$  se puede aproximar con la expansión de Taylor de primer orden alrededor de  $\hat{\mathbf{x}} = [\hat{x}_1, \dots, \hat{x}_p]$ . La covarianza [Dossantos-Uzarralde07] es

$$\mathbf{C}_Y = \mathbf{F}_X \mathbf{C}_X \mathbf{F}_X^\top \quad (\text{B.6})$$

donde, el Jacobiano  $\mathbf{F}_X$  está definido por

$$\mathbf{F}_X = \left. \frac{\partial f}{\partial X} \right|_{\hat{\mathbf{x}}} = \begin{bmatrix} \frac{\partial f_1}{\partial X_1} & \cdots & \frac{\partial f_1}{\partial X_p} \\ \vdots & \ddots & \vdots \\ \frac{\partial f_q}{\partial X_1} & \cdots & \frac{\partial f_q}{\partial X_p} \end{bmatrix}_{\hat{\mathbf{x}}} \quad (\text{B.7})$$

Para reducir el error de aproximación, se puede usar la expansión de Taylor de segundo orden [Dossantos-Uzarralde07].

**Ejemplo B.2 Conversión de un punto** [Bailey02]. Sea  $\mathbf{p}_i = [\rho_i, \phi_i]^\top$ , un punto en coordenadas polares; y  $\mathbf{C}_P$ , la matriz de covarianza asociada. Expresa  $\mathbf{p}_i$  en coordenadas rectangulares y calcule la covarianza asociada.

Solución. Para convertir un punto de coordenadas polares a cartesianas se usa,

$$f(\mathbf{p}) = \begin{bmatrix} \rho \cos \phi \\ \rho \sin \phi \end{bmatrix} \quad (\text{B.8})$$

de tal manera que el punto en coordenadas rectangulares se obtiene como  $\mathbf{q}_i = f(\mathbf{p}_i)$ . El Jacobiano se calcula con (B.7), es decir

$$A = \left. \frac{\partial f}{\partial \mathbf{p}} \right|_{\mathbf{p}_i} = \begin{bmatrix} \cos \phi_i & -\rho_i \sin \phi_i \\ \sin \phi_i & \rho_i \cos \phi_i \end{bmatrix}; \quad (\text{B.9})$$

finalmente, aplicando (B.6) se obtiene que  $\mathbf{C}_X = \mathbf{A} \mathbf{C}_P \mathbf{A}^\top$ .  $\square$

**Ejemplo B.3** *Covarianza de una línea a partir de puntos en coordenadas rectangulares.* Sea  $D = \{\mathbf{p}_1, \dots, \mathbf{p}_n\}$  un conjunto de puntos. El  $i$ -ésimo punto  $\mathbf{p}_i \in D$  se representa por sus coordenadas rectangulares  $\mathbf{p}_i = [x_i, y_i]^\top$ . Se desea obtener la línea que mejor represente a  $D$ , así como la matriz de covarianza asociada a la línea.

Solución. Los parámetros de mínimos cuadrados ortogonales son

$$\mathbf{z} = \begin{bmatrix} r \\ \alpha \end{bmatrix} = \begin{bmatrix} \bar{x} \cos \alpha + \bar{y} \sin \alpha \\ \frac{1}{2} \arctan \frac{-2s_{xy}}{s_{yy} - s_{xx}} \end{bmatrix},$$

donde

$$\begin{aligned} \bar{x} &= \frac{1}{n} \sum_{i=1}^n x_i, & \bar{y} &= \frac{1}{n} \sum_{i=1}^n y_i, \\ s_{xx} &= \sum_{i=1}^n (x_i - \bar{x})^2, & s_{yy} &= \sum_{i=1}^n (y_i - \bar{y})^2, \\ s_{xy} &= \sum_{i=1}^n (x_i - \bar{x})(y_i - \bar{y}). \end{aligned}$$

Para obtener la covarianza de la línea  $\mathbf{C}_l$  se puede considerar que las mediciones  $\mathbf{p}_i$  y  $\mathbf{p}_j$  son independientes; entonces  $\mathbf{C}_l = \sum_{i=1}^n B_i \mathbf{C}_{X_i} B_i^\top$ ; donde  $\mathbf{C}_{X_i}$  es la covarianza del  $i$ -ésimo punto y

$$B_i = \begin{bmatrix} \frac{\partial r}{\partial x_i} & \frac{\partial r}{\partial y_i} \\ \frac{\partial \alpha}{\partial x_i} & \frac{\partial \alpha}{\partial y_i} \end{bmatrix}; \quad (\text{B.10})$$

donde

$$\begin{aligned}\frac{\partial \alpha}{\partial x_i} &= \frac{n-1}{n} \cdot \frac{(s_{yy} - s_{xx})(\bar{y} - y_i) + 2s_{xy}(\bar{x} - x_i)}{(s_{yy} - s_{xx})^2 + 4s_{xy}^2} \\ \frac{\partial \alpha}{\partial y_i} &= \frac{n-1}{n} \cdot \frac{(s_{yy} - s_{xx})(\bar{x} - x_i) - 2s_{xy}(\bar{y} - y_i)}{(s_{yy} - s_{xx})^2 + 4s_{xy}^2} \\ \frac{\partial r}{\partial x_i} &= \frac{1}{n} \cos \alpha + (\bar{y} \cos \alpha - \bar{x} \sin \alpha) \frac{\partial \alpha}{\partial x_i} \\ \frac{\partial r}{\partial y_i} &= \frac{1}{n} \sin \alpha + (\bar{y} \cos \alpha - \bar{x} \sin \alpha) \frac{\partial \alpha}{\partial y_i}\end{aligned}$$

□

**Ejemplo B.4** *Covarianza de una línea a partir de puntos en coordenadas polares.* [Garulli04].

Solución. Si se obtiene una línea de mínimos cuadrados a partir del conjunto de puntos  $D = \{\mathbf{p}_1, \dots, \mathbf{p}_n\}$ , donde cada punto se expresa en su forma polar  $\mathbf{p}_i = [\rho_i, \phi_i]^\top$  y se conoce su matriz de covarianza  $\mathbf{C}_P$ . Entonces

$$\mathbf{C}_l = \sum_{i=1}^n J_i \mathbf{C}_P J_i^\top \quad (\text{B.11})$$

donde  $J_i = B_i A_i$ ; y  $A_i$  y  $B_i$  están expresados por las ecs. B.9 y B.10, respectivamente. □

## B.2. El Filtro de Kalman

El KF (del inglés *Kalman Filter*) fue desarrollado por R. E. Kalman [Kalman60]. El KF sirve, entre otras cosas, para predecir el comportamiento de sistemas dinámicos lineales. Un sistema dinámico es un sistema complejo que presenta un cambio o evolución de su estado en un tiempo. Un sistema dinámico se dice discreto si el tiempo se mide en pequeños lapsos.

Consideremos el siguiente modelo discreto de un proceso,

$$\tau_{k+1} = \mathbf{A}\tau_k + \mathbf{B}\mathbf{u}_k + \mathbf{q}_k \quad (\text{B.12})$$

donde  $\tau_k$  representa el estado del sistema,  $\mathbf{u}_k$  son acciones independientes del sistema (usualmente una acción de control) y  $\mathbf{q}_k$  es un ruido gaussiano blanco con covarianza  $\mathbf{Q}_k$ .

**Algoritmo B.1:** El filtro de Kalman (KF)

**Entrada:** La estimación del estado anterior  $\hat{\tau}_k^+$  y  $\mathbf{C}_k^+$ , la acción de control  $\mathbf{u}_k$

**Salida:** La estimación del estado siguiente  $\hat{\tau}_{k+1}^+$  y  $\mathbf{C}_{k+1}^+$

- 1 Calcular la predicción del nuevo estado:  $\hat{\tau}_{k+1}^-$  y  $\mathbf{C}_{k+1}^-$  (ecs. B.13 y B.14)
- 2 Calcular la ganancia de Kalman:  $\mathbf{K}_{k+1}$  (ec. B.17)
- 3 Estimar el nuevo estado y su covarianza:  $\hat{\tau}_{k+1}^+$  y  $\mathbf{C}_{k+1}^+$  (ecs. B.19 y B.20)
- 4 **regresar**  $\hat{\tau}_{k+1}^+$  y  $\mathbf{C}_{k+1}^+$

La predicción del sistema se obtiene como

$$\hat{\tau}_{k+1}^- = \mathbf{A}\tau_k^+ + \mathbf{B}\mathbf{u}_k \quad (\text{B.13})$$

$$\mathbf{C}_{k+1}^- = \mathbf{A}\mathbf{C}_k^+\mathbf{A}^\top + \mathbf{B}\mathbf{U}_k\mathbf{B}^\top + \mathbf{Q}_k \quad (\text{B.14})$$

aquí, las matrices  $\mathbf{C}_k$ ,  $\mathbf{U}_k$  y  $\mathbf{Q}_k$  son las matrices de covarianza de  $\tau_k$ ,  $\mathbf{u}_k$  y  $\mathbf{q}_k$ , respectivamente. Note que en (B.13) y en (B.14) sólo se usa el modelo de movimiento.

En cada instante se realiza una medición (observación) del estado del sistema, representada por  $\mathbf{z}_{k+1}$ , y definida como:

$$\mathbf{z}_{k+1} = \mathbf{H}\tau_{k+1} + \mathbf{r}_{k+1} \quad (\text{B.15})$$

donde  $\mathbf{r}_k$  es un ruido gaussiano blanco con covarianza  $\mathbf{R}_k$ .

Para obtener el estimado después de incorporar las mediciones obtenidas se puede calcular la *innovación* (el error entre la predicción y la observación)

$$\mathbf{v}_{k+1} = \mathbf{z}_{k+1} - \mathbf{H}\hat{\tau}_{k+1}^-; \quad (\text{B.16})$$

y la ganancia de Kalman

$$\mathbf{K}_{k+1} = \mathbf{H}\mathbf{C}_{k+1}^-\mathbf{H}^\top + \mathbf{R}_{k+1}. \quad (\text{B.17})$$

Finalmente, se calcula la actualización

$$\mathbf{W}_{k+1} = \mathbf{C}_{k+1}^-\mathbf{H}^\top\mathbf{K}_{k+1}^{-1}; \quad (\text{B.18})$$

$$\tau_{k+1}^+ = \tau_{k+1}^- + \mathbf{v}_{k+1}\mathbf{W}_{k+1} \quad (\text{B.19})$$

$$\mathbf{C}_{k+1}^+ = \mathbf{C}_{k+1}^- - \mathbf{W}_{k+1}\mathbf{K}_{k+1}\mathbf{W}_{k+1}^\top \quad (\text{B.20})$$

**Algoritmo B.2:** El filtro de Kalman Extendido (EKF)

**Entrada:** La estimación del estado anterior  $\hat{\tau}_k^+$  y  $\mathbf{C}_k^+$ , la acción de control  $\mathbf{u}_k$

**Salida:** La estimación del estado siguiente  $\hat{\tau}_{k+1}^+$  y  $\mathbf{C}_{k+1}^+$

- 1 Calcular la predicción del nuevo estado:  $\hat{\tau}_{k+1}^-$  y  $\mathbf{C}_{k+1}^-$  (ecs. B.23 y B.24)
- 2 Calcular la ganancia de Kalman:  $\mathbf{K}_{k+1}$  (ec. B.26)
- 3 Estimar el nuevo estado y su covarianza:  $\hat{\tau}_{k+1}^+$  y  $\mathbf{C}_{k+1}^+$  (ecs. B.19 y B.20)
- 4 **regresar**  $\hat{\tau}_{k+1}^+$  y  $\mathbf{C}_{k+1}^+$

**B.3. El Filtro de Kalman Extendido**

El Filtro de Kalman es un estimador recursivo lineal; sin embargo, muchos sistemas evolucionan en el tiempo de acuerdo a un modelo no lineal del proceso:

$$\tau_{k+1} = \mathbf{g}(\mathbf{u}_k, \tau_k) + \mathbf{q}_k \quad (\text{B.21})$$

$$\mathbf{z}_{k+1} = \mathbf{h}(\tau_{k+1}) + \mathbf{r}_{k+1} \quad (\text{B.22})$$

donde  $\mathbf{u}_k$  es la acción de control y  $\mathbf{q}_k$  es un vector de ruido correspondiente a errores en el modelo del sistema. A diferencia de un sistema lineal, para calcular covarianzas es necesario linealizar –como se indica en (B.6). De tal manera que el paso de predicción en el EKF es

$$\hat{\tau}_{k+1}^- = \mathbf{g}(\mathbf{u}_k, \hat{\tau}_{k-1}) \quad (\text{B.23})$$

$$\mathbf{C}_{k+1}^- = \mathbf{G}_\tau \mathbf{C}_k^+ \mathbf{G}_\tau^\top + \mathbf{G}_u \mathbf{U}_k \mathbf{G}_u^\top + \mathbf{Q}_k \quad (\text{B.24})$$

aquí los jacobianos son  $\mathbf{G}_\tau = \frac{\partial \mathbf{f}}{\partial \tau} \Big|_{\hat{\tau}, \hat{\mathbf{u}}_k}$  y  $\mathbf{G}_u = \frac{\partial \mathbf{f}}{\partial \mathbf{u}} \Big|_{\hat{\tau}, \hat{\mathbf{u}}_k}$ . De forma análoga, se calculan

$$\mathbf{v}_{k+1} = \mathbf{z}_{k+1} - \mathbf{h}(\hat{\tau}_{k+1}^-); \quad (\text{B.25})$$

$$\mathbf{S}_{k+1} = \mathbf{H}_\tau \mathbf{C}_{k+1}^- \mathbf{H}_\tau^\top + \mathbf{R}_k \quad (\text{B.26})$$

$$\mathbf{W}_{k+1} = \mathbf{C}_{k+1}^- \mathbf{H}_\tau^\top \mathbf{S}_{k+1}^{-1}, \quad (\text{B.27})$$

El algoritmo B.2 describe una iteración del filtro de Kalman extendido.



## Apéndice C

# Publicaciones del autor

A continuación se muestran los distintos artículos científicos publicados en el trabajo desarrollado en esta tesis (se ordenan por fecha):

1. **Lara C. and Romero L. Finding Multiple Lines from 2D Range Data. The 15th IASTED International Conference on Robotics and Applications. Nov 1-3, 2010.** Este artículo expone la razón de verosimilitud, el desarrollo es el mismo que el mostrado en el capítulo 2 de esta tesis. Para obtener el mapa se usa el SLAM estocástico que se introduce en el capítulo 3.
2. **Romero L. and Lara C. Line Maps in Cluttered Environments. MICA 2010, Pachuca México. Nov 8-13, 2010.** Este artículo también discute la razón de verosimilitud; la principal diferencia con el desarrollo presentado en esta tesis es que en el artículo se deduce la razón de verosimilitud partiendo de la búsqueda del mejor número de marcas. Para encontrar el mapa de líneas se usa el método SAM [Kaess09]; y para mejorar el criterio de compatibilidad conjunta se usa una heurística basada en las longitudes de los segmentos de líneas.

3. Lara C., Romero L., Flores J.J., Gomez C. **A Simple Sample Consensus Algorithm to Find Multiple Models.** *Lecture Notes in Computer Science. Progress in Pattern Recognition, Image Analysis and Applications.* Páginas 918-925. Este artículo aborda el problema de obtener un conjunto de líneas a partir de una secuencia de mediciones de rango; el algoritmo propuesto usa un enfoque frecuentista que considera la relación entre las diferentes marcas. Además, el artículo explica la distancia direccional –útil para detectar la *secuencia de predicción* que se usa en el capítulo 4.
4. Lara C., Romero L. **A Robust Iterative Closest Point Algorithm with Augmented Features.** *MICAI 2008: Advances in Artificial. Volume 5317/2008.* Páginas 605-614. En este artículo se usa el enfoque de asociación datos-datos para resolver el problema de registro entre dos secuencias de mediciones de rango. Para mejorar el algoritmo clásico ICP se usa la información de las normales en una determinada vecindad. El enfoque propuesto en este artículo es ortogonal a la idea que se propone a lo largo de esta tesis –mientras que en este artículo se usan puntos con información de marcas, en la tesis se usan marcas que usan información de puntos.
5. Lara C., y Romero, L. **Robust Local Localization of a Mobile Robot in Indoor Environments Using Virtual Corners.** *Progress in Pattern Recognition, Image Analysis and Applications. Lecture Notes in Computer Science. Volume 4756/2008. In proceedings of the 12th Iberoamerican Congress on Pattern Recognition. CIARP 2007. Viña del Mar-Valparaiso, Chile. Pages 901-910. Springer Berlin / Heidelberg.* Este artículo trata sobre la estimación del movimiento relativo de un robot móvil; para ello, se usan dos secuencias de mediciones de rango obtenidas en posiciones adyacentes del robot. El método propuesto extrae líneas de cada una de las imágenes y luego encuentra las *esquinas virtuales*; una esquina virtual es la intersección entre dos líneas cualquiera. El movimiento del robot se estima asociando esquinas de una imagen con esquinas de la otra; la mejor estimación se encuentra usando un método de selección aleatoria y una función objetivo frecuentista.

A lo largo de mis estudios doctorales también se han publicado artículos en otras áreas, a decir:

1. Lara C., Flores J.J., and Calderon F.. **On the Hyperbox - Hyperplane Intersection Problem**. INFOCOMP Journal of Computer Science. Vol 8. Num 4. p. 21 – 27. Dic 2009. ISSN: 1807-4545.
2. C. Lara, J.J. Flores and F. Calderon. **A School Timetabling Solution Using Bee Algorithm**. MICAI 2008. México D.F.
3. Flores J.J. and Lara C. **Solución al Problema de Asignación de Horarios Escolares usando un Algoritmo de Abejas**. In proceedings of the 3rd International Congress of Management Quality and Enterprise Competitiveness, Sept. 2008, Morelia, Mich.



# Referencias

- [Bailey02] Bailey, T. *Mobile Robot Localisation and Mapping in Extensive Outdoor Environments*. Tesis Doctoral, Australian Centre for Field Robotics, The University of Sydney, 2002.
- [Bolles81] Bolles, R. C. y Fischler, M. A. A ransac-based approach to model fitting and its application to finding cylinders in range data. *En IJCAI*, págs. 637–643. 1981.
- [Borges00] Borges, G. A. A split-and-merge segmentation algorithm for line extraction in 2-d range images. *En ICPR '00: Proceedings of the International Conference on Pattern Recognition*, pág. 1441. IEEE Computer Society, Washington, DC, USA, 2000.
- [C. Cadena09] C. Cadena, F. R. y Neira, J. Efficient large scale SLAM including data association using the combined filter. *En European Conference on Mobile Robotics, ECMR '09*. Mlini/Dubrovnik, Croatia, September 2009.
- [Dossantos-Uzarralde07] Dossantos-Uzarralde, P., Mounier, P., Brassart, M., y Couvert, M. Covariance matrices for cross sections issued from optical model calculations. *En International Conference on Nuclear Data for Science and Technology*, págs. 297–300. 2007.
- [Duda73] Duda, R. O., Hart, P. E., y Stork, D. G. *Pattern classification and scene analysis*. Wiley New York, 1973.

- [Elfes89] Elfes, A. *Occupancy grids: a probabilistic framework for robot perception and navigation*. Tesis Doctoral, Carnegie Mellon University, Pittsburgh, PA, USA, 1989.
- [Feder99] Feder, H. J. S., Leonard, J. J., y Smith, C. M. Adaptive mobile robot navigation and mapping. *I. J. Robotic Res.*, 18(7):650–668, 1999.
- [Garulli04] Garulli, A., Giannitrapani, A., Rossi, A., y Vicino, A. Mobile robot SLAM for line-based environment representation. *En In Proceedings of 44th IEEE Conference on Decision and Control and European Control Conference - IEEE CDC–ECC*, págs. 2041–2046. 2004.
- [Guo04] Guo, A. y Siegelmann, H. T. Time-warped longest common subsequence algorithm for music retrieval. *En ISMIR 2004, 5th International Conference on Music Information Retrieval*. 2004.
- [Han04] Han, F., Tu, Z., y Zhu, S.-C. Range image segmentation by an effective jump-diffusion method. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 26(9):1138–1153, 2004. ISSN 0162-8828. doi: <http://doi.ieeecomputersociety.org/10.1109/TPAMI.2004.70>.
- [Hartley00] Hartley, R. y Zisserman, A. *Multiple view geometry in computer vision*. Cambridge University Press, New York, NY, USA, 2000. ISBN 0-521-62304-9.
- [Heller05] Heller, K. A. y Ghahramani, Z. Bayesian hierarchical clustering. *En ICML*, págs. 297–304. 2005.
- [Kaess09] Kaess, M. y Dellaert, F. Covariance recovery from a square root information matrix for data association. *Journal of Robotics and Autonomous Systems, RAS*, 57:1198–1210, Dec 2009.

- [Kalman60] Kalman, Rudolph, y Emil. A new approach to linear filtering and prediction problems. *Transactions of the ASME—Journal of Basic Engineering*, 82(Series D):35–45, 1960.
- [Kalman61] Kalman, R. E. y Bucy, R. S. New results in linear filtering and prediction theory. *Transactions of the ASME. Series D, Journal of Basic Engineering*, 83:95–107, 1961.
- [Kuipers91] Kuipers, B. y Byun, Y.-T. A robot exploration and mapping strategy based on a semantic hierarchy of spatial representations. *Journal of Robotics And Autonomous Systems*, 8:47–63, 1991.
- [Lara-Alvarez09] Lara-Alvarez, C., Romero, L., Flores, J. F., y Gomez, C. A simple sample consensus algorithm to find multiple models. En E. Bayro-Corrochano y J.-O. Eklundh, eds., *CIARP*, tomo 5856 de *Lecture Notes in Computer Science*, págs. 918–925. Springer, 2009. ISBN 978-3-642-10267-7.
- [Lara06] Lara, C. *Construcción de mapas geométricos bidimensionales de ambientes interiores con un telémetro láser*. Proyecto Fin de Carrera, Universidad Michoacana, Facultad de Ingeniería Eléctrica, Morelia, Michoacán. México, Junio 2006.
- [Lara07] Lara, C. y Romero, L. Robust local localization of a mobile robot in indoor environments using virtual corners. En D. M. Luis Rueda y J. Kittler, eds., *Progress in Pattern Recognition, Image Analysis and Applications. 12th Iberoamerican Congress on Pattern Recognition, CIARP 2007*, págs. 901–910. Springer-Verlag, 2007.
- [Lara08] Lara, C., Romero, L., y Calderón, F. A robust iterative closest point algorithm with augmented features. En *MICAI '08: Proceedings of the 7th Mexican International Conference on Artificial Intelligence*, págs. 605–614. Springer-Verlag, Berlin, Heidelberg, 2008. ISBN 978-3-540-88635-8.

- [Lara09] Lara, C., Flores, J. J., y Calderon., F. On the hyperbox – hyperplane intersection problem. *INFOCOMP Journal of Computer Science*, 8(4):21–27, 2009.
- [Leavers93] Leavers, V. F. Which hough transform? *CVGIP: Image Underst.*, 58(2):250–264, 1993. ISSN 1049-9660. doi: <http://dx.doi.org/10.1006/ciun.1993.1041>.
- [Leonard91] Leonard, J. y Durrant-Whyte, H. Simultaneous map building and localization for an autonomous mobile robot. *En Intelligent Robots and Systems '91. 'Intelligence for Mechanical Systems, Proceedings IROS '91. IEEE/RSJ International Workshop on*, págs. 1442–1447 vol.3. Nov 1991. doi:10.1109/IROS.1991.174711.
- [Martinelli04] Martinelli, A. y Tomatis, N. Open challenges in SLAM: An optimal solution based on shift and rotation invariants. *En In Proceedings of the International Conference on Robotics and Automation - ICRA*, págs. 1327–1332. 2004.
- [Montemerlo05] Montemerlo, M. y Thrun, S. FastSLAM 2.0: An improved particle filtering algorithm for simultaneous localization and mapping that provably converges, 2005.
- [Neira01] Neira, J. y Tardós, J. Data association in stochastic mapping using the joint compatibility test. *IEEE Transactions on Robotics and Automation*, 17(6):890–897, December 2001.
- [Nguyen07] Nguyen, V., Gächter, S., Martinelli, A., Tomatis, N., y Siegwart, R. A Comparison of Line Extraction Algorithms using 2D Range Data for Indoor Mobile Robotics. *Autonomous Robots*, 23(2):97–111, August 2007.
- [Nieto05] Nieto, J., Bailey, T., y Nebot, E. M. Scan-SLAM: Combining

- EKF-SLAM and scan correlation. *En International Conference on Field and Service Robotics*, págs. 167–178. 2005.
- [Romero01] Romero, L. *Construcción de mapas y localización global de robots móviles: un enfoque probabilista*. Tesis Doctoral, ITESM, Campus Cuernavaca, Cuernavaca, Morelos, Mexico, 2001.
- [Romero05] Romero, L. y Arellano, J. J. Robust local localization of a mobile robot using a 2-D laser range finder. *En ENC '05: Proceedings of the Sixth Mexican International Conference on Computer Science*, págs. 248–255. IEEE Computer Society, Washington, DC, USA, 2005. ISBN 0-7695-2454-0. doi: <http://dx.doi.org/10.1109/ENC.2005.37>.
- [Rousseeuw87] Rousseeuw, P. J. y Leroy, A. M. *Robust regression and outlier detection*. John Wiley & Sons, Inc., New York, NY, USA, 1987.
- [Schnabel07] Schnabel, R., Wahl, R., y Klein, R. Efficient ransac for point-cloud shape detection. *Computer Graphics Forum*, 26(2):214–226, jun. 2007.
- [Siegwart04] Siegwart, R. y Nourbakhsh, I. R. *Introduction to Autonomous Mobile Robots*. Bradford Book, 2004. ISBN 026219502X.
- [Simpson05] Simpson, R. C. Smart wheelchairs: A literature review. *Journal of Rehabilitation Research and Development*, 42(4):423 – 436, 2005.
- [Sivia96] Sivia, D. S. *Data Analysis: A Bayesian Tutorial (Oxford Science Publications)*. Oxford University Press, July 1996. ISBN 0198518897.
- [Smith87] Smith, R., Self, M., y Cheeseman, P. Estimating uncertain spatial relationships in robotics. *En Robotics and Automation. Proceedings. 1987 IEEE International Conference on*, tomo 4, pág. 850. 1987.

- [Stachniss09] Stachniss, C. *Robotic Mapping and Exploration*. Springer Publishing Company, Incorporated, 2009. ISBN 3642010962, 9783642010965.
- [Tardós02] Tardós, J., Neira, J., Newman, P. M., y Leonard, J. J. Robust mapping and localization in indoor environments using sonar data. *International Journal of Robotics Research*, 21(4):311–330, 2002.
- [Thrun98] Thrun, S. Finding landmarks for mobile robot navigation. *En In Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, págs. 958–963. 1998.
- [Thrun99] Thrun, S., Bennewitz, M., Burgard, W., Cremers, A. B., Dellaert, F., Fox, D., Haehnel, D., Rosenberg, C., Roy, N., Schulte, J., y Schulz, D. Minerva: A second generation mobile tour-guide robot. *En Proc. of the IEEE International Conference on Robotics and Automation (ICRA '99)*. 1999.
- [Thrun00] Thrun, S., Beetz, M., Bennewitz, M., Burgard, W., Cremers, A., Dellaert, F., Fox, D., Hahnel, D., Rosenberg, C., Roy, N., Schulte, J., y Schulz, D. Probabilistic algorithms and the interactive museum tour-guide robot minerva. *International Journal of Robotics Research*, 19(1):972–999, November 2000.
- [Thrun03] Thrun, S., Martin, C., Liu, Y., Hähnel, D., Emery-Montemerlo, R., Chakrabarti, D., y Burgard, W. A real-time expectation maximization algorithm for acquiring multi-planar maps of indoor environments with mobile robots. *IEEE Transactions on Robotics and Automation*, 20(3):433–442, 2003.
- [Thrun05] Thrun, S., Burgard, W., y Fox, D. *Probabilistic Robotics (Intelligent Robotics and Autonomous Agents)*. The MIT Press, September 2005. ISBN 0262201623.

- 
- [Tipaldi10] Tipaldi, G. D. y Arras, K. O. FLIRT - interest regions for 2D range data. *En Proc. of the IEEE International Conference on Robotics and Automation (ICRA '2010)*. 2010.
- [Tomatis08] Tomatis, N. Hybrid, metric-topological representation for localization and mapping. *En Robotics and Cognitive Approaches to Spatial Mapping*, págs. 43–63. Springer, 2008.
- [Washington99] Washington, R., Golden, K., Bresina, J., Smith, D. E., Anderson, C., y Smith, T. Autonomous rovers for mars exploration. *En In Proceedings of The 1999 IEEE Aerospace Conference*, págs. 237–251. 1999.