



# **UNIVERSIDAD NACIONAL DEL SUR**

TESIS DE DOCTOR EN CONTROL DE SISTEMAS

ESTIMADOR NO LINEAL PARA REDES DE SENSORES

ING. SAÑUDO, SILVANA ROMINA

BAHIA BLANCA

ARGENTINA

2011

## **PREFACIO**

Esta Tesis se presenta como parte de los requisitos para optar al grado Académico de Doctor en Control de Sistemas, de la Universidad Nacional del Sur y no ha sido presentada previamente para la obtención de otro título en esta Universidad u otra. La misma contiene los resultados obtenidos en investigaciones llevadas a cabo en el ámbito del Departamento de Ingeniería Eléctrica y Computadoras durante el período comprendido entre el 15 de Agosto de 2006 y el 30 de Marzo de 2011, bajo la dirección de Doctor en Control de Sistemas Favio Román Masson.

[Firma del Alumno]



UNIVERSIDAD NACIONAL DEL SUR  
Secretaría General de Posgrado y Educación Continua

La presente tesis ha sido aprobada el ..../..../..... , mercedo  
la calificación de .....(.....)

En el entorno académico, en primer lugar me gustaría expresar mi sentido agradecimiento a mi Director, el Dr. Favio Masson, por su claridad, disponibilidad, por aportar sus conocimientos y ser una excelente guía, pero a la vez darme libertad para explorar las diferentes posibilidades del problema. También agradezco que me haya transmitido su entusiasmo, visión de futuro y motivación en el tema abordado, así como en lo cotidiano. Agradezco a la gente, compañeros y técnicos, del Grupo Investigación en Sistemas Electrónicos y Electromecatrónicos y el Instituto de Investigaciones en Ingeniería Eléctrica, especialmente al Dr. Pedro Julian y al Dr. Pablo Mandolesi, por acompañarme y aportar sus conocimientos y experiencia en tantas oportunidades. Agradezco a mis compañeros del Laboratorio de Control y Robótica y aledaños por su compañía, por hacer más amena mi estadía en la Universidad Nacional del Sur y en los viajes a congresos. El aporte financiero de esta investigación lo realizó la Agencia de Ciencia y Tecnología en su primera instancia, y luego CONICET. El agradecimiento más sincero e infinito a mi familia y amigos que estuvieron presentes en todo el transcurso de esta tesis y de la vida, acompañando con su apoyo incondicional, sus enseñanzas, sus consejos y su cariño. Muy profundamente, agradezco a mis padres que se sacrificaron para que pueda llevar a cabo esta carrera, por todo lo que me han dado, no tengo palabras, los amo. Con la alegría de llegar a la recta final, deseo dejar constancia de mi agradecimiento a todas y cada una de las personas que de una forma u otra han ayudado a que esto fuera realidad.

**Resumen:** En redes de sensores el bajo consumo es una prioridad, y la búsqueda del consumo reducido impone restricciones muy fuertes en todos los bloques del nodo: sensor, procesador y unidad de comunicación. Debido a esto, el procesador no cuenta con grandes capacidades de cómputo ni de almacenamiento de datos.

La localización y seguimiento de fuentes u objetos son las aplicaciones principales en redes de sensores. La limitación fundamental en la actualidad es la falta de un algoritmo de estimación que resuelva estos problemas de forma completa y descentralizada. Esta tesis presenta una solución a este tema basada en un Filtro de Partículas. En general, los filtros de partículas no pueden implementarse directamente en una red de sensores por sus exigencias en términos de comunicación, memoria y procesamiento. La solución propuesta, el Filtro de Partículas Acotado, resuelve en forma completa todos los temas mencionados. Este filtro cumple con los requerimientos impuestos por las redes de sensores, y permite realizar la estimación utilizando medidas ruidosas de varios sensores no muy precisos, explotando la fusión de la información de varios nodos para obtener un estimado, aún cuando los modelos de los sensores presentan múltiples hipótesis de medida.

Un elemento esencial de una red de sensores es su capacidad de cobertura espacial y capacidad de comunicación. Ambos factores están limitados por el consumo de potencia disponible. El Filtro de Partículas Acotado es capaz de capturar la esencia de la estimación reduciendo al mínimo la información (número de parámetros) a transmitir ó recibir, gracias al almacenamiento de variables locales. Esto es fundamental para producir una solución al problema de localización y seguimiento, bajo restricciones reales de consumo.

**Abstract:** Low power consumption is a priority in sensor networks, because it imposes strong constraints in all node parts: sensor, processor and communication unit. Due to this, the processor has low computational and data storage capabilities. Sources and/or objects localization and tracking are the primary applications in sensor networks.

A fundamental limitation nowadays is the lack of an estimation algorithm able to solve the estimation problem in a comprehensive and decentralized way. This thesis presents a solution to this issue, based on a Particle Filter. In general, particle filters cannot be implemented in Sensor networks due to the requirements in terms of communication, memory and processing. The proposed solution is the Bounded Particle Filter algorithm, which solves all mentioned issues in a complete way. This filter fulfills sensor networks requirements, and allows the estimation using noisy measurements, exploiting the fusion of many nodes information to produce an estimate, even with multiple hypothesis sensor models.

An essential element of a sensor network is its spatial coverage and communication capacity. Both factors are limited by the available power consumption. The proposed Bounded Particle Filter can capture the essence of the estimation minimizing the information to transmit (number of parameters) or receive, thanks to the local storage of variables. This is essential for producing a solution to the problem of tracking and localization, under realistic power constraints.

# Tabla de Contenidos

Tabla de Contenidos	VI
<b>1. Introduction</b>	<b>2</b>
1.1. Motivación . . . . .	2
1.2. Enfoque. Propuesta. . . . .	6
1.3. Contribuciones . . . . .	7
1.4. Organización de la Tesis . . . . .	8
<b>2. Redes de Sensores</b>	<b>9</b>
2.1. Introducción . . . . .	9
2.2. Paradigma Redes de Sensores . . . . .	14
2.2.1. Plataforma Utilizada en las Aplicaciones. . . . .	18
2.3. Implementaciones destacadas. . . . .	19
2.4. Conclusiones. . . . .	22
<b>3. Filtro de Partículas</b>	<b>23</b>
3.1. Introducción . . . . .	23
3.2. Fusión de Datos. . . . .	26
3.2.1. Fusión de Datos en Redes de Sensores. . . . .	27
3.2.2. Trabajos relacionados . . . . .	28
3.3. Filtrado Recursivo de Bayes. . . . .	28
3.3.1. Algoritmo Bayesiano Recursivo Exacto . . . . .	30
3.4. Métodos Secuenciales de Monte Carlo . . . . .	32
3.4.1. Trabajos Relacionados . . . . .	40
3.5. Conclusiones . . . . .	45
<b>4. Filtro de Partículas Acotado</b>	<b>46</b>
4.1. Filtro de Partículas Acotado . . . . .	46
4.1.1. Propuesta. . . . .	46
4.1.2. Algoritmo General. . . . .	47
4.2. Aplicaciones de Localización y Seguimiento . . . . .	55
4.2.1. Sensores de Rango y Ángulo Utilizados . . . . .	57

4.2.2. Filtro de Partículas Acotado en Coordenadas Cartesianas . . . . .	57
4.2.3. Filtro de Partículas Acotado en Coordenadas Polares . . . . .	66
4.2.4. Implementación sobre Nodos de una Red de Sensores . . . . .	90
4.3. Contribuciones . . . . .	109
4.4. Conclusiones . . . . .	110
<b>5. Conclusiones</b>	<b>112</b>
5.1. Resumen del trabajo . . . . .	112
5.2. Contribuciones . . . . .	113
5.3. Líneas futuras . . . . .	114
<b>Bibliografía</b>	<b>115</b>
<b>Bibliografía</b>	<b>115</b>

# Lista de Algoritmos

1.	Algoritmo de Muestreo Secuencial por Importancia. . . . .	36
2.	Algoritmo Filtro de Partículas. . . . .	39
3.	Adaptación de la cota de peso <i>CotaP</i> . . . . .	52
4.	Algoritmo Filtro de Partículas Acotado. . . . .	54



# Capítulo 1

## Introduction

### 1.1. Motivación

Los avances en Sistemas Micro Electro Mecánicos (MEMs) y redes inalámbricas han hecho posible la creación de pequeños nodos sensores multi-funcionales de comunicación inalámbrica, bajo costo, poca cobertura de sensado y comunicación limitada; poca capacidad de procesamiento y almacenamiento, bajo ancho de banda y bajo consumo de energía. Los nodos pueden tener uno o mas sensores según sea necesario. Las redes conformadas por dichos nodos, denominadas *Redes de Sensores Inalámbricas* (WSN, por su nombre en inglés, Wireless Sensor Networks)[Yick et al. (2008)], son un paradigma de la medición distribuida; los nodos colectan información física del ambiente y la comunican. Se utilizan en áreas donde no es posible montar redes comunes, ya sea por cuestiones del ambiente (ambiente hostil, tóxico, poco accesible, con obstáculos, entre otros) o razones estratégicas.

Inicialmente la investigación en WSN apuntó a desarrollar prototipos de hardware y protocolos de red eficientes en energía; dichos esfuerzos permitieron que hoy día se califiquen las redes como de gran escala, inalámbricas, *ad hoc*, multi-hop (de múltiples caminos recorridos mediante el salto de nodo en nodo y la utilización de nodos intermedios para la comunicación de dos nodos lejanos), de nodos pequeños, íntegramente comunicadas, mayoritariamente inmóviles que pueden ser aleatoriamente diseminadas sobre un área de interés, etcétera. Para lograr dichas características, el común de los

nodos poseen limitaciones físicas como restricciones en su capacidad de procesamiento, memoria y comunicación, por ello en general los nodos realizan tareas simples sin llegar a un resultado, solo a un pre-procesamiento de la información; la mayor parte del procesamiento se realiza en una computadora. Hace un tiempo ya que las WSN se consideran para aplicaciones civiles, incluso en Septiembre de 1999, Business Week publicó que era una de las tecnologías más importantes del siglo XXI [Week (1999)].

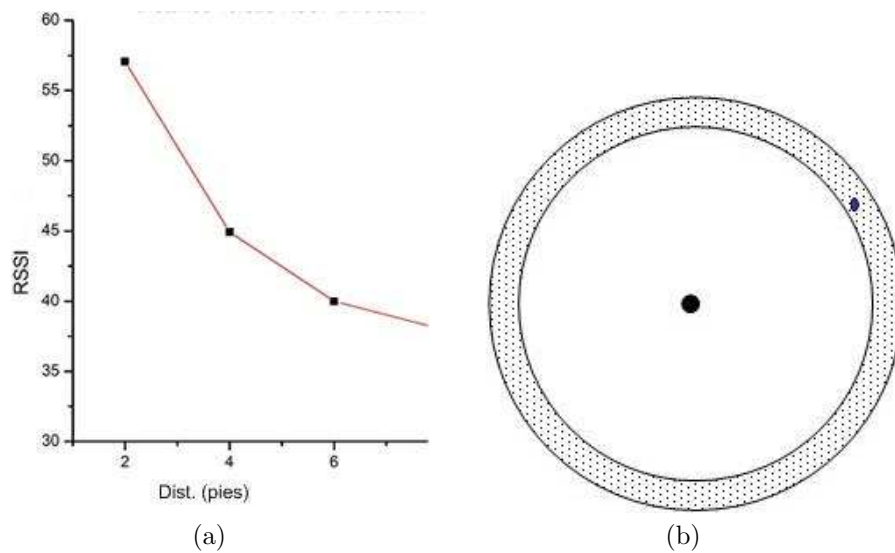
Se han considerado una herramienta científica para monitoreo de fenómenos físicos sin necesidad de la intervención humana, permitiendo estudiar el comportamiento y especies del mundo real. El monitoreo también es útil para mejorar la eficiencia de procesos industriales y como mecanismo para indagar en problemas de protección del medio ambiente y reforzar las leyes en ese campo. De manera más general, la red en conjunto trabaja con un fin determinado.

Otro de los objetivos más comunes es la localización (o detección) de una fuente, objeto u evento y su seguimiento, que es la aplicación que se aborda en la presente tesis. El movimiento de la fuente u objeto sigue trayectorias arbitrarias y continuas; y la tarea principal de la red es medir su presencia y estimar a partir de ésta, su ubicación. Como las medidas poseen incertezas y las dinámicas de los modelos no son conocidas perfectamente, los métodos probabilísticos son una forma de plantear estos problemas. Los objetos o fuentes a localizar y/o seguir tienen dinámica (modelo de movimiento) no lineal y no Gaussiana, además el modelo de observación de los sensores es no lineal, conteniendo ruidos no Gaussianos y las funciones de distribución de probabilidad son arbitrarias.

Las características de los sensores utilizados también modifica las funciones de distribución de probabilidad del sistema. Para ejemplificar esto se puede analizar que sucede con medidas provenientes de la potencia de RF recibida (RSSI).

La Fig. 1.1(a) muestra un modelo ideal de la variación de la potencia con la distancia. Esto es, la potencia cae a partir de que se aleja del transmisor con una ley inversa a la distancia. Es una medida de rango donde conociendo la potencia con la que se transmite, se puede obtener una distancia relativa entre transmisor y receptor. En el

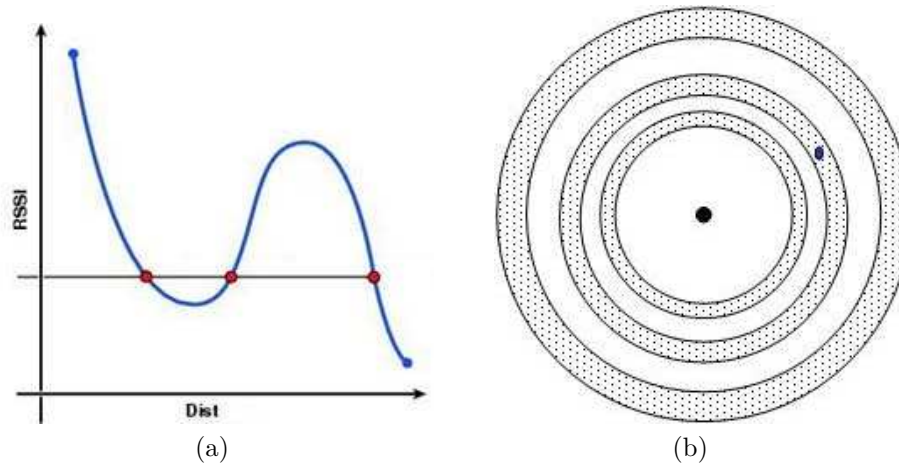
caso mostrado en la figura, para cada valor de potencia se posee un único valor posible de distancia. La Fig. 1.1(b) muestra otra perspectiva, donde la fuente es representada por un punto negro y el receptor por uno azul. En este caso se ha incorporado la incertidumbre en la medida. La zona grisada representa un área alrededor de la potencia/distancia recibida con una determinada probabilidad que el receptor se encuentre en ella. Si la medida de rango tuviera una incertidumbre gaussiana, la zona grisada puede representar dos veces el valor de la desviación estándar y la probabilidad que el receptor se encuentre allí es del 95 %.



**Figura 1.1:** Caso de hipótesis simple: (a) Curva RSSI-Distancia, (b) Representación de la medida, en negro el sensor y en azul el objeto a localizar.

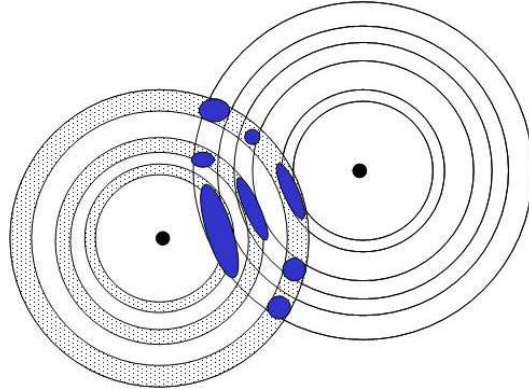
En un escenario real, la señal de RF llega al receptor por varios caminos. La señal llega en forma directa pero también a partir de reflexiones en el suelo o la infraestructura cercana así como por efecto de la refracción. El efecto en el receptor es que la señal se compone de la suma de los distintos caminos en forma constructiva y/o destructiva. En este caso, si se construye un modelo simple que tenga en cuenta este fenómeno, se obtendría uno como el mostrado en la Fig. 1.2(a). En este caso, se puede observar que para un valor determinado de potencia medida, hay tres posibles distancias. A semejanza de la Fig. 1.2(b), si se gráfica esto desde la perspectiva de la fuente para una posición del receptor y considerando incertidumbre en las medidas, la zona grisada

será la que tendrá una probabilidad determinada donde encontrar el receptor.



**Figura 1.2:** Caso de hipótesis múltiples: (a) Curva RSSI-Distancia, (b) Representación de la medida, en negro el sensor y en azul el objeto a localizar.

Si se fusiona la información proveniente de dos sensores con la característica apuntada en la Fig. 1.2, el resultado de la estimación debería ser como el mostrado en la Fig. 1.3. En ella se observan las mediciones individuales con sus respectivas probabilidades. La unión de ambas informaciones debería resultar en múltiples hipótesis de donde se encuentra el receptor, que allí son mostradas en azul. Indudablemente este tipo de estimaciones no son gaussianas lo que hace imposible tratar al problema con filtros como el de Kalman. Algunas alternativas consideran tratar al problema como una suma de gaussianas para así utilizar al filtro de Kalman. Sin embargo estas requieren de un gran procesamiento en la etapa de remuestreo de la distribución final y que imposibilita su implementación en los nodos de una red de sensores.



**Figura 1.3:** Representación de la medida para hipótesis múltiples en la fusión de dos sensores, en negro el sensor y en azul las diferentes hipótesis.

Para este tipo de problemática se ha vuelto popular la utilización del *Filtro de Partículas* (PF por su nombre en inglés, Particle Filter); ya que otros filtros estimadores abordan modelos lineales, ruidos Gaussianos ó bien requieren información precisa para su inicialización. La principal desventaja del PF es que requiere comunicar mucha información para transmitir la estimación.

Es importante tener en cuenta que la transmisión de datos en un nodo inalámbrico de una red de sensores es la tarea que mas energía consume, más cuando cada nodo transmite sus datos a una central para que realice la fusión. En la presente tesis se considera una forma descentralizada, nodo a nodo, de fusión.

## 1.2. Enfoque. Propuesta.

Esta tesis estudia sistemas de localización de fuentes u objetos que presentan modelos no lineales de distribución aleatoria, se propone como base para el diseño del algoritmo de estimación el Filtro de Partículas, ya que presenta una forma simple y efectiva de representar modelos de procesos estocásticos y modelos de propagación arbitrarios con funciones de distribución de probabilidad arbitrarias; lo que lo hace la herramienta adecuada para la tarea que se desea abordar. Una vez comprendida la implementación del Filtro de Partículas, se adapta el algoritmo de modo de poder implementarlo sobre un nodo comercial, que cumpla con las restricciones que el nodo y

la red de sensores presentan.

Una de las formas de alcanzar el objetivo de localización es enviando toda la información de los nodos a una central de procesamiento. En la presente tesis se considera una forma alternativa de lograr dicho objetivo basada en un procesamiento distribuido dentro de cada nodo y se logra una baja en los recursos de comunicación y energía consumidos. Una estimación del parámetro de interés es comunicada de nodo a nodo y cada nodo actualiza el parámetro ajustando la estimación anterior.

### 1.3. Contribuciones

La presente tesis provee:

- Presentación de un algoritmo de estimación para redes de sensores, el Filtro de Partículas Acotado (BPF, por su nombre en inglés, Bounded Particle Filter), para aplicaciones de localización y seguimiento en redes de sensores. El algoritmo permite realizar una estimación conjunta, totalmente descentralizada, de una fuente, objeto u evento, sin importar la forma que tengan los modelos de sistema y observación; abarcando cualquier función de distribución de probabilidad resultante.
- Análisis del BPF en aplicaciones de localización y seguimiento con datos obtenidos de una red real (sensores reales en campo), realizando la fusión de datos fuera de línea.
- Adaptación del BPF para su implementación sobre una plataforma comercial de nodo de la red, permitiendo llegar a resultados de localización y cumpliendo con las restricciones que presenta el sistema (nodo/red); bajo consumo, poca comunicación, procesamiento simple y escaso almacenamiento de datos.
- Análisis del funcionamiento del BPF implementado en un nodo en aplicaciones de localización para estudiar el efecto de la variación de los parámetros de diseño en el desempeño del BPF.

## 1.4. Organización de la Tesis

En el presente capítulo se han explicado la motivación y el alcance de la investigación en filtros no lineales para su aplicación sobre redes de sensores. En el capítulo siguiente se presentan exhaustivamente los antecedentes de Redes de Sensores, se detallan las restricciones a tener en cuenta y las ventajas que representa el uso de una red de sensores en lugar de un único y preciso sensor. En el tercer capítulo se definen algunos parámetros básicos de la teoría de probabilidad, se expone la noción de fusión de datos y su importancia en redes de sensores. Además se desarrollan los algoritmos que dan lugar al Filtro de Partículas, dicho filtro y otras investigaciones que lo modifican con el fin de reducir los problemas que este presenta. En el Capítulo 4 se desarrolla el algoritmo de Filtro de Partículas Acotado propuesto; luego se exponen algunas de las aplicaciones implementadas del BPF; un estudio estadístico para la variación de parámetros de diseño del algoritmo y se extraen conclusiones respecto a los resultados obtenidos. En el Capítulo 5 se resume el trabajo presentado, se exponen las contribuciones del algoritmo presentado y se abre el espectro de líneas futuras de investigación.

# Capítulo 2

## Redes de Sensores

### 2.1. Introducción

Las redes de sensores son sistemas inalámbricos de bajo consumo que dispersos en gran cantidad modifican el paradigma del único sensor costoso y preciso y escasa capacidad de interpolación geográfica. Estos dispositivos son posibles hoy debido al avance en el desarrollo de circuitos integrados de bajo consumo. La perspectiva subyacente es la posibilidad de monitorear en cualquier lugar, en cualquier momento.

En [Chong and Kumar (2003)] se realiza una reseña histórica sobre WSN. Los primeros desarrollos fueron hechos para aplicaciones militares. Durante la Guerra Fría, el Sound Surveillance System (SOSUS) utilizó una red de sensores acústicos (hidrophones) en lugares estratégicos del fondo del océano para detectar y seguir submarinos Soviéticos. Con el pasar de los años se han desarrollado sensores más sofisticados para tal fin, y los SOSUS son utilizados por la Administración Oceanográfica y Atmosférica Nacional (NOAA) para monitoreo de eventos en el océano (sísmicos y de actividad animal) [Nishimura and Conlon (1994)]. También durante la Guerra fría se desarrolló y utilizó una red de radares para la defensa aérea de Estados Unidos y Canadá. El sistema evolucionó con los años incluyendo globos aerostáticos como sensores y aviones con un sistema de alerta y control (AWACS); que fueron utilizados además para detección de contrabando de drogas. Hasta el momento las redes utilizaban una estructura de procesamiento jerárquica por varios niveles hasta llegar al usuario; a veces eran operadores humanos los encargados de dicha comunicación. En 1980 con el programa en la



Agencia de Proyectos de Investigación Avanzada en Defensa (DARPA) se conocieron las Redes de Sensores Distribuidas (DSN). Para ese entonces, el Arpanet (antecesor de la Internet) llevaba varios años operativo, con cerca de 200 operadores en red en universidades e institutos de investigación. R. Kahn, co inventor del protocolo TCP/IP, clave en el desarrollo de Internet y director de la Oficina de Técnicas del Procesamiento de la Información (IPTO) en DARPA; planteó la extensión de Arpanet a redes de sensores. Hasta aquí el procesamiento era hecho mayormente en computadoras PDP-11 y VAX corriendo UNIX y VMS. Los módem operaban en 300 a 9600 Bd, y Ethernet recién se popularizaba. En 1978 en un Taller de Redes de Sensores Descentralizados (DSN) se identifican los componentes tecnológicos para DSN como sensores (acústicos), protocolos de comunicación, algoritmos y técnicas de procesamiento y software distribuido. Debido a la poca disponibilidad de componentes tecnológicos, el Taller de DSN resultante debió incluir apoyo en el direccionamiento computacional distribuido, en el procesamiento de señales, seguimiento y bancos de prueba. Continuó la evolución en aplicaciones de DSN, como el banco de prueba de tiempo real para seguimiento acústico de aviones de bajo vuelo realizado en el Laboratorio MIT de Lincoln Lacoss (1987). Los sensores eran arreglos acústicos (nueve micrófonos distribuidos en tres triángulos concéntricos con el mayor de 6 metros de lado), contaba con una computadora PDP11/34 y un arreglo de procesadores para el procesamiento de las señales acústicas. El nodo computadora para el seguimiento consistía en tres procesadores MC68000 con 256 kB de memoria y 512 kB de memoria compartida, y un sistema operativo desarrollado especialmente para tal fin. La comunicación era vía Ethernet y microondas de radio. Dicho nodo se puede ver en la Fig. 2.1.



**Figura 2.1:** Nodo del Laboratorio del MIT de 1987.

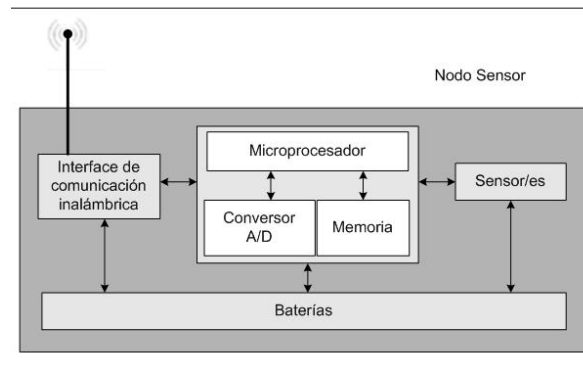
Con la aparición de microsistemas integrados por múltiples sensores, unidades de procesamiento y unidades de comunicación inalámbrica; se logra reducir errores producidos por la intervención humana. Los sensores son la interfaz con el medio físico y registran variables diversas. Los datos recolectados son procesados por un sub-circuito electrónico en el que se acondiciona, filtra y amplifica la señal; y mediante su programación se pueden realizar operaciones aritméticas, sincronización, detección de patrones, tareas de temporizado y control, almacenamiento de datos, etc. [Yick et al. (2008)] Para transmitir la información recolectada y procesada se incluye un bloque circuital para establecer comunicación inalámbrica con otro nodo o con un administrador central; y finalmente los datos receptionados en el administrador central son comunicados a una PC.

El impacto social y económico de un sistema compuesto por cientos o miles de nodos que sea capaz de establecer una red de comunicación, detectar ciertas variables de interés en una miríada de puntos, interpolar con sus vecinos la información de distinta índole o fuente para así lograr una estimación certera de la situación que se intenta medir y generar una alarma temprana es una solución que tiene un potencial muy

significativo ([Gharavi and Kumar (2003)], [Mainwaring et al. (2002a)], [Kumar et al. (1999)]). Un nodo consta de cuatro componentes principales, como se puede ver en la Fig. 2.2: el sensor, que traduce una variable física en una variable eléctrica; el procesador, que toma la señal del sensor, realiza operaciones de filtrado y amplificación para acondicionarla y la procesa utilizando un algoritmo de cálculo; la unidad de comunicación, que transmite los datos producidos por el procesador y recibe datos de otros nodos o de un administrador central; y una fuente de alimentación. Estas redes son hoy posibles en parte por el avance en el desarrollo de circuitos integrados de bajo consumo (microprocesadores, transmisores/receptores de RF, sensores) ([Green (2003)], [Kumar and Chee-Yee (2003)], [Steere et al. (2000)], [Estrin et al. (1999)], [Heinzelman et al. (2000)], [Tunhein and AS. (2002)]) que hacen que los nodos de la red no necesiten mantenimiento alguno y su costo sea muy bajo en base a grandes escalas de integración.

Un ejemplo de aplicación es el control ambiental ([Steere et al. (2000)], [Mainwaring et al. (2002a)]), sea de contaminantes en aire o en aguas. El método usual de trabajo es la colocación de uno o pocos equipos que miden en puntos específicos elegidos en base a un análisis previo. Este análisis previo no permite identificar con la dinámica necesaria otros puntos de análisis nuevos ni correlacionar esa información con posibles causas (derrames, etcétera). Mucho menos, generar alarmas tempranas de siniestros con el detalle deseable de la ubicación del evento para actuar con premura. Sin embargo una red de sensores desplegada en una zona amplia sí posee estas características deseables, aunque cada nodo individual no tenga gran precisión numérica, el sistema tiene la ventaja de la multiplicidad de fuentes y de sensores que hacen que globalmente se logren resultados superiores que con un único dispositivo.

Si bien en los comienzos las redes de sensores tenían aplicaciones militares, con el tiempo su uso se extendió a un gran número de aplicaciones civiles, ayudando a comprender mejor el mundo real ([Mainwaring et al. (2002b)], [Polastre et al. (2006)]) y sus especies, permitiendo una vida mas confortable [Haenselmann et al. (2007)], permitiendo conocer y hacer más eficientes los procesos industriales, alertando sobre futuras rupturas o deformación de estructuras [Chebrolu et al. (2008)], ayudando con



**Figura 2.2:** Nodo, sus componentes.

la seguridad en determinadas zonas (emitiendo alertas sobre desastres naturales, para protección del medio ambiente) y el monitoreo personal de la salud [Challenges et al. (2005)] entre otros.

Generalmente las redes de sensores tienen poca o ninguna infraestructura; en base a ello se las puede clasificar en redes estructuradas y redes no estructuradas. Las redes no estructuradas deben tener alta densidad de nodos sensores distribuidos aleatoriamente en el campo; en general no se accede a los nodos ni para su mantenimiento, se utilizan en aplicaciones donde se desean cubrir zonas de difícil acceso. Los nodos tienen que localizarse mediante la comunicación con sus vecinos, valiéndose de mediciones relativas hasta obtener su posicionamiento global en la red. En las redes estructuradas algunos o la totalidad de los nodos se distribuyen de una manera predeterminada en lugares fijos; en este caso se necesita un número menor de nodos; ya que la ubicación es de fácil acceso y estratégica de modo de cubrir en forma óptima la totalidad de la zona; con lo que se puede manejar la conectividad y detectar fallas. El ambiente juega un rol fundamental en la determinación del tamaño de la red, el esquema de distribución y la topología.

En esta disciplina, aunque cada nodo individual no sea certero en términos de precisión numérica, el sistema tiene la ventaja de la multiplicidad de sensores que hacen que globalmente se logren resultados superiores que con un único dispositivo. Esa capacidad de integrar la información para tomar una decisión es el proceso de fusión de

datos. Para la fusión de datos es importante contar con un modelo del sensor a utilizar que sea bueno y sencillo. Para que esta herramienta sea útil, no se puede dejar de tratar como un conjunto a los algoritmos de estimación o fusión con las restricciones de los elementos de medición [Durrant-Whyte and Stevens (2001)]. Las redes de sensores son de dominio multidisciplinario, donde se requiere una fuerte cooperación entre usuarios, expertos en el dominio de la aplicación, diseñadores de hardware y desarrolladores de software para realizar sistemas eficientes para aplicaciones específicas.

## 2.2. Paradigma Redes de Sensores

Las siguientes aplicaciones ejemplifican porque el uso de una red de sensores es superadora de una aplicación de un único sensor:

- Agricultura, en la prevención de incendios (detección de focos ígneos) y heladas, el monitoreo de temperatura en un solo nodo es útil mientras se desee monitorear ese punto y un área mínima alrededor, pero no permite saber como varia la temperatura en la región, no detecta olas de calor y/o frentes fríos, lo que permitiría la prevención de incendios u heladas. A escalas de campos y plantaciones permite optimizar el riego, la fertilización, etcétera.
- Seguridad militar, en el monitoreo de tropas enemigas, con la habilidad de detectar temperatura, luz, aceleración, sonido, y pudiendo agregar sensores de rango y ángulo como el láser se logra obtener suficiente información espacial que al ser procesada produce un esquema para detección de vehículos robusto. Esto es más difícil de lograr con un único sensor, especialmente en zonas con obstáculos.
- Detección de nubes tóxicas, al incluir en un único nodo, sensores de polvo y/o gases peligrosos en zonas aledañas a una planta química u otra similar de alta peligrosidad de contaminación, si se utiliza un único sensor se puede detectar un nivel de alarma, pero no se logra saber la dirección o velocidad de propagación del escape; cosa que con el uso de varios sensores podría conocerse, incluso se podría determinar su alcance y evitar evacuaciones inútiles.

Existen innumerables aplicaciones en redes de sensores además de las citadas; en seguridad, vigilancia perimetral y superficial, patrullaje remoto, instrumentación en hogares, hospitales, edificios, autopistas y rutas, monitoreo del medio ambiente, monitoreo de condiciones de entorno en minería, agricultura de precisión intensiva, trazabilidad y vigilancia de ganado, etc. El paradigma que se presenta es medir en cualquier lugar en cualquier momento.

Para describir el desempeño de una red de sensores se necesita la definición de varios parámetros. La mayoría de ellos están relacionados entre sí, y no todos ellos necesitan ser especificados para lograr una descripción del sistema. A continuación se describen los parámetros más importantes de un nodo sensor [Pottie and Kaiser (2004)]:

- **Leyes de propagación de señales a sensar:** Toda señal decae en amplitud rápidamente a medida que aumenta la distancia de transmisión debido a efectos de absorción y disipación en el espacio; es por ello que un sensor distante requiere operaciones costosas como deconvolución (estimación del canal e inversión) para rearmar parcialmente la disipación. También puede haber obstrucciones como paredes ó follajes densos que no permiten la detección de objetos sin importar el sensor. Es por ello que para cubrir grandes áreas, con ambientes complicados, es necesario el uso de múltiples sensores distribuidos.
- **Comunicación:** Es muy importante la distancia espacial en la construcción de redes. Cuando se utilizan antenas cerca del plano de tierra, la intensidad se pierde a razón de la cuarta potencia de la distancia debido a la cancelación parcial por rayos reflejados en el suelo. La propagación es afectada por la robustez de las superficies, la presencia de objetos reflejantes u obstructores y la elevación de la antena. Bajo ciertas condiciones de energía y ancho de banda, el número de bits que puede ser transportado exitosamente está limitado por una combinación entre las leyes de Maxwell (de propagación de radiación electromagnética) y el teorema de capacidad de Shannon (que establece relaciones fundamentales con el ancho de banda), la relación señal/ruido (SNR) y el período entre bits. Además al

estar distribuidos en el ambiente, algunos nexos entre nodos vecinos pueden llegar a estar obstruidos, pero entre otros la comunicación es limpia, evitando que los obstáculos del ambiente y los umbrales ("Shadowing") afecten el conocimiento del área en su totalidad; y mejor aún cuanto más densa es la distribución, ya que cuanto menores son las distancias entre nodos se mejora la comunicación entre ellos, se reducen los umbrales y permiten utilizar menos energía para la comunicación (se reduce el costo de recepción y retransmisión de radio entre nodos). Con el objeto de miniaturizar y reducir los costos de los nodos, elementos como las antenas están integradas a los circuitos o son de dimensión reducida.

- Consumo de energía en circuitos CMOS: La comunicación y procesamiento de datos con circuitos CMOS [Tunhein and AS. (2002)] tiene una limitada eficiencia en el consumo de energía; un transistor consume energía cada vez que es activado, evento que marca la frecuencia de conmutación; por lo tanto la energía es proporcional a ésta frecuencia, al área del transistor (su capacidad interna) y el cuadrado del salto de tensión ( $V_{activo} - V_{corto}$ ). Los componentes de alta frecuencia de switcheo y grandes amplitudes de tensión dominan el costo de energía. El límite de consumo de energía en el procesamiento de señales en tecnologías que se comunican por RF está dado por la confiabilidad requerida en la recepción de mensajes a determinada distancia. Debido a esto, la etapa amplificadora no puede reducirse; además debido a los límites de la densidad de corriente en semiconductores; ésta etapa gasta al menos cuatro veces la energía radiada, es por ello que en el tiempo domina el consumo de energía de los radios; lo que hace imprescindible que el radio esté apagado el mayor tiempo posible y transmita al nivel mínimo requerido. El procesamiento de datos también se abarata con el tiempo. Debido a que los circuitos integrados de aplicación específica (ASIC) temporizan a velocidades más bajas y usan menor precisión numérica, tienen un consumo varias veces menor de energía que los procesadores de señal digitales (DSP).

Si bien las Redes de Sensores presentan muchas ventajas, también tienen una serie de restricciones a tener muy en cuenta [Sadler (2005)]:

- **Restricción de consumo energético en comunicación:** Una de las tareas que mayor consumo requiere es la comunicación por radio frecuencia (RF); ya que se desea que sea robusta ante interferencias o choque de paquetes. Es de esperar que un nodo utilice de dos a tres veces más energía en la recepción que en la transmisión de un paquete; ya que requiere mayor funcionalidad para la adquisición y sincronización de la señal, la decodificación de los datos y demás. Es por ello que para maximizar la vida de la red es necesario minimizar las comunicaciones [Heinzelman et al. (2000)]. La selección de la arquitectura debe permitir que la red entera se entrelace, y para lograrlo se debe tener en cuenta la densidad de nodos y el área a cubrir, la frecuencia de muestreo, los requerimientos de tiempo de vida y otros factores. Además de la arquitectura se debe tener en cuenta la conectividad que existe entre nodos [Schmid and Wattenhofer (2004)], ésta es función de la ubicación de los mismos, de los canales de radio, del control de la potencia de transmisión y el tráfico de paquetes.
- **Restricción de consumo energético en componentes:** Luego de la comunicación, el consumidor de energía dominante es el reloj (cristal), es por ello que se debe elegir uno del mínimo consumo y óptima frecuencia.
- **Restricción de consumo energético por modo de operación:** El sistema en general permite a los nodos actuar en diferentes estados, como estado ocioso (idle) donde solo opera el reloj, estado de procesamiento de la señal, estado de escucha, estado de transmisión, entre otros. Las transiciones de estado no son instantáneas y puede haber un costo de energía significativo al pasar a un estado encendido (on-state); además de una demora temporal (delay).
- **Restricción de almacenamiento y procesamiento de datos:** Hay una gran variedad de tareas de procesamiento, el tema central es optimizar el desempeño de





**Figura 2.3:** Nodo comercial Mica2 de Crossbow.

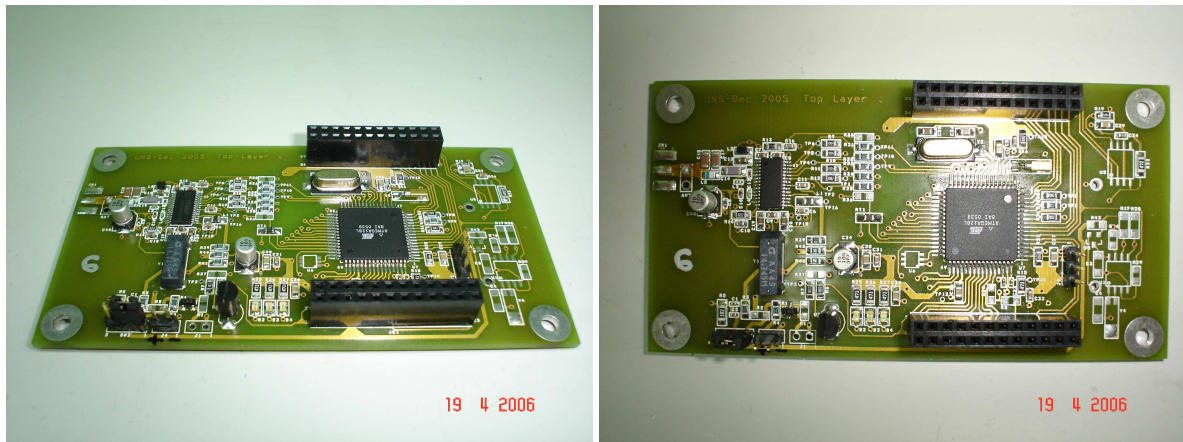
procesamiento y almacenamiento de datos minimizando la energía utilizada y explotando las oportunidades que brinda la red. Las preguntas básicas a considerar para el diseño son, cuál es la tarea de sensado y la densidad de sensores que se necesita, cuál es la frecuencia de sensado y el nivel de sincronización necesario para dichas mediciones. La idea es distribuir adecuadamente las tareas en la red.

### 2.2.1. Plataforma Utilizada en las Aplicaciones.

MICA2 es un módulo nodo desarrollado en la UC Berkeley y diseñado por Crossbow Technologies Inc. [Crossbow, 2006] y se puede ver en la Fig. 2.3.

Para el análisis y la simulación de las redes propuestas se utilizó una versión de las MICA2 producida en la Universidad Nacional del Sur, denominados MicaUNS, que se pueden ver en la Fig. 2.4.

La configuración típica del nodo contiene un procesador ATmega128L de 8 MHz, 8-bit con una memoria para instrucciones de 128 KB, 4 KB de memoria RAM, y 512 KB de memoria flash externa. Se alimenta con 2.7-3.3 V., y la radio funciona a frecuencias de uso libre (433 MHz, 915 MHz, 2.4 GHz). Los nodos son programados mediante un sistema operativo denominado TinyOS de Berkeley [Levis et al. (2005)] y el lenguaje de programación es nesC ([Brewer et al.], [Gay et al. (2003)]); orientado a objetos. Como



**Figura 2.4:** Prototipo de nodo MicaUNS.

se puede apreciar, hay restricciones de memoria que limitan el procesamiento y almacenamiento de datos. Además la comunicación es función de los protocolos utilizados y el tráfico de paquetes, lo que produciría que algunos de éstos se pierdan o lleguen corruptos a destino. Para minimizar el mantenimiento de los nodos, es preciso optimizar al máximo el consumo de energía; y teniendo en cuenta que la tarea que mayor consumo requiere es la comunicación de paquetes por RF, también se debe optimizar el intercambio de información. Al ser los nodos utilizados dispositivos inalámbricos (wireless) que se comunican por RF, pueden ser distribuidos no arbitrariamente para favorecer el proceso de comunicación y reducir el consumo de energía. A cada nodo se le adjunta un sensor ó varios que permiten obtener las medidas requeridas por el sistema.

### 2.3. Implementaciones destacadas.

Las características de una WSN particular dependen de su aplicación; aunque hay características que son comunes a la mayoría de ellas, como que monitorean fenómenos físicos con el muestreo de sensores en los nodos, que los nodos son diseminados cerca del fenómeno de interés y operan sin intervención humana; y finalmente que los nodos se comunican inalámbricamente. Las características comunes y las propias de la aplicación definen los requerimientos de hardware, software y diseño de la red. En cuanto al software de las WSN, se debe adaptar el diseño de arquitecturas y algoritmos clásico

a la aplicación.

Las aplicaciones de las WSN pueden clasificarse en dos categorías, monitoreo y localización/seguimiento. Las aplicaciones de monitoreo pueden ser en ambientes interiores o exteriores, para monitoreo de la salud y el bienestar, monitoreo de energía, monitoreo de automatizaciones y procesos en fábricas e industrias, monitoreo sísmico y estructural, entre otros.

La presente tesis se focaliza en aplicaciones de detección y localización de eventos o fuentes y su seguimiento, donde las estimaciones de los estados se basan en medidas ruidosas aportadas por los sensores y en la información obtenida a priori. Las aplicaciones de seguimiento incluyen el seguimiento de objetos, animales, humanos y vehículos. En [Yick et al. (2008)] se presentan ejemplos de aplicaciones diferentes que han sido desarrolladas y testeadas en ambientes reales.

La localización utilizando sensores acústicos (micrófonos) tiene numerosas aplicaciones. Ejemplos de esto son: la localización bajo el agua de fuentes acústicas utilizando arreglos de hidrófonos, donde se analiza la señal de sonar; la localización de la cabeza de disertantes en video conferencias e interfases multimedias con la computadora, extendiendo arreglos de micrófonos en las salas; y la localización de vehículos en campo abierto. En esta última aplicación además de utilizar el sonido de los vehículos como información de rango, es muy común utilizar radares, láser, RSSI, entre otros.

La utilización de imágenes para la localización es atractiva por las características propias de ese sensor. Es pasivo y entrega mucha información útil aunque requiere mucho procesamiento [Lee and Aghajan (2006)]. En [Arora et al.] se presenta una aplicación militar de SN para detectar intrusos e identificar entre personas, soldados y vehículos. Con un clasificador se detecta el tipo de intruso y luego se localiza dentro del área convexa, dentro de los sensores que lo detectan.

La base de la localización está en el procesamiento de la información obtenida; los sensores miden condiciones relativas al ambiente que los rodea y las transforman en señales eléctricas; al procesar las señales se revelan propiedades de objetos situados y/o eventos ocurriendo en cercanías al sensor. El sensor envía los datos, usualmente por RF,

a un nodo central o base en forma directa o mediante un centro de concentración de datos (gateway); condicionado a las restricciones de energía y ancho de banda [Sadler (2005)].

Es muy importante el manejo de la información, ya que no es posible construir un esquema de direccionamiento global para redes numerosas (todos comunicados con todos); y en general existe un gran tráfico de información redundante que proviene del hecho de que varios sensores cercanos a un fenómeno generan datos equivalentes. Dicha redundancia necesita ser explotada por protocolos de ruteo [Akkaya and Younis (2005)] para reducir el tráfico de información y mejorar el cumplimiento de restricciones de energía, ancho de banda, potencia de transmisión, capacidad de procesamiento y almacenamiento. Otra estrategia es realizar un procesamiento se presenta en [Blatt and Hero (2006)], donde se hace localización mediante un procesamiento de la información en forma distribuida. Cada nodo procesa lo mas grueso de la información con sus datos y manda solo un subconjunto de la misma al nodo central.

Sumado a lo expuesto es necesario en aplicaciones de localización y seguimiento resolver dos cuestiones que son relevantes para lograr la solución completa del problema. Ellas son la localización de cada nodo en el espacio de estados ([He et al. (2003)], [Patwari et al. (2003)]) y la sincronización de los tiempos en cada procesador ([Marioti et al. (2004)], [Sivrikaya and Yener (2004)], [Sañudo and Masson (2006b)]). Para el caso de la localización se utiliza la triangulación en caso de que se cuente con algunos nodos en la red que posean información de GPS; tambien se utiliza información de tiempo de arribo (TOA) o fuerza de la señal recibida (RSS) de los mensajes, entre otros. Para la sincronización se utiliza información de relojes externos (GPS) o algoritmos que tratan de reducir el error enviando mensajes periódicamente y utilizando la información de tiempo de los mismos; y compensando los retardos e incertidumbres conocidos de la transmisión y recepción de los mensajes.

La aplicación completa al problema de localización y seguimiento requiere de la resolución de varios aspectos relevantes. El período de muestro y de comunicación, la necesidad o no de sincronismo, la cantidad de nodos necesarios, parámetros a medir,

la distribución y seguridad de los nodos y su ubicación, calidad de los datos, entre otras. En [Raman and Chebrolu (2008)] se hace una crítica a lo expuesto sobre SN en congresos y publicaciones y se aportan detalles a tener en cuenta a la hora de presentar una aplicación o sistema. También se cita la falta de características básicas de diseño de los nodos, como que tipo de antena, de canal y de radio son los adecuados para las comunicaciones, el patrón de tráfico de paquetes, si se requiere el uso de GPS en algunos nodos, el tipo de batería necesaria, etc.

Muchas de estas características son impuestas o limitadas por las restricciones de los nodos y de la WSN. En [Sadler (2005)] se presenta un detalle de las restricciones que imponen las redes de sensores de acuerdo al tipo de sensado que se desee realizar (localización, monitoreo, seguimiento, por ejemplo) y los principios básicos para el diseño de la red; además de estrategias para reservar energía, mejorar el desempeño y vida útil del sistema.

## **2.4. Conclusiones.**

En este capítulo se ha introducido el concepto de las redes de sensores, su nacimiento, su desarrollo y los problemas que se enfrentan cuando se aplican para el seguimiento y/o localización de eventos u objetos. En el próximo capítulo se introducirá un algoritmo de estimación que es la base de el algoritmo propuesto en esta tesis para resolver el problema de localización en redes de sensores.

# Capítulo 3

## Filtro de Partículas

### 3.1. Introducción

El Filtro de Partículas es un estimador de estados de un sistema dinámico; es un método secuencial de Monte Carlo utilizado comúnmente en seguimiento de objetos. Fue propuesto en 1993 por N. Gordon, D. Salmond y A. Smith como Bootstrap Filter para implementar filtros Bayesianos recursivos. Básicamente, el Filtro de Partículas se compone de un conjunto de muestras (las partículas) y pesos asociados a cada una de esas muestras. Las partículas son estados posibles del proceso, que se pueden representar como puntos en el espacio de estados de dicho proceso.

Posee cuatro etapas principales; inicialización de las partículas, su propagación o predicción de acuerdo al modelo dinámico de los estados, la actualización con las observaciones hechas, el remuestreo de las partículas que mejor aproximan la densidad a posterior y la estimación final.

Para comprender claramente éstos conceptos se presentan algunas definiciones de términos, variables y nociones básicas de probabilidad.

Un conjunto es *discreto* si está formado por un número finito de elementos, o si sus elementos se pueden enumerar en secuencia.

Al realizar un experimento aleatorio es necesario, para su tratamiento matemático, cuantificar los resultados de modo que se asigne un número real a cada uno de los resultados posibles del experimento. De este modo se establece una relación funcional entre elementos del espacio muestral asociado al experimento y los números reales.

Formalmente se dice que una *variable aleatoria*  $X$  es una función real definida en el espacio muestral asociado a un experimento aleatorio,  $\Omega$ .

$$X : \Omega \rightarrow \mathbb{R}$$

Se llama rango de una variable aleatoria  $X$  y se denota  $\mathfrak{R}_X$ , al conjunto de los valores reales que ésta puede tomar, según la aplicación  $X$ . Dicho de otro modo, el rango de una variable aleatoria es el recorrido de la función por la que ésta queda definida:

$$\mathfrak{R}_X = \{x \in \mathbb{R} / \exists \omega \in \Omega : X(\omega) = x\}$$

Una variable aleatoria es *discreta* si su recorrido es un conjunto discreto; es el tipo de variable que se desea estimar con el filtro de Partículas.

En lo sucesivo se habla de *distribución de probabilidad*  $p(X)$  de una variable aleatoria  $X$ , lo que en teoría de probabilidad y estadística [Box et al. (1978)] se define como una función que asigna a cada suceso definido sobre la variable aleatoria la probabilidad de que dicho suceso ocurra. La distribución de probabilidad está definida sobre el conjunto de todos los eventos, formando el rango de valores de la variable aleatoria. Cuando la variable aleatoria toma valores en el conjunto de los números reales, la distribución de probabilidad está completamente especificada por la función de distribución, cuyo valor en cada real  $X$  es la probabilidad de que la variable aleatoria sea menor o igual que  $x$ , es decir,  $F_X(x) = p(X \leq x)$ .

Una característica importante de la distribución es su *media*,  $\eta = \sum_i \mathbf{x}_i / N$  para las realizaciones  $\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_N$ . El valor medio  $\eta$  también se denomina *esperanza matemática* de  $X$ , se nota  $E[X]$  y es el primer momento o centro de gravedad de la distribución de  $X$ . Define una línea de balance y da una medida de la ubicación de la distribución. Como medida de la extensión de la distribución se define la dispersión estadística *desviación estándar*,  $\sigma$  (STD, por su nombre en inglés, Standard Deviation), cuya potencia cuadrada es la *varianza*. La *varianza* se nota con  $\sigma^2$ . Una medida de cuan alejada de la media se encuentra cada observación particular de  $X$  es la desviación  $(X - \eta)$ . La *varianza* es el valor medio del cuadrado de tales desviaciones tomados para todos

los valores de  $\mathbf{x}$ :

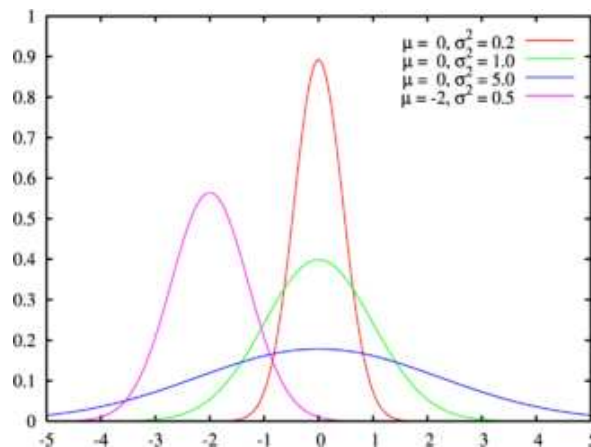
$$\sigma^2 = E[X - \eta]^2 = \frac{\sum_i (\mathbf{x}_i - \eta)^2}{N}$$

Como en lugar de tener la distribución completa se tienen muestras discretas de la misma, en ese caso se habla de *promedio y varianza de muestras* o *desviación estándar de las muestras*. En la presente tesis a menudo se habla de media y varianza, y se notan como  $\eta$  y  $\sigma^2$  respectivamente, aunque siempre se trate de muestras discretas de la distribución. Otra medida importante de una distribución es su *modo*, aunque es menos representativa que la media, salvo en las distribuciones con datos cualitativos. El *modo* de una distribución es el valor de la variable que más veces se repite en la misma; a veces aparecen distribuciones de variables con más de un modo (distribuciones bimodales, distribuciones trimodales, distribuciones multimodales, etc.).

La *Distribución Gaussiana* es una distribución de la que se hablará en lo sucesivo, en ella las muestras difieren de la media en forma simétrica, con desviaciones pequeñas y su fórmula matemática es:

$$p(\mathbf{x}) = \mathbf{c} \frac{1}{\sigma} e^{-\frac{(\mathbf{x}-\eta)^2}{2\sigma^2}}$$

donde  $\mathbf{c}$  es una constante, su forma se muestra en la Fig. 3.1. Esta distribución es importante debido a que la característica aleatoria de los errores es a menudo Gaussiana.



**Figura 3.1:** Curva de distribución Gaussiana. Ejemplos para diferentes valores de media  $\eta$  y varianza  $\sigma^2$ .



A partir de este momento se notara a las variables aleatorias con letras minúsculas y se diferenciara a las muestras de las mismas por sus subíndices  $i$ .

## 3.2. Fusión de Datos.

El objetivo de la fusión de datos es obtener una medida implícita de la incertidumbre de las observaciones para permitir que la información de los sensores sea asociada de manera consistente [Durrant-Whyte and Stevens (2001)]. Frecuentemente la probabilidad es la única manera racional de modelar incertidumbres; aunque la realidad sugiere métodos alternativos de modelado de la incertidumbre; las técnicas de modelado probabilístico juegan un papel esencial en los métodos de fusión de datos. A continuación se desarrollan algunas definiciones del modelado probabilístico. La función de densidad de probabilidad (*pdf* por su nombre en inglés, Probability Distribution Function),  $p_{\mathbf{x}}(\cdot)$  se define sobre la variable aleatoria  $\mathbf{x}$  y se nota  $p_{\mathbf{x}}(\mathbf{x})$  o  $p(\mathbf{x})$ . La *pdf* conjunta  $p_{\mathbf{xy}}(\mathbf{x}, \mathbf{y})$  se define de manera similar, e integrándola sobre la variable  $\mathbf{x}$  como muestra (3.1), da la *pdf* marginal  $p_{\mathbf{y}}(\mathbf{y})$ .

$$p_{\mathbf{y}}(\mathbf{y}) = \int_{\mathbf{x}} p_{\mathbf{xy}}(\mathbf{x}, \mathbf{y}) d\mathbf{x} \quad (3.1)$$

En igual forma, integrando sobre  $\mathbf{y}$  se obtiene  $p_{\mathbf{x}}(\mathbf{x})$ . La *pdf* condicional  $p(\mathbf{x}|\mathbf{y})$  se define como en (3.2) y tiene las propiedades de una *pdf* con  $\mathbf{x}$  la variable dependiente dado  $\mathbf{y}$  que toma un valor específico.

$$p(\mathbf{x}|\mathbf{y}) = \frac{p(\mathbf{x}, \mathbf{y})}{p(\mathbf{y})} \quad (3.2)$$

La regla de la cadena de densidad condicional puede ser utilizada para expandir *pdf* conjuntas en términos de densidades condicionales y marginales, despejando (3.2) se obtiene (3.3).

$$p(\mathbf{x}, \mathbf{y}) = p(\mathbf{x}|\mathbf{y}) p(\mathbf{y}) \quad (3.3)$$

Reemplazando (3.3) en (3.1) se logra (3.4).

$$p_{\mathbf{y}}(\mathbf{y}) = \int_{\mathbf{x}} p_{\mathbf{x}/\mathbf{y}}(\mathbf{y}|\mathbf{x}) p_{\mathbf{x}}(\mathbf{x}) d\mathbf{x} \quad (3.4)$$

La ecuación (3.4) se conoce como el Teorema de Probabilidad Total e indica que la probabilidad total de un estado  $\mathbf{y}$  puede ser obtenida considerando las maneras de las cuales  $\mathbf{y}$  puede ocurrir dado que  $\mathbf{x}$  toma un valor específico ( $p_{\mathbf{xy}}(\mathbf{y}|\mathbf{x})$ ) pesado por la probabilidad de que cada uno de los valores de  $\mathbf{x}$  sea instanciado ( $p_{\mathbf{x}}(\mathbf{x})$ ).

### 3.2.1. Fusión de Datos en Redes de Sensores.

En el Capítulo 2 se señaló que para ciertas aplicaciones es una opción superadora utilizar múltiples nodos, pero para diseñar una WSN se debe realizar un estudio previo del ambiente de interés con el fin de lograr una correcta ubicación de los nodos; un estudio del sistema y una recolección de datos necesarios para determinar los sensores adecuados al caso. Una vez que se tiene la red de sensores en funcionamiento y se obtiene la información de los nodos, la fusión de datos combina en forma consistente y óptima toda esta información. Es importante describir precisamente la manera en la cual la información se relaciona con el estado subyacente del mundo. La información será de diferentes fuentes, imperfecta e incluso incompleta; pero por otro lado puede disponerse de algunas certezas sobre los estados o pueden existir conocimientos acerca de ciertas restricciones y relaciones en o entre los estados.

La combinación de observaciones de diferentes sensores pretende potenciar las virtudes de cada sensor y minimizar sus desventajas, con el fin de realizar inferencias sobre el mundo exterior. Se pretende obtener un mejor resultado a partir de múltiples sensores realizando inferencias que pueden no ser posibles a partir de uno solo.

Existen diversas maneras de realizar la fusión de datos, dependiendo de la aplicación y la distribución de la red. Se puede fusionar entre nodos vecinos o bien realizar un procesamiento previo de la información en el nodo y luego transmitir los resultados a un nodo de mayor jerarquía o bien a un nodo central de procesamiento [Blatt and Hero (2006)].

En general en localización y seguimiento el procesamiento se realiza en línea (online), al instante, sobre todo en aplicaciones militares y de seguridad; y requieren algoritmos rápidos y convergentes. En el caso de monitoreo de ambientes o especies puede

interesar tener un gran volumen de datos almacenados pero el procesamiento puede ser realizado fuera de línea (offline) en el momento que sea necesario.

En la presente tesis se presenta un algoritmo de fusión de datos en línea con bajo requerimiento de cálculo, comunicación y almacenamiento; aplicable en problemas que requieren una fusión sobre un esquema distribuido; es decir, la fusión se realiza nodo a nodo y se comunica directamente un resultado (WSN descentralizada).

### 3.2.2. Trabajos relacionados

Un asunto importante a tener en cuenta para la fusión de datos en redes de sensores es la elección del nodo líder (así se llama al nodo que realiza la fusión con los datos de los sensores vecinos) con el objeto de reducir las comunicaciones, tema que se aborda en [Ihler et al. (2005)]. Este nodo es además el encargado de seleccionar al nuevo líder para el instante posterior (pudiendo elegirse a si mismo) basándose en la información recibida. Con esta estrategia se logra mantener las comunicaciones limitadas a un grupo de sensores de la red, el más cercano a la fuente que se está siguiendo. El trabajo también presenta una forma de medir el error entre la distribución verdadera y la estimada al introducir una medición actual y el efecto que esta tendrá en las estimaciones posteriores que afecta la elección del líder.

En [Kaplan (2006)] se seleccionan los mejores nodos, en el sentido del error de posición en media cuadrática, para localizar un objetivo; y en [Wu et al. (2006)] se propone un método de despliegue de sensores centralizado que maximiza el cubrimiento de un área con obstáculos. En [Subramaniam et al. (2006)] y [Chen et al. (2006)] se presentan métodos para detectar las malas mediciones con poco requerimiento de memoria y en tiempo real.

## 3.3. Filtrado Recursivo de Bayes.

Uno de los problemas mas usuales en el procesamiento de señales es la estimación del estado de un sistema que evoluciona, utilizando mediciones estocásticamente relacionadas a él. Se define el estado del sistema como  $\mathbf{x}_k$  en el tiempo  $t = k\Delta t$  donde

$\Delta t$  es el intervalo de tiempo entre muestras. Se define a la medición u observación del sistema como  $\mathbf{z}_k$  en el mismo instante de tiempo. A  $\mathbf{x}_k$  y  $\mathbf{z}_k$  se las considera variables aleatorias debido al ruido y la incertidumbre en la evolución del proceso. Las secuencias de estados y mediciones en  $t = \Delta t, 2\Delta t, \dots, k\Delta t$  son  $\mathbf{x}_{1:k} \triangleq \{\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_k\}$  y  $\mathbf{z}_{1:k} \triangleq \{\mathbf{z}_1, \mathbf{z}_2, \dots, \mathbf{z}_k\}$ . Ahora se puede ser mas preciso en cuanto al problema de estimación y se puede decir que se desea estimar el estado  $\mathbf{x}_k$  utilizando las mediciones obtenidas hasta el momento,  $\mathbf{z}_{1:k}$ . Debido a que  $\mathbf{x}_k$  es una variable aleatoria, toda la información que aportan las mediciones está contenida en la densidad posterior  $p(\mathbf{x}_k | \mathbf{z}_{1:k})$  y con el Filtrado Bayesiano se genera dicho estimado para cada  $k = 1, 2, \dots$ . Es decir, en lugar de estimar el estado, se estima una densidad de probabilidad del mismo, permitiendo producir cualquier estadística de  $\mathbf{x}_k$ . Se puede resumir el proceso de estimación en dos pasos, en el primero, de predicción, se mapea la densidad posterior previa  $p(\mathbf{x}_{k-1} | \mathbf{z}_{1:k-1})$  en la densidad de predicción de un paso  $p(\mathbf{x}_k | \mathbf{z}_{1:k-1})$ ; en el segundo paso se realiza una actualización de la medición, con ella y el resultado del primer paso se produce la densidad posterior deseada,  $p(\mathbf{x}_k | \mathbf{z}_{1:k})$ .

Se define la predicción Bayesiana con (3.5) y la actualización del estado con una nueva medición con (3.6).

$$p(\mathbf{x}_k | \mathbf{z}_{1:k-1}) = \int_{\mathbf{x}_{k-1}} p(\mathbf{x}_k | \mathbf{x}_{k-1}, \mathbf{z}_{1:k-1}) p(\mathbf{x}_{k-1} | \mathbf{z}_{1:k-1}) d\mathbf{x}_{k-1} \quad (3.5)$$

$$p(\mathbf{x}_k | \mathbf{z}_{1:k}) = \frac{p(\mathbf{z}_k | \mathbf{x}_k, \mathbf{z}_{1:k}) p(\mathbf{x}_k | \mathbf{z}_{1:k-1})}{p(\mathbf{z}_k | \mathbf{z}_{1:k-1})} \quad (3.6)$$

Es complejo realizar el filtrado Bayesiano recursivamente debido a la presencia de los términos  $p(\mathbf{x}_k | \mathbf{x}_{k-1}, \mathbf{z}_{1:k-1})$  en (3.5) y  $p(\mathbf{z}_k | \mathbf{x}_k, \mathbf{z}_{1:k})$  en (3.6) que requieren el almacenamiento de todas las mediciones anteriores  $\mathbf{z}_{1:k-1}$ . Para lograr la recursividad del filtrado se deben imponer restricciones al proceso estocástico que permiten eliminar las observaciones anteriores.

Lo primero que se supone es que la secuencia de estados es Markov de primer orden, es decir que  $p(\mathbf{x}_k | \mathbf{x}_{0:k-1}) = p(\mathbf{x}_k | \mathbf{x}_{k-1})$ . Conceptualmente, para un proceso Markov, el valor de la variable aleatoria  $\mathbf{x}_{k-1}$  provee tanta información sobre  $\mathbf{x}_k$  como el valor

del proceso en  $t = (k-1)\Delta t$  y todos los instantes anteriores. En este sentido, la noción típica de “estado” se ve a partir de la definición de Markov, el estado del sistema  $\mathbf{x}_k$  es cualquier colección de variables que hagan  $\mathbf{x}_{0:k-1}$  no consecuente para la estimación de valores futuros de la secuencia. En segundo lugar se supone que, condicionado a  $\mathbf{x}_k$ ,  $\mathbf{z}_k$  es independiente del resto de la secuencia de estados y del resto de las mediciones; esto se puede expresar como:

$$p(\mathbf{z}_{1:k} | \mathbf{x}_{1:k}) = \prod_{j=1}^k p(\mathbf{z}_j | \mathbf{x}_j) \quad (3.7)$$

Cuando  $p(\mathbf{x}_k | \mathbf{z}_{1:k})$  no se puede resolver analíticamente, una alternativa es utilizar métodos de Monte Carlo [Carpenter et al. (1999)] para la aproximación e integración de la función de probabilidad mediante un gran conjunto de muestras.

Con las suposiciones anteriores se prueban los lemas de filtrado de Bayes Recursivo, el resultado para la predicción Bayesiana recursiva se muestra en (3.8) y para la actualización del estado recursiva, con una nueva medición, se muestra en (3.9).

$$p(\mathbf{x}_k | \mathbf{z}_{1:k-1}) = \int_{\mathbf{x}_{k-1}} p(\mathbf{x}_k | \mathbf{x}_{k-1}) p(\mathbf{x}_{k-1} | \mathbf{z}_{1:k-1}) d\mathbf{x}_{k-1} \quad (3.8)$$

$$p(\mathbf{x}_k | \mathbf{z}_{1:k}) = \frac{p(\mathbf{z}_k | \mathbf{x}_k) p(\mathbf{x}_k | \mathbf{z}_{1:k-1})}{p(\mathbf{z}_k | \mathbf{z}_{1:k-1})} \quad (3.9)$$

### 3.3.1. Algoritmo Bayesiano Recursivo Exacto

Se considera el modelo de un sistema estocástico (que basa su resultado en probabilidades que cambian en el tiempo) discreto donde la evolución del estado  $\mathbf{x}_k$  está dada por

$$\mathbf{x}_k = f_k(\mathbf{x}_{k-1}, \mathbf{u}_k) \quad (3.10)$$

donde  $\mathbf{x}_k$  es el vector de estados de  $n_x \times 1$  y  $\mathbf{u}_k$  es el vector de ruido del sistema de  $n_u \times 1$ . Cada  $f_k$  mapea  $\mathbb{R}^{n_x} \times \mathbb{R}^{n_u}$  en  $\mathbb{R}^{n_x}$ .

Y se considera el modelo de observación (3.11), donde  $\mathbf{z}_k$  es el vector de observación de  $n_z \times 1$  y  $\mathbf{v}_k$  es el vector de ruido de medición de  $n_v \times 1$  que modela perturbaciones

aleatorias e incertidumbres del modelo. Además  $h_k$  es la función de observación variante en el tiempo y mapea  $\mathbb{R}^{n_x} \times \mathbb{R}^{n_v}$  en  $\mathbb{R}^{n_z}$ .

$$\mathbf{z}_k = h_k(\mathbf{x}_k, \mathbf{v}_k) \quad (3.11)$$

donde  $\mathbf{z}_k$  es el vector de observación de  $n_z \times 1$  y  $\mathbf{v}_k$  es el vector de ruido de medición de  $n_v \times 1$  que modela perturbaciones aleatorias e incertidumbres del modelo de observación. Además  $h_k$  es la función de observación variante en el tiempo y mapea  $\mathbb{R}^{n_x} \times \mathbb{R}^{n_v}$  en  $\mathbb{R}^{n_z}$ . Afortunadamente se trabaja con una versión de  $\mathbf{x}_k$  discreta y finita, por lo tanto las integrales (3.8) y (3.9) se convierten en sumas finitas que se pueden calcular fácilmente, mas específicamente la ecuación (3.8) de predicción puede calcularse como sigue, para  $i = 1, 2, \dots, n_x$ .

$$Pr(\mathbf{x}_k = i | \mathbf{z}_{1:k-1}) = \sum_{j=1}^N Pr(\mathbf{x}_k = i | \mathbf{x}_{k-1} = j) Pr(\mathbf{x}_{k-1} = j | \mathbf{z}_{1:k-1}) \quad (3.12)$$

El término  $Pr(\mathbf{x}_k = i | \mathbf{x}_{k-1} = j)$  es un elemento de la matriz de transición de Markov. La ecuación (3.9) de actualización puede calcularse con:

$$Pr(\mathbf{x}_k = i | \mathbf{z}_{1:k}) = \frac{p(\mathbf{z}_k | \mathbf{x}_k = i) Pr(\mathbf{x}_k = i | \mathbf{z}_{1:k-1})}{\sum_{j=1}^N p(\mathbf{z}_k | \mathbf{x}_k = j) Pr(\mathbf{x}_k = j | \mathbf{z}_{1:k-1})} \quad (3.13)$$

Luego la recursión de (3.12) y (3.13) requiere solo la evaluación de las probabilidades  $Pr(\mathbf{x}_k = i | \mathbf{x}_{k-1} = j)$  y la función de probabilidad  $p(\mathbf{z}_k | \mathbf{x}_k = i)$ . Los algoritmos basados en dicha recursividad forman la base para la solución Bayesiana óptima. La propagación recursiva de la densidad encuentra su forma analítica en un conjunto restrictivo de casos; incluyendo el Filtro de Kalman (KF) ([Kalman and Bucy (1961)], [Welch and Bishop (1995a)]) y los Filtros Basados en Grillas [Rabiner and Juang (1986)].

Además, cuando la solución analítica es intratable, hay filtros que permiten aproximar dicha solución, son los Filtros de Kalman Extendidos (EKF) ([Ljung (1979)], [Wan and Van Der Merwe (2000)]), los Filtros Basados en Grillas Aproximados y los Filtros de Partículas (PF) [Gordon et al. (2002a)].

En [Bhowmik and Roy (2007)] se realiza una comparación de la técnica de estimación del KF y los Filtros Basados en Grillas, para sistemas lineales y no lineales.

Si las funciones  $f_k$  y  $h_k$  son lineales y la distribución de los ruidos es Gaussiana, se puede obtener un estimado óptimo recursivo de  $\mathbf{x}_k$  utilizando un KF; la distribución posterior será Gaussiana y la recursión del KF permite una actualización eficiente de la media y la varianza de la distribución con cada observación recibida. Si las funciones  $f_k$  y  $h_k$  son no lineales, el EKF y las extensiones (modificaciones, adaptaciones) del mismo son aplicables para obtener la estimación del estado. Sin embargo, cuando las funciones de proceso y observación son no lineales y además la naturaleza de los ruidos e incertidumbres no permiten representar la pdf como Gaussiana, las condiciones para la aplicación del Filtro de Kalman no son satisfechas y los resultados que se obtienen no son por lo general correctos. El PF funciona correctamente bajo dichas condiciones y provee un estimado correcto, convergente y consistente, de  $\mathbf{x}_k$ .

Como se ha presentado, el enfoque Bayesiano es construir la pdf del estado basada en toda la información disponible. El objetivo del PF es representar la pdf requerida con un conjunto de muestras aleatorias (partículas), en lugar de hacerlo en forma cerrada, sobre el espacio de estados. Las técnicas de muestreo evitan tener que definir una grilla en el espacio de estados; y como resultado de la estimación las muestras se concentran naturalmente en regiones de alta densidad de probabilidad. Esto permite tratar cualquier no linealidad funcional y ruidos de medición o modelado del sistema con cualquier distribución.

### 3.4. Métodos Secuenciales de Monte Carlo

Los filtros estimadores utilizan diferentes métodos para la aproximación de la densidad posterior [Chen (2003)]. El método de aproximación Gaussiana/ Laplaciana es el mas simple, ya que la aproximación es analíticamente manejable, suponiendo que la densidad posterior es Gaussiana, algo que en la aplicación de interés no se cumple. Otro método es el Multi-grilla y de Aproximación por Puntos Másicos; en este caso la grilla debe ser suficientemente densa para obtener una buena aproximación de la pdf

posterior; la desventaja es que el espacio de estados no se puede particionar de forma no uniforme para dar mayor resolución a un estado de gran densidad. Además, para densidades multi-modo se sugiere una grilla por modo y luego la aproximación por puntos másicos, que no es trivial y tiene alta complejidad; lo que la hace inadecuada para su implementación en un nodo.

Otro método de aproximación es la Aproximación por Suma de Gaussianas, motivada por la observación de que toda distribución no Gaussiana puede aproximarse, con algún grado de precisión, con una suma de un número suficiente de Gaussianas que admita una solución manejable para el cálculo del primer y segundo momento de cada una de ellas. El Filtro de Gaussianas ([Alspach and Sorenson (1971)], [Alspach and Sorenson (1972)]) utiliza esta idea y corre un banco de EKF en paralelo para obtener un estimado, es decir, para estimar el primer y segundo momento de cada Gaussiana de la sumatoria. La obtención de la representación en cada iteración, es decir recalcular las gaussianas que representan la probabilidad a posterior (remuestreo), es la etapa que insume mayor procesamiento, lo que dificulta su programación y el cumplimiento de los requerimientos de bajo procesamiento de un nodo. Por lo expuesto los Métodos Secuenciales de Monte Carlo, que se presentan a continuación, son los elegidos para la representación de la pdf del estado.

Los métodos secuenciales de Monte Carlo (SMC por su nombre en inglés, Sequential Monte Carlo) [Liu and Chen (1998)] permiten la inferencia de la distribución posterior en modelos de proceso y observación que pueden ser no lineales y no Gaussianos. Los métodos SMC son escalables y la precisión de los estimados depende del número de partículas utilizadas para la aproximación de la medición.

En lo sucesivo con un superíndice  $i$  se nota el número de muestra y con un subíndice  $1 : k$  los instantes de tiempo en que se toman las muestras. Para mayor claridad en la presentación del algoritmo, en principio se abstrae el término secuencial, aunque en la presentación final se volverá a explicitar.

En los métodos SMC se estiman las propiedades de una distribución arbitraria  $p(\mathbf{x}^i, \mathbf{z})$  utilizando muestras aleatorias de la distribución. El número de muestras debe



ser suficiente para cubrir el espacio dinámico de las variables de interés.

Cuando no se conoce exactamente la distribución  $p(\mathbf{x}^i, \mathbf{z})$ , se puede muestrear otra distribución conocida  $q(\mathbf{x}^i, \mathbf{z})$  denominada Distribución de Importancia. Se toman entonces las muestras  $\mathbf{x} = \{\mathbf{x}^i, i = 1, \dots, N_s\}$  de la distribución  $q(\mathbf{x}^i, \mathbf{z})$  sabiendo que se cumple que  $p(\mathbf{x}^i, \mathbf{z})$  es proporcional a  $q(\mathbf{x}^i, \mathbf{z})$ .

Luego se hace una corrección asignando un peso positivo a cada muestra, el valor de los pesos es proporcional a la relación (3.14), denominada Función de Importancia.

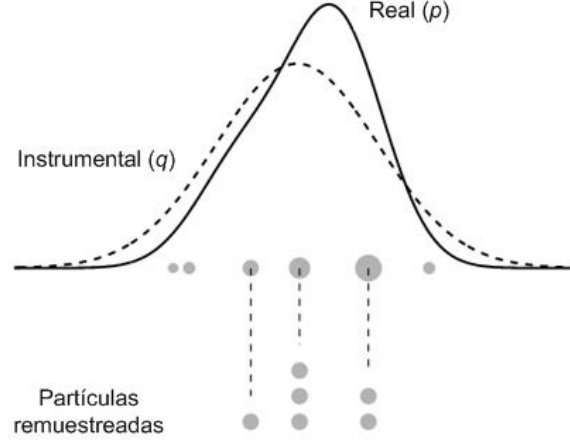
$$w_i \propto \frac{p(\mathbf{x}^i, \mathbf{z})}{q(\mathbf{x}^i, \mathbf{z})} \quad (3.14)$$

En general cuando se aplica Muestreo por Importancia [Arulampalam et al. (2002)] a modelos de proceso o a inferencia Bayesiana, la relación es conocida solo hasta un factor de escala, una constante. Aunque el Muestreo por Importancia principalmente se utiliza para lograr aproximar  $p(\mathbf{x}^i, \mathbf{z})$ , puede ser utilizado directamente para muestrear la distribución  $p(\mathbf{x}^i, \mathbf{z}^i)$ ; esto último se logra con el Remuestreo por Importancia.

El Remuestreo por Importancia es seguido por un paso adicional de muestreo aleatorio que es un procedimiento de dos etapas. En la primer etapa (etapa de muestreo por importancia) se toma una muestra  $\tilde{\mathbf{x}} = \{\tilde{x}^j, j = 1, \dots, N_s\}$  de la distribución de importancia  $q(\mathbf{x}^i, \mathbf{z})$  y se calcula la versión normalizada de los pesos. En la segunda se toma una muestra de tamaño  $N_s$ ,  $\mathbf{x} = \{x^j, j = 1, \dots, N_s\}$  del conjunto intermedio de los puntos  $\tilde{\mathbf{x}}$ ; teniendo en cuenta sus pesos. Las muestras mas probables son replicadas en proporción a sus pesos como se muestra en la Fig. 3.2.

Cuando el método se aplica secuencialmente se habla de Muestreo Secuencial por Importancia (SIS por su nombre en inglés, Sequential Importance Sampling) ([Gordon et al. (2002a)], [MacEachern et al. (1999)]) y de Remuestreo Secuencial por Importancia (SIR por su nombre en inglés, Sequential Importance Resampling). El SIR también es conocido como Bootstrap Filter ó Condensation Algorithm.

Aquí  $\mathbf{x}_{0:k} = \{\mathbf{x}_j, j = 1, \dots, k\}$  es el conjunto de todos los estados hasta el instante  $k$  donde  $\{\mathbf{x}_{0:k}^i, i = 1, \dots, N_s\}$  es un conjunto de muestras de los estados y  $\{w_k^i, i = 1, \dots, N_s\}$  los pesos respectivos; así se tiene un conjunto de muestras aleatorias (posibles estados



**Figura 3.2:** Remuestreo de partículas. Arriba: La muestra evaluada de distribución  $q$  (línea de puntos) con los pesos de importancia normalizados asociados; representados por círculos de tamaño proporcional a dichos pesos. La densidad de probabilidad verdadera,  $p$ , se representa en línea completa. Abajo: Luego del remuestreo, todas las partículas tienen igual peso de importancia y algunas de ellas han sido duplicadas o descartadas.

del proceso) asociadas con sus pesos  $\{\mathbf{x}_{0:k}^i, w_k^i\}_{i=1}^{N_s}$  que caracterizan la pdf posterior  $p(\mathbf{x}_{0:k}, \mathbf{z}_{1:k})$ .

Los pesos son normalizados de manera que  $\sum_{i=1}^{N_s} w_k^i = 1$ . Cuando el algoritmo se extiende a secuencial, se fuerza la restricción de que las muestras son independientes temporalmente, la distribución propuesta puede evaluarse para cada  $k$  con (3.15).

$$q(\mathbf{x}_{0:k}, \mathbf{z}_{1:k}) = q(\mathbf{x}_0) \prod_{j=1}^k q(\mathbf{x}_j | \mathbf{x}_{j-1}, \mathbf{z}_j) \quad (3.15)$$

Además, sean  $\mathbf{x}_{0:k}^i \sim q(\mathbf{x}_{0:k}, \mathbf{z}_{1:k})$  para  $i = 1, \dots, N_s$  las muestras generadas por una Densidad de Importancia  $q(\mathbf{x}_{0:k}, \mathbf{z}_{1:k})$ , la probabilidad a posteriori se puede evaluar con (3.16).

$$p(\mathbf{x}_{0:k}, \mathbf{z}_{1:k}) \approx \sum_{i=1}^{N_s} w_k^i \delta(\mathbf{x}_{0:k} - \mathbf{x}_{0:k}^i) \quad (3.16)$$

Donde los pesos están normalizados y  $\delta(\cdot)$  es la función Delta Dirac. Puede demostrarse que cuando  $N_s \rightarrow \infty$  la aproximación (3.16) se acerca a la verdadera densidad posterior.

A diferencia de algoritmos de estimación donde la pdf es conocida, en el PF no se hacen suposiciones respecto a ella. Por ejemplo la distribución  $p(\mathbf{x}_k | \mathbf{z}_{1:k})$  en el KF es una Gaussiana multivariada debido a la linealidad y ruido Gaussiano que se supone en los modelos de proceso y observación, sin embargo en el PF solo se requiere evaluar las distribuciones y crear muestras de ellas.

El peso normalizado de la partícula  $i$ -ésima se puede calcular con (3.17).

$$w_k^i \propto \frac{p(\mathbf{x}_{0:k}^i, \mathbf{z}_{1:k})}{q(\mathbf{x}_{0:k}^i, \mathbf{z}_{1:k})} \quad (3.17)$$

Los pesos no normalizados se evalúan recursivamente con (3.18) debido a que  $q(\mathbf{x}_k^i | \mathbf{x}_{k-1}^i, \mathbf{z}_k)$  ya no está condicionado a  $\mathbf{z}_{1:k-1}$  por (3.15) y además  $q(\mathbf{x}_k | \mathbf{x}_{0:k-1}, \mathbf{z}_{1:k}) = q(\mathbf{x}_k | \mathbf{x}_{k-1}, \mathbf{z}_k)$ , ya que solo se necesita una estimación filtrada de  $p(\mathbf{x}_k, \mathbf{z}_{1:k})$  a cada instante.

$$w_k^i = w_{k-1}^i \frac{p(\mathbf{z}_k | \mathbf{x}_k^i) p(\mathbf{x}_k^i | \mathbf{x}_{k-1}^i)}{q(\mathbf{x}_k^i | \mathbf{x}_{k-1}^i, \mathbf{z}_k)} \quad (3.18)$$

En Alg. 1 se resume la descripción en pseudo-código del algoritmo SIS.

---

**Algoritmo 1** Algoritmo de Muestreo Secuencial por Importancia.

---

Inicialización

**for**  $i = 1$  to  $N_s$  **do**

$\tilde{\mathbf{x}}_0^i \approx q(\mathbf{x}_0 | \mathbf{z}_0)$

Asignación inicial de los pesos de importancia

$$\tilde{w}_0^i = \frac{p(\mathbf{z}_0 | \tilde{\mathbf{x}}_0^i) p(\tilde{\mathbf{x}}_0^i)}{q(\tilde{\mathbf{x}}_0^i | \mathbf{z}_0)}$$

**end for**

**for**  $k = 1$  to  $T$  **do**

**for**  $i = 1$  to  $N_s$  **do**

Propagación de partículas:  $\tilde{\mathbf{x}}_k^i \approx q(\tilde{\mathbf{x}}_k^i | \tilde{\mathbf{x}}_{k-1}^i, \mathbf{z}_k)$

Cálculo de pesos  $\tilde{w}_k^i = w_{k-1}^i \frac{p(\mathbf{z}_k | \tilde{\mathbf{x}}_k^i) p(\tilde{\mathbf{x}}_k^i | \tilde{\mathbf{x}}_{k-1}^i)}{q(\tilde{\mathbf{x}}_k^i | \tilde{\mathbf{x}}_{k-1}^i, \mathbf{z}_k)}$

**end for**

Normalización de los pesos  $w_k^i = \frac{\tilde{w}_k^i}{\sum_{i=1}^{N_s} \tilde{w}_k^i}$  para  $i = 1, \dots, N_s$

Cálculo del estimado:  $p(\mathbf{x}_k, \mathbf{z}_k) = \sum_{i=1}^{N_s} w_k^i p(\tilde{\mathbf{x}}_k^i | \tilde{\mathbf{x}}_{k-1}^i, \mathbf{z}_k)$

**end for**

---

El propósito del algoritmo es propagar y actualizar las muestras recursivamente

para obtener un conjunto de valores del espacio de estados que estarán aproximadamente distribuidos como la función de probabilidad verdadera. Dado que el método es adaptivo dará como resultado una mayor densidad de partículas donde la probabilidad es mayor.

Al ser eliminadas las partículas de pesos pequeños y replicarse las de peso elevado, surge el problema del *empobrecimiento de muestras*. Este empobrecimiento se manifiesta en que luego de varias iteraciones solo una pequeña porción de las partículas contienen toda la probabilidad (i.e. pesos muy grandes) significando la falla a largo plazo del SIS. Sin embargo, es posible remuestrear las partículas y evitar en el corto plazo la acumulación del error y convertir a la aproximación de la distribución en mas estable. En esto consiste el SIR, aunque su eficiencia no es buena debido a que se debe realizar la elección de la densidad de importancia en el paso de predicción, sin incorporar información de la última medición disponible  $\mathbf{z}_k$ . La elección de la distribución de importancia propuesta en el SIS y el SIR juega un rol crucial en su desempeño final ([Handschin (1970)], [Zaritskii et al. (1976)], [Kong et al. (1994)]).

Tanto el filtro SIS como el SIR utilizan muestreo por importancia, la diferencia entre ellos es que en el SIR siempre se realiza la etapa de remuestreo, mientras que en el SIS se calculan los pesos secuencialmente y el remuestreo se realiza solo cuando es necesario. El remuestreo apunta a eliminar las muestras con peso bajo y replicar las de peso elevado; las partículas actualizadas con sus pesos asociados se reemplazan por nuevas partículas con pesos equivalentes ( $w_k^i = \frac{1}{N_s}$ ).

Se debe resaltar que el remuestreo tiene efectos secundarios en el desempeño del algoritmo. En primer lugar limita la posibilidad de paralelizar el proceso de estimación dado que para efectuarlo se requiere tener el total de las partículas calculadas con sus pesos. En segundo lugar las partículas de peso elevado son seleccionadas muchas veces estadísticamente, lo que provoca una pérdida de su diversidad, y en casos donde el proceso tiene ruidos pequeños todas las partículas colapsan en un solo punto luego de varias iteraciones. Se ha trabajado mucho en esquemas para contrarrestar este efecto; una de las estrategias de remuestreo mas conocida es el Remuestreo Sistemático

[Arulampalam et al. (2002)].

El algoritmo de Filtro de Partículas estándar se puede ver en Alg. 2, con una función general propuesta y remuestreo opcional en cada paso. Hay variantes del algoritmo que propagan más partículas que las seleccionadas y que tienen un número variable de estas en cada paso temporal. Se debe notar que cuando el remuestreo (la selección) ocurre, la estimación debe llevarse a cabo utilizando las partículas pesadas (las de la etapa previa al remuestreo). El Filtro Bootstrap [Gordon et al. (2002a)] utiliza la densidad de transición de los estados  $p(\mathbf{x}_k^i | \mathbf{x}_{k-1}^i)$ , como distribución de importancia; luego los pesos de importancia se simplifican a la ecuación (3.19).

$$w_k^i = w_{k-1}^i p(\mathbf{z}_k | \mathbf{x}_k^i) \quad (3.19)$$

Una característica distintiva del Filtro Bootstrap es que el peso incremental no depende en forma directa de la trayectoria pasada de las partículas (aunque esa información está contenida implícitamente en  $w_{k-1}^i$ ), sino solo de la función de verosimilitud.

En esquemas más sofisticados, el remuestreo solo se realiza cuando la distribución de los pesos se degenera, esto puede medirse monitoreando los cambios en la entropía del patrón de los pesos con el tiempo [Liu and Chen (1998)].

A modo de resumen se detallan las etapas del PF, que comienza por la etapa de Propagación ó Predicción que traslada las partículas resultantes de la estimación en el instante anterior con el modelo de proceso. La siguiente etapa es la de Actualización, en la que, una vez que se obtiene una observación, los pesos son actualizados siguiendo la distribución de probabilidad del proceso. Una vez realizada esta etapa, el conjunto de partículas con sus pesos respectivos se pueden utilizar para formar el estimado del estado, es en la etapa de Estimación. En este punto se puede lograr una estimación a posteriori utilizando las partículas y sus pesos respectivos. Si se implementa una etapa de Remuestreo, esta se realiza antes de la estimación para obtener un conjunto de partículas concentradas en la región del espacio de mayor probabilidad. Finalmente se aplica una etapa de Remuestreo que permite contener el conjunto de partículas mayormente concentrado en la región del espacio de estados de interés.

---

**Algoritmo 2** Algoritmo Filtro de Partículas.

---

- Inicialización.
    - for**  $i = 1$  to  $N_s$  **do**
    - $\tilde{\mathbf{x}}_0^i \approx q(\mathbf{x}_0 | \mathbf{z}_0)$
    - Asignación inicial de los pesos de importancia
    - $$\tilde{w}_0^i = \frac{p(\mathbf{z}_0 | \tilde{\mathbf{x}}_0^i) p(\tilde{\mathbf{x}}_0^i)}{q(\tilde{\mathbf{x}}_0^i | \mathbf{z}_0)}$$
    - end for**
  - Procedimiento secuencial
    - for**  $k = 1$  to  $T$  **do**
    - for**  $i = 1$  to  $N_s$  **do**
    - Propagación de partículas:  $\tilde{\mathbf{x}}_k^i \approx q(\tilde{\mathbf{x}}_k^i | \mathbf{x}_{k-1}^i, \mathbf{z}_k)$
    - Actualización de pesos 
$$\tilde{w}_k^i = w_{k-1}^i \frac{p(\mathbf{z}_k | \tilde{\mathbf{x}}_k^i) p(\tilde{\mathbf{x}}_k^i | \mathbf{x}_{k-1}^i)}{q(\tilde{\mathbf{x}}_k^i | \mathbf{x}_{k-1}^i, \mathbf{z}_k)}$$
    - end for**
    - Normalización de los pesos  $w_k^i = \frac{\tilde{w}_k^i}{\sum_{i=1}^{N_s} \tilde{w}_k^i}$  para  $i = 1, \dots, N_s$
    - if** Remuestreo **then**
    - Seleccionar  $N_s$  índices de partículas  $j_i \in 1, \dots, N_s$  de acuerdo con los pesos  $\{w_{k-1}^j\}_{j=1}^{N_s}$ .
    - Asignar  $\mathbf{x}_{k-1}^i = \tilde{\mathbf{x}}_{k-1}^{j_i}$  y  $w_{k-1}^i = 1/N_s$  para  $i = 1, \dots, N_s$ .
    - else**
    - Asignar  $\mathbf{x}_{k-1}^i = \tilde{\mathbf{x}}_{k-1}^i$  para  $i = 1, \dots, N_s$ .
    - end if**
    - Cálculo de la estimación.
    - end for**
- 

En resumen, el algoritmo de Filtro de Partículas proporciona una forma simple y efectiva de modelar procesos estocásticos con funciones de distribución de probabilidad y modelos de propagación arbitrarios. Se basa para esto en métodos secuenciales de Monte Carlo [Carpenter et al. (1999)]. Para representar la densidad de probabilidades se utilizan puntos mäsicos (partículas) que son estados posibles del proceso, distribuidos sobre su espacio de estados. Cuando se obtiene una medida, el PF utiliza éstos puntos mäsicos para aproximar la distribución a posteriori.

Para obtener una buena aproximación se requiere un número significativo de partículas lo que implica gran capacidad de almacenamiento y comunicación.

### 3.4.1. Trabajos Relacionados

La estimación a partir de diversas fuentes distribuidas implica que es necesario seleccionar una estructura de fusión. En [Sheng and Hu (2005)] se presenta un PF distribuido de dos niveles para aplicaciones de seguimiento de un móvil. Se divide la red en subconjuntos no correlacionados dentro de los cuales los nodos se comunican secuencialmente. El segundo nivel corre Filtros de Partículas en paralelo para obtener estadísticas suficientes del estimado y luego envía la información a un nodo central para obtener allí el estimado final. Se presentan dos algoritmos de PF, el primero actualiza los pesos de importancia de un subconjunto de nodos a otro. El estimado de cada subconjunto es construido con las observaciones y estimaciones de los subconjuntos precedentes; y los estimados parciales se comunican inalámbricamente al siguiente subconjunto. El PF del segundo nivel es utilizado por cada subconjunto para actualizar los pesos propios en paralelo basándose solo en sus observaciones locales. Los estimados locales son comunicados por toda la región de nodos activos hasta un nodo central para actualizar la estimación final. Para reducir la información a transmitir se aproximan los estimados con un Modelo de Mezcla de Gaussianas (GMM, por su nombre en inglés Gaussian Mixture Model) de baja dimensión, por lo tanto en lugar de transmitir las partículas se transmiten los parámetros del GMM. En el primer nivel los PF requieren un número mayor de partículas y menos comunicación que en el segundo nivel.

En [Coates (2001)] se presenta un algoritmo Bayesiano descentralizado y su extensión al control de la información. Se consideran diferentes representaciones de la pdf, una pdf Gaussiana (Filtro Informativo), una pdf discreta (PF), una pdf como grilla y una pdf híbrida. El filtro en general se utiliza para clasificación y localización de móviles y/o fuentes, en el caso de una representación de la pdf híbrida por ejemplo, se realiza la clasificación del móvil con un PF (pdf discreta) y su localización con una pdf Gaussiana. Las experimentaciones se realizan con una red de sensores activa formada por sensores pasivos y sensores robots.

Muchas investigaciones tienen el fin de analizar el comportamiento de las WSN y los algoritmos, y compararlos. En [Akyldiz et al. (2008)] se describen investigaciones

de prototipos de sensores multimedia y su integración en bancos de prueba para evaluación de algoritmos y protocolos de WSN; se citan productos comerciales e interfaces para la programación de los nodos; además en [Reichenbach et al. (2008)] se realiza una comparación de diferentes algoritmos de localización y su eficiencia mediante simulaciones de los mismos; comparando el producto de error de potencia (PEP, por su nombre en inglés, Power-Error-Product).

Siguiendo con los algoritmos de localización y seguimiento, en estas aplicaciones los sistemas incluyen modelos no lineales y ruidos no Gaussianos en su dinámica ([Arunlampalam et al. (2002)], [Daum (2005)]). Por ejemplo, si se utiliza un sensor de rango solamente para ubicar el objeto de interés, la actualización resultante es claramente no Gaussiana en coordenadas cartesianas. Lo mismo vale para el caso de sensores de ángulo solamente. La utilización de un filtro que supone distribuciones Gaussianas lleva indefectiblemente a inconsistencias. Algunas formas de solucionar este problema es retrasando la actualización hasta obtener información suficiente de manera que la forma de la pdf posterior del conjunto se aproxima a una Gaussiana. En el ejemplo de sensores de rango solamente, con al menos tres medidas de distintos sensores no ubicados en el mismo lugar, sería posible obtener una estimación prácticamente Gaussiana. Este problema aparece a menudo en aplicaciones de robótica como el SLAM [Durrant-Whyte and Bailey (2006)].

En [T.Zhao and A.Nehorai (2007)] se realiza una estimación Bayesiana distribuida que se transmite entre nodos. Se presentan dos métodos para representar la estimación; una representación de densidad Gaussiana y otra como combinación lineal de funciones de densidad Gaussianas.

Otra cuestión importante es si las incertidumbres de las estimaciones a priori son grandes y se produce una actualización muy precisa comparada con esta. Si se utilizan filtros basados en linealizaciones como el Filtro Extendido de Kalman ([Ljung (1979)], [Welch and Bishop (1995b)]) se viola la suposición común de linealizar cerca del punto de operación, llevando a resultados catastróficos.

Hay una gran variedad de Filtros de Partículas, modificaciones al filtro SIR, de



modo de adaptarlo para determinada tarea y requerimientos. En el caso de su implementación sobre redes de sensores, es primordial reducir al mínimo los datos a tener almacenados y a comunicar. En las tareas de localización y seguimiento, que son las que se desea abordar aquí, los modelos de sistema y observación son no lineales y eventualmente con medidas no Gaussianas, es por ello que es muy importante lograr una buena aproximación de la pdf; en general para que un PF trabaje con precisión se requiere un gran número de partículas en espacios de 3 dimensiones para aproximar en forma precisa a la función de distribución de probabilidades, dependiendo del sistema con que se este trabajando. En [Campillo and Rossi (2006)] se proponen y comparan métodos de estimación eficientes basados en simulación utilizando PF de convolución; primero se consideran estimadores no Bayesianos como el Estimador de Mínimos Cuadrados Condicional (CLSE) y el Estimador de Máxima Probabilidad (MLE) y luego en el contexto de estimadores Bayesianos se presenta un método de Monte Carlo.

En [Kotecha and Djuris (2003)] se aproxima la estimación por la media y la covarianza de la variable de estado desconocida (la distribución de partículas) utilizando además muestreo por importancia. Debido a que para la actualización solo basta con la media, la varianza y una nueva medida del sensor, el algoritmo es óptimo para aplicarlo cuando es necesario transmitir mensajes con la menor extensión y frecuencia posible; pero se debe tener en cuenta que los estimados serán representativos solo en el caso que las distribuciones de probabilidad sean Gaussianas; de lo contrario podrían aproximarse dichas distribuciones como suma de Gaussianas [Kotecha and Djuric (2001)] aumentando ampliamente el procesamiento requerido.

En [Fox (2003)] presenta un algoritmo para acotar el error de aproximación de la pdf que da la representación de partículas, que es medido utilizando la distancia de Kullback-Leibler. Se elige un número pequeño de muestras si la densidad se focaliza en una pequeña parte del espacio de estados, y un número elevado de muestras si la incertidumbre es mayor. En [Mourad et al. (2008)] se presenta un algoritmo para redes móviles ad-hoc (MANET, Movil Ad-hoc Networks). La incertidumbre de la estimación de cada nodo se propaga en forma de intervalos. Se llega a una estimación en línea

garantizada y robusta de la posición del sensor móvil.

En [Manjunath et al. (2008)] se presenta un algoritmo que elige si utilizar un SIR PF o un Unscented PF (UPF) según sea el error en la estimación; de ésta forma acopla la estructura simple tanto de la evaluación de los pesos como el muestreo del SIR con la precisión del UPF. Se demuestra que cuando los modelos de medida son altamente no lineales el nuevo algoritmo es más preciso que el SIR y más rápido que el UPF. La eficiencia del filtro SIR es baja debido a que la densidad de importancia no incluye la predicción de la última observación. El UPF utiliza el Unscented KF como densidad de importancia, incorporando la observación más actual en la función de probabilidad, para propagar las partículas.

Otro de los factores importantes en los algoritmos de PF es el remuestreo. Se han publicado muchos métodos para hacerlo en forma sencilla y eficiente [Bolić et al. (2003)]. En [Park et al. (2007)] se utiliza el Filtro Genético en el PF para solucionar el problema de empobrecimiento de muestras. La idea básica es asociar las partículas como cromosomas del algoritmo genético. El Filtro Genético consiste entonces en tres pasos. El paso de muestreo donde una nueva población de cromosomas se genera de la población anterior de acuerdo a la probabilidad de transición. El paso de operación genética, el cruzamiento y mutación se aplican al conjunto temporario de cromosomas; previniendo el empobrecimiento de muestras; el cruzamiento se realiza eligiendo un par de cromosomas y combinándolos para producir un nuevo par de hijos, dichos hijos se ubican en la región convexa de los padres. Y finalmente el paso de remuestreo es donde se evalúan los pesos de los cromosomas y se remuestran de acuerdo a ello. En resumen, cuando se reciben las partículas del instante anterior, se mide y se realiza el cruzamiento y mutación de las partículas; luego se evalúan sus pesos y se realiza el remuestreo. El cruzamiento se realiza eligiendo un par de partículas y se combinan para producir un nuevo par de partículas; la mutación genera una partícula como combinación de la posición de la partícula y de la observación. Las operaciones genéticas extienden el espacio de estados, permitiendo seguir un vehículo que se mueve rápido, debido a que siempre hay partículas cerca del estimado de la observación debido a la mutación. Las

operaciones genéticas son muy fáciles de implementar en hardware debido a que son únicamente sumas y restas.

El problema más recurrente de los PF es el empobrecimiento de muestras. En [Gordon et al. (2002a)] se presenta el Filtro Bootstrap; donde se incorporan dos métodos para evitar el empobrecimiento de muestras, llamados “*Roughening*” y “*Prior Editing*” que requieren procesamiento considerable. En [Douc et al. (2007)] se plantea una forma de selección de las partículas mediante un análisis asintótico de los pesos. El PF Auxiliar permite afectar la ubicación de las partículas asignando libremente un conjunto de *pesos de importancia de primer etapa* (first stage weights). Esto implica asignar grandes pesos a partículas cuya generación caerá en zonas del espacio de estados de mayor probabilidad. A pesar de ello, ésta posibilidad no es aprovechada en muchos casos en forma óptima, por ello aquí se presenta una manera de identificar asintóticamente los pesos de importancia de primer etapa mas eficientes. El resultado es importante para la ubicación óptima de las partículas y brinda una interpretación en términos de la reducción de varianza de dichas partículas.

En [Carpenter et al. (1999)] se muestra un método para monitorear la eficiencia de diferentes filtros de partículas (SIR, entre otros) y que permite detectar aproximadamente en que iteración del algoritmo ocurre el empobrecimiento de muestras y se propone una manera de calcular el número de partículas efectivo necesario para lograr una precisión determinada en la medición. También se presenta un algoritmo de Filtro de Partículas de mejor desempeño, aunque mas difícil de implementar debido a su complejidad computacional. Lo que se puede destacar es que todos los filtros evaluados presentan empobrecimiento de muestras luego de un determinado número de iteraciones.

Como se dijo anteriormente, la estimación en aplicaciones de localización y seguimiento es una gran rama de investigación; hay varias publicaciones que realizan una presentación y estudio de un conjunto de estimadores que brindan un panorama de la evolución de dichos algoritmos, [Welch and Bishop (1995b)], [Handschin (1970)], [Chen (2003)], [Gordon et al. (2002b)], [Arulampalam et al. (2002)], [Daum (2005)].

## 3.5. Conclusiones

En el presente capítulo se han dado algunas nociones básicas de estadística necesarias para entender los conceptos presentados en el siguiente capítulo, que presenta al Filtro Acotado de Partículas, objeto de esta tesis.

También se ha presentado el concepto de fusión de datos; a partir del cual se introdujeron el filtrado recursivo de Bayes y los Algoritmos de Muestreo Secuencial que derivan en el Filtro de Partículas. Finalmente se han señalado algunos trabajos de investigación sobre los temas tratados en caso de que se quiera ampliar el marco de conocimientos.

# Capítulo 4

## Filtro de Partículas Acotado

### 4.1. Filtro de Partículas Acotado

#### 4.1.1. Propuesta.

En el capítulo anterior se presentó el Filtro de Partículas, un estimador que permite aproximar funciones de distribución de probabilidad arbitrarias de sistemas con modelos no lineales de proceso y observación y con ruidos no Gaussianos. Esto lo hace apropiado para aplicaciones de localización y seguimiento de fuentes, objetos u eventos; que es la tarea que se desea abordar. Se presentaron además las restricciones de procesamiento, almacenamiento y comunicación de datos en redes de sensores inalámbricas. A continuación se presenta el resultado mas importante de esta tesis, que es un algoritmo denominado Filtro de Partículas Acotado que permite implementar un estimador con las condiciones expuestas y que además cumple con las restricciones de las WSN.

El Filtro de Partículas Acotado opera en las etapas de predicción y actualización en cada nodo, en esencia, como un Filtro de Partículas; pero su inicialización y estimación son especialmente modificadas. Esta modificación, que consiste en acotar las zonas de alta probabilidad, permite operar con las restricciones de las WSN.

El estimado esta formado por las cotas que encierran las partículas mas pesadas y la estadística de la distribución de las partículas seleccionadas que conforman dicha área según la aplicación lo requiera, como ser la partícula mas probable, la media y

la varianza de las muestras, entre otras. Sin embargo en muchas aplicaciones lo mas importante es reducir al mínimo la comunicación entre nodos, es en ese caso que en la estructura del algoritmo propuesto es posible transmitir las cotas, de modo que el nodo siguiente genere las partículas con distribución uniforme dentro de ellas y comience una nueva iteración del algoritmo. En las aplicaciones presentadas a continuación se demuestra que con mínima comunicación se logra un buen resultado.

El algoritmo propuesto permite el cálculo de un estimado íntegramente dentro del nodo; tiene mínimos requerimientos de procesamiento, almacenamiento y comunicación; y la fusión de datos es totalmente distribuida.

#### 4.1.2. Algoritmo General.

El algoritmo propuesto es una modificación del Filtro de Partículas que permite representar funciones de probabilidad no Gaussianas y cumple con los requerimientos de mínimo procesamiento, almacenamiento y transmisión de datos que imponen las redes de sensores. El algoritmo es denominado Filtro de Partículas Acotado (BPF, por su sigla en inglés Bounded Particle Filter) y comienza en un nodo líder con una distribución de partículas inicial, que se extiende sobre todo el espacio de estados de interés. Luego de una medición y una actualización de los pesos de las partículas, se realiza la selección y se envían al siguiente nodo propiedades importantes de las partículas seleccionadas. En la versión de mínima información transmitida, se comunican al siguiente nodo de la red las cotas del espacio estimado que encierran las partículas seleccionadas. La selección del nodo excede el alcance de la tesis; un método de selección posible es el nodo vecino mas informativo [Kaplan (2006)].

Cada nodo recibe las cotas y genera una distribución de partículas en el área delimitada por las mismas, donde se realiza la predicción y la actualización de los pesos. La densidad de probabilidad es representada por partículas pesadas que son posibles estados del proceso, distribuidas sobre el espacio de estados total. En el caso de mínima transmisión de datos, como distribución *a priori* utiliza  $N_s$  partículas generadas dentro de las cotas resultado de la iteración anterior con peso uniforme  $w_0^i = 1/N_s$ .

Sobre dicha distribución se calcula la predicción con el modelo de proceso (4.1).

$$\tilde{\mathbf{x}}_{k/k-1}^i = f[\tilde{\mathbf{x}}_{k-1/k-1}^i] + \mathbf{u}_k^i \quad (4.1)$$

Donde  $\tilde{\mathbf{x}}_{k/k}^i$  representa cada partícula,  $f[\cdot]$  es la función que propaga los estados en el tiempo y  $\mathbf{u}_k$  es el ruido aditivo de distribución arbitraria. Para cada partícula que representa un estado posible, cuando se realiza una observación o medición del sensor  $\mathbf{z}_k$ , se calculan los nuevos pesos que representan la función de probabilidad condicionada en la partícula  $i$ . Estos se calculan según la ecuación (4.2) para lo que se requiere la función de verosimilitud  $p(\mathbf{z}_k | \mathbf{x}_k)$ :

$$w_i = \frac{p(\mathbf{z}_k | \tilde{\mathbf{x}}_{k/k-1}^i)}{\sum_{j=1}^N p(\mathbf{z}_k | \tilde{\mathbf{x}}_{k/k-1}^j)} \quad (4.2)$$

Cuando la observación  $\mathbf{z}_k$  posee ruido aditivo gaussiano de media cero y de varianza conocida  $R$ , la ecuación (4.2) se resume en:

$$w_i = \frac{e^{-(z - h_i[\tilde{\mathbf{x}}_{k/k-1}^i])^2 / 2R}}{\sqrt{2\pi R}} \quad (4.3)$$

Donde  $\mathbf{v}_k$  es el vector de ruido de medición que modela perturbaciones aleatorias e incertidumbres del modelo de observación y  $h_i[\tilde{\mathbf{x}}_{k/k-1}^i]$  es la predicción de la observación sobre la partícula  $i$  calculada con (4.4).

$$h^i[\tilde{\mathbf{x}}_{k/k-1}^i] = \tilde{z}_k^i = h_i[\tilde{\mathbf{x}}_{k-1/k-1}^i] + \mathbf{v}_k^i \quad (4.4)$$

En este punto en el PF, un algoritmo del tipo SIR [Gordon et al. (2002a)] realiza el remuestreo que selecciona en forma probabilística las partículas de la distribución *a posteriori*. La etapa de remuestreo requiere un procesamiento intensivo que, por sus restricciones, un nodo de una red de sensores no podría realizar. Debido a esto, se propone una estrategia que selecciona las partículas ubicadas sobre el área más probable, comparando sus pesos con un umbral de peso. El umbral de peso debe considerar el número de partículas que se requiere que sobrevivan y las incertidumbres del proceso y

del sensor. En esta tesis se denota el umbral de peso como  $CotaP$ , es un umbral adaptivo en cada iteración para lograr conservar un número mínimo de partículas  $N_{min}$  que sea representativo del estado.  $N_{min}$  debe ser función de la incertidumbre y ruido del sensor y los modelos.

Las partículas cuyo peso supere el umbral son seleccionadas y representan los estados mas probables de la estimación.

$N_s$  es el número inicial de partículas generadas en cada iteración. Este número, que en principio puede ser fijo, puede optimizarse en la evolución de la estimación. La racionalidad detrás de esto es la siguiente. En la primer iteración se parte de un espacio grande dada la incertidumbre inicial y conecuentemente debo tener un número grande de partículas para representar la densidad. Luego de varias iteraciones, este espacio se acota a una región que será función de las incertidumbres de modelo y observación y por lo tanto mucho menor que el espacio inicial y no requeriría de un número de parrtículas importantes para representarla.

En un PF la varianza de los pesos aumenta con el tiempo, y aparece el problema de degeneración o empobrecimiento de partículas, resultando en una sola partícula con peso no despreciable [Arulampalam et al. (2002)]. En el algoritmo de BPF se seleccionan las partículas de mayor peso, de ellas se extraen las cotas del área que las contiene y la estadística de interés; la siguiente generación de partículas se hará dentro de dichas cotas, de modo que la varianza de los pesos se reduce y, en el caso de la localización, el algoritmo se detiene cuando la varianza en el peso de las partículas se mantiene casi constante para las  $N_{min}$  partículas. Además el hecho de realizar una regeneración de partículas dentro de las cotas en cada iteración del algoritmo y teniendo en cuenta que en la selección se exige un número mínimo de partículas seleccionadas, se evita el problema de degeneración.

Luego de la selección de partículas, se extrae el área mínima en que se encuentran en forma de límites mínimos y máximos. En las aplicaciones de localización y seguimiento, los límites están dados en rango y ángulo debido a que se utilizan sensores de rango y/o ángulo; centrados en la posición del sensor. Cuando  $\mathbf{z}_k$  es un nivel de presión sonora,



la predicción de la medición es calculada con (4.5), donde  $r_i$  es la distancia entre el sensor y la partícula  $i$ , y  $A$  y  $C$  son constantes para aproximar el modelo a la curva de medición del sensor.

$$h_i[\tilde{\mathbf{x}}_{k/k-1}^i] = \frac{A}{\sqrt[2]{r_i}} + C \quad (4.5)$$

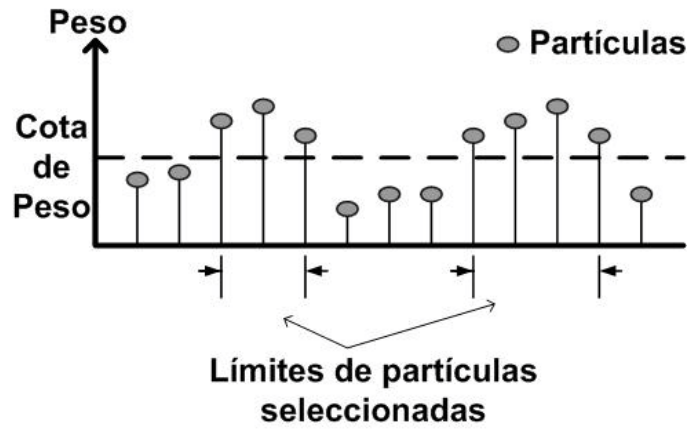
El modelo no tiene en cuenta los ecos de potencia, ni la reverberación debido a la presencia de obstáculos como paredes, ni los efectos en la medición debidos al viento ([Li et al. (1994)], [Salomons (1994)]).

Cuando  $\mathbf{z}_k$  es una medida de rango, la predicción de la medición es calculada con (4.6), donde  $r$  es la distancia entre el sensor y la partícula  $i$ .

$$h_i[\tilde{\mathbf{x}}_{k/k-1}^i] = r_i \quad (4.6)$$

La posición del sensor y los límites son comunicados al nodo siguiente, o al nodo central de procesamiento. En [Sañudo and Masson (2006a)] las partículas son encerradas con cotas de ejes cartesianos X-Y. Debido al tipo de sensor utilizado, sensor de rango, se pierde mucha información con ese método ya que se encierran zonas de baja probabilidad en la estimación, por esa razón se decide encerrar las partículas en coordenadas polares.

La Fig. 4.1 muestra un ejemplo en una dimensión, y las regiones acotadas más probables. Las zonas acotadas y la información propia del sensor son transmitidas al nodo siguiente, donde se actualiza la estimación utilizando una nueva observación.



**Figura 4.1:** Esquema de selección de partículas en una dimensión: Se seleccionan las partículas cuyo peso supera la *Cota de peso* y se extraen los extremos de los intervalos que las contienen.

Como se puede observar en la Fig. 4.1, es fundamental el valor umbral  $CotaP$  que selecciona las partículas. Una primera aproximación es calcularlo como un porcentaje  $P$  del peso máximo. Así  $CotaP$  será calculado a partir de la ecuación (4.7) y donde  $P$  se establecerá en función de las varianzas de los ruidos aditivos del modelo dinámico y de observación tanto como del número mínimo de partículas sobrevivientes  $N_{min}$ . Como ejemplo, en los casos experimentales presentados en esta tesis,  $P$  resulta en un 80% cuando las medidas son obtenidas por un sensor laser dado su precisión. En el caso de los Sensores de Nivel de Potencia Acustica,  $P$  resultó en un 70%.

En la siguiente sección se hará un análisis del valor de este parámetro  $P$ . Mas allá de esta estimación de la  $CotaP$ , si como resultado del proceso de selección el número de partículas obtenido es menor a  $N_{min}$ , se debe reducir su valor de forma de alcanzar este número, como se muestra en el Alg. 3. Resultando el cálculo de la ecuación (4.7) una buena primera aproximación.

$$CotaP = W_{max} \frac{P}{100} \quad (4.7)$$

Un esquema que permite observar las diferentes etapas en el proceso de localización de una fuente en una red de sensores se muestra en la Fig. 4.2, donde se utilizan sensores

---

**Algoritmo 3** Adaptación de la cota de peso  $CotaP$ .

---

$$CotaP = W_{max} \frac{P}{100}$$

Selección de partículas cuyo  $w_i \geq CotaP$ .

**while** Número de partículas seleccionadas  $\leq N_{min}$  **do**

Reducir  $P$ .

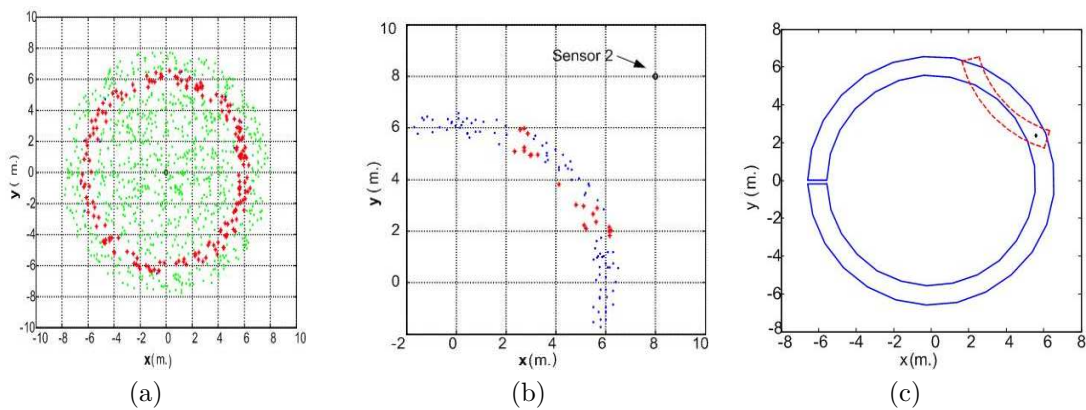
$$CotaP = W_{max} \frac{P}{100}$$

Selección de partículas cuyo  $w_i \geq CotaP$ .

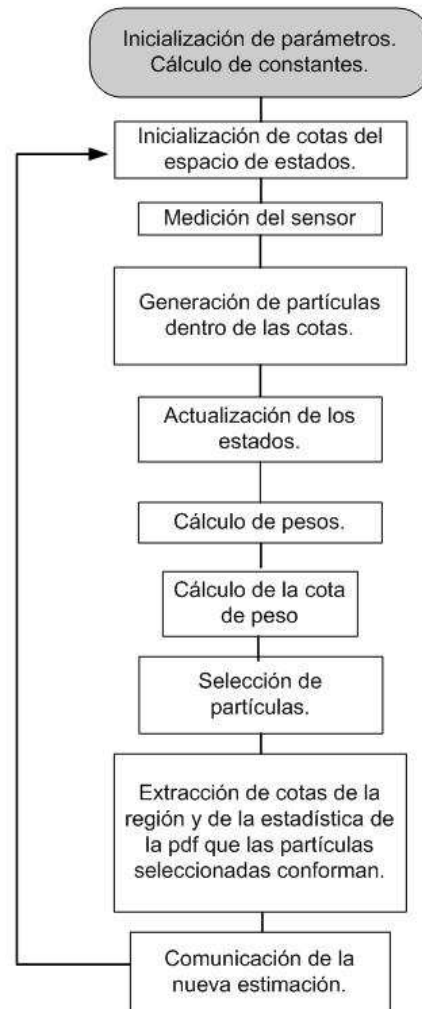
**end while**

---

de rango. Primero, en la Fig. 4.2(a), las partículas son distribuidas uniformemente sobre toda el área; cuando una medición es recibida se realiza la primer estimación, las cotas son extraídas y transmitidas con la posición del nodo, al nodo siguiente. En el nodo que sigue, Fig. 4.2(b), las partículas son generadas dentro de las cotas recibidas del nodo anterior y luego trasladadas respecto del sensor actual. Con una nueva medición una nueva estimación es generada y nuevamente las cotas son extraídas. El mismo procedimiento es repetido hasta que la estimación no reduzca el área acotada debido al alcance y precisión de los sensores o debido a que las partículas han alcanzado su número mínimo de supervivencia,  $N_{min}$ , y la varianza de sus pesos se mantiene casi constante. En la Fig. 4.2(c) se ven los resultados de cada estimación, el objeto a localizar es el círculo negro encerrado por ambas estimaciones.



**Figura 4.2:** Ejemplo visual del funcionamiento del Filtro de Partículas Acotado para una fusión descentralizada entre dos nodos con sensores de rango: (a) Estimación inicial del nodo sensor ubicado en (0,0), en rojo las partículas seleccionadas. (b) Segunda estimación, del nodo sensor ubicado en (8,8), y como resultado de la estimación se comunican las cotas que encierran las partículas rojas, las mas probables. (c) Áreas estimadas en cada caso.



**Figura 4.3:** Diagrama en bloques del Algoritmo BPF.

El algoritmo es resumido en el diagrama de bloques en la Fig. 4.3 y en el Alg. 4. Inicialmente se generan las partículas en el espacio total de estados, con una distribución uniforme, si es el caso mas simple del filtro que implica transmitir sólo las cotas. Una vez que se recibe una observación se propagan las partículas con (4.1). Luego se calcula la predicción de la observación para cada partícula con (4.4) y los pesos con (4.3). Se normalizan los pesos y se calcula la cota de peso como un porcentaje del peso

máximo. Se realiza la selección de partículas, la extracción de las cotas que las encierran y la información que se desee de la distribución que ellas forman para comunicar como dato de estimación.

---

**Algoritmo 4** Algoritmo Filtro de Partículas Acotado.
 

---

◦ **Inicialización:**  $k = 0$  Las cotas iniciales  $\mathbf{c}_0$  contienen el espacio de estados total.  $\mathbf{x}_0$  partículas iniciales de distribución uniforme dentro de  $\mathbf{c}_0$ .

**for**  $i = 1$  to  $N_s$  **do**  
 Asignación inicial de los pesos de importancia  
 $w_0^i = \frac{1}{N_s}$   
**end for**

**for**  $k = 1$  to  $T$  **do**  
 ◦ **Muestreo de Importancia:**  
**for**  $i = 1$  to  $N_s$  **do**  
 Propagación de partículas:  $\tilde{\mathbf{x}}_{k/k-1}^i = f[\tilde{\mathbf{x}}_{k-1/k-1}^i] + \mathbf{u}_k^i$   
 Cálculo de pesos:  $\tilde{w}_k^i = p(\mathbf{z}_k | \tilde{\mathbf{x}}_{k/k-1}^i)$  con (4.3).  
**end for**  
 Normalización de los pesos  $w_k^i = \frac{\tilde{w}_k^i}{\sum_{i=1}^{N_s} \tilde{w}_k^i}$  para  $i = 1, \dots, N_s$   
 Selección: Determinar  $CotaP$  y se seleccionan las partículas cuyo  $w_k^i \geq CotaP$   
 Verificación de representatividad: Verificar que el número de partículas seleccionadas supere  $N_{min}$ ; de lo contrario reducir  $CotaP$  y volver al paso de Selección.  
 ◦ **Estimación:**  
 Determinar el estimado  $\hat{x}_k$  utilizando las partículas  $\tilde{\mathbf{x}}_k^i$  y los pesos  $w_k^i$  o bien extrayendo las cotas que delimitan el área mínima que las contiene,  $\mathbf{c}_k$ .  
 ◦ **Regeneración:**  
 Generar aleatoriamente  $\mathbf{x}_k$  dentro del área delimitada por las cotas  $\mathbf{c}_k$  y se asignan los pesos iniciales  $w_k^i$  para  $i = 1, \dots, N_s$ .  
 Volver al Muestreo de Importancia cuando se obtiene la medición siguiente.  
**end for**

---

Hay una gran línea de investigación en el sentido de lograr algoritmos que cumplan con las restricciones de las redes de sensores; especialmente abocadas a lograr una representación de la estimación de modo de resumir al máximo la información que se deba transmitir. En [Gning et al. (2007)] se utilizan *box particles* (partículas en forma de cajas) en lugar de partículas discretas, lo que permite utilizar un número mas reducido de partículas al costo de desperdiciar información de los sensores, ya que para armar las cajas (boxes) se considera la incertidumbre máxima de cada estado. Si en el BPF se quisiera conservar más información, se podrían definir varias cotas

de peso y tomar subcotas de los estados para acotar varias zonas de las partículas más pesadas, en lugar de las cotas generales. En este caso, el resultado sería similar al que se presenta en esta tesis pero con menos procesamiento y sin desaprovechar la información de los sensores. En [Abdallah et al. (2008)] se presenta un algoritmo de PF que maneja intervalos de datos y utiliza análisis de intervalos y técnicas de satisfacción de restricciones. Las partículas son cajas asociadas con pesos cuya probabilidad se define con un modelo acotado del error de observación. El problema de Satisfacción de Restricciones (CSP) es básicamente encontrar la mínima caja que esté contenida en la caja previa y además esté contenida en el conjunto de cajas que satisfacen la restricción. Se sigue manteniendo el problema de tener que transmitir muchos datos (el conjunto de cajas) para la siguiente estimación.

## 4.2. Aplicaciones de Localización y Seguimiento

En esta sección se presentan diversas redes de sensores aplicadas a localización de fuentes y objetos, y al seguimiento de objetos con el BPF. Se presentan los resultados de su comportamiento ante la variación de diferentes parámetros. Para el análisis que se presenta a continuación, es necesario definir algunos de los parámetros y conceptos usados a lo largo de la sección:

- Número de Partículas Iniciales  $N_s$ : Es la cantidad de partículas a generar inicialmente, en el comienzo de cada iteración del algoritmo BPF. En el caso de la primera estimación, cuando no hay información anterior, se generan las partículas con distribución uniforme; al igual que en el caso de que la única información recibida de la estimación anterior sean las cotas que encierran la región más probable del espacio de estados. En el caso que se reciba información adicional a las cotas, como ser la media de las partículas, o la probabilidad en varias zonas dentro de las cotas, se generan las partículas haciendo uso de dicha información. Se podría adaptar este parámetro dependiendo de si se tiene información anterior o no, es decir, utilizar un  $N_s$  mayor en el caso de tener que realizar la estimación

en el espacio total de interés, o bien uno menor en caso de que se estime sobre un espacio que es resultado de una estimación anterior, como se presentó en la Sección 4.1. Además, si se pudiera agregar procesamiento al nodo, se podría calcular  $N_s$  como el valor eficaz  $N_{eff}$  según lo presentado en [Arulampalam et al. (2002)].

- **Corrida:** Se denomina así al conjunto de estimaciones consecutivas hasta llegar a la estimación final. Cada estimación surge de aplicar el algoritmo BPF. Una corrida esta formada por la aplicación del BPF varias veces en forma secuencial hasta lograr esa estimación final.
- **Número de Estimaciones Consecutivas ó Número de Iteraciones hasta llegar a un estimado final:** En éste caso es importante aclarar que el algoritmo realiza un número determinado de estimaciones consecutivas; la última de éstas estimaciones es a la que se denomina estimación final. Luego de la estimación final el algoritmo comienza nuevamente a estimar sin tener en cuenta la información obtenida hasta el momento; es decir, se inicia con una distribución uniforme de partículas sobre el área (espacio de estados) total a abarcar con la red de sensores.
- **Orden de los sensores:** Se refiere al orden en que se reciben las mediciones de los nodos y en cuyo orden se procederá a estimar en forma consecutiva (secuencial). En el caso del BPF implementado sobre los nodos, el orden de los sensores es el orden en que los nodos van comunicando el estimado entre sus vecinos.
- **Número Mínimo de Partículas Sobrevivientes ó Número Mínimo de Partículas  $N_{min}$ :** Se trata del número mínimo de partículas que se considera representativo del estado; depende de la incertidumbre de los modelos de dinámica del objeto y de los modelos de observación; y de la precisión de los sensores. El algoritmo conserva las partículas que superan una cota determinada; en caso de ser la cantidad de partículas a conservar menor a  $N_{min}$  se modifica la cota de peso (se reduce) hasta lograr el objetivo (Alg. 3).

### 4.2.1. Sensores de Rango y Ángulo Utilizados

En el trabajo presentado se utilizan sensores de rango, por ello la predicción de la medición,  $h_i[\tilde{\mathbf{x}}_{k/k-1}^i]$ , es calculada con la distancia entre el sensor y la partícula  $i$  cuando la observación o la medición del sensor,  $\mathbf{z}_k$ , es directamente la medida de rango.

Para obtener datos experimentales de rango y ángulo se utilizará un láser LMS 291 ([Sic], [LMS]) cuyo alcance máximo programado es de 8 m.; el error en rango es de 1 mm. y el error en ángulo es de 0.5 °; teniendo en cuenta el alcance del láser y que el error máximo lo aporta cuando el objeto a localizar se encuentra en línea recta de sus laterales (0° y 180°). Los datos basados en este sensor son bastante precisos.

Para verificar el funcionamiento del filtro con modelos de ruido más imprecisos, se utilizaron los sensores de nivel de presión sonora ([Sondon et al. (2007a)], [Sondon et al. (2007b)]), que fueron diseñados en la Universidad Nacional del Sur. Con estos sensores es posible realizar mediciones de polución acústica o detección de ángulos de arribo de vehículos. La arquitectura de los sensores combina diseño analógico y digital. El procesamiento de la señal muestreada es realizado en un dispositivo de lógica reconfigurable cuya estructura ha sido hecha en lenguaje de descripción de hardware (HDL, Hardware Description Language). El rango de niveles que soporta es de 55 a 87 dBA.

Cuando un objeto se localiza mediante la medición de rango u ángulo pero no emite sonido, se denomina, simplemente, objeto. Se denominan fuente aquellos objetos que emiten sonido y mediante la medición del nivel de sonido se realiza la localización.

### 4.2.2. Filtro de Partículas Acotado en Coordenadas Cartesianas

Se presenta una aplicación de localización en 2D de un objeto, utilizando sensores de rango y ángulo, con el propósito de resaltar aspectos del funcionamiento del BPF que luego serán tenidos en cuenta en la implementación sobre una red de sensores real. En este análisis se representa la estimación como un área rectangular, cotas máximas y mínimas en cada una de las coordenadas, que encierra las partículas seleccionadas. Esto



simplifica tanto el cálculo de las cotas como la información final del estimado. La red analizada tiene una estructura de fusión centralizada en la que cada nodo comunica el resultado de la estimación individual a un nodo base que en este caso será una computadora que hará la mayor parte del procesamiento. En este análisis se simplifica el problema de fusión dado que se desea resaltar la operación del filtro propuesto.

En esta experiencia se pondrá de relieve además aspectos de la implementación que tienen un impacto relevante en el funcionamiento del filtro; como la forma en que se calculan las cotas o cómo se generan las partículas dentro de estas.

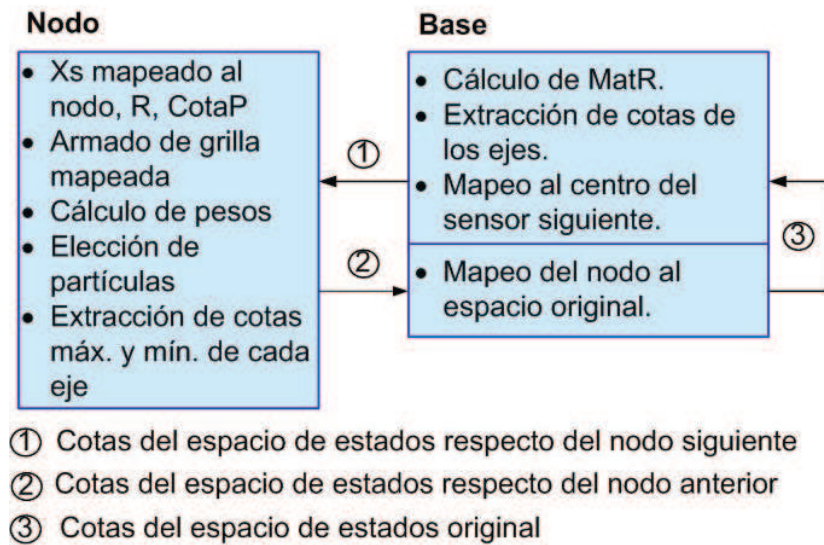
La implementación del BPF se realizó en parte sobre los nodos, pero la parte de mayor procesamiento en la base, una computadora, con Matlab [The MathWorks (1994-2010)].

### Implementación Práctica

Con el objetivo de analizar el funcionamiento del BPF en una aplicación concreta de localización, se adoptaron algunas simplificaciones prácticas en la implementación del algoritmo BPF.

En el instante inicial se distribuyen  $N_s$  partículas sobre el espacio de estados a cubrir (total), equi-espaciadas en cada una de las coordenadas; a ésta distribución se la denomina grilla.

Se adopta una estructura de red centralizada en la que el nodo líder funciona como nodo base y el resto de los nodos comunican sus resultados de su estimación local a este. El esquema propuesto es el que se muestra en la Fig. 4.4. Allí  $R$  es la incertidumbre del sensor,  $CotaP$  la cota de comparación de peso para la selección de las partículas,  $X_s$  la posición del sensor y  $MatR$  es la matriz de rotación en (4.8), donde  $\alpha = atan(\frac{y_a}{x_a})$ . El funcionamiento del esquema es el siguiente. Cada sensor luego de la estimación genera localmente las cotas que transmite a la estación base, y que están expresadas en coordenadas locales centradas en su posición. En la base, se transforman estas cotas a un eje de coordenadas global, que luego con  $MatR$  y la posición de los sensores se transforman al eje de coordenadas local del nuevo sensor. Estas cotas se envían al nuevo



**Figura 4.4:** Distribución de las tareas del algoritmo entre la base o nodo central y los nodos sensores

sensor, quien genera dentro de ellas las nuevas partículas para procesar su observación y generar un nuevo estimado utilizando *CotaP* como el umbral de partículas más pesadas.

Como se dijo, el objetivo de esta implementación es realizar un análisis del comportamiento del filtro. Por un lado se está tratando de localizar un objeto fijo lo que simplifica el proceso de modelado al no requerir un modelo de movimiento ni tener restricciones de tiempo. Además se limita el procesamiento en los nodos descargado gran parte de este en la base quien además almacena todos los datos intermedios del proceso de estimación

$$MatR = \begin{bmatrix} \cos(\alpha) & \text{sen}(\alpha) \\ -\text{sen}(\alpha) & \cos(\alpha) \end{bmatrix} \quad (4.8)$$

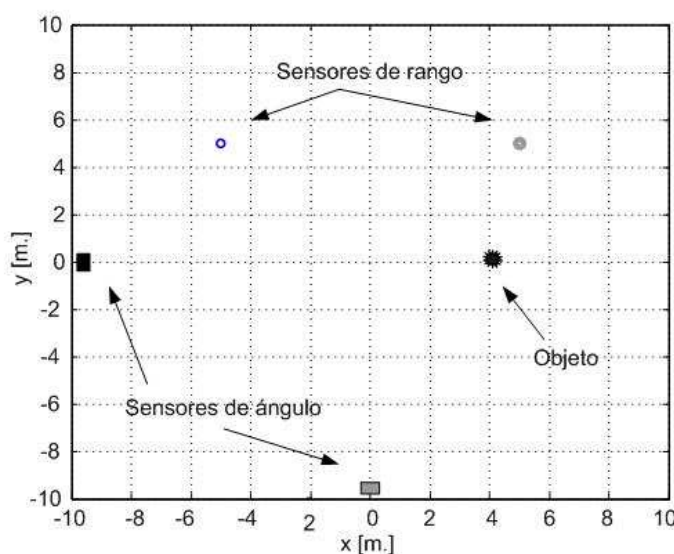
Con cada una de las medidas obtenidas por cada nodo, el conjunto de partículas se irá ubicando sobre zonas restringidas del espacio total de estados (se contará con un sub-espacio diferente según sea el sensor y su posición), es por ello que deben ser trasladadas a un espacio común a la totalidad de los sensores para la fusión; un espacio

con media cero y en el que los rectángulos que contienen las partículas se encuentren alineados al espacio de estados. Esto se logra utilizando como se dijo, la matriz de rotación ( $MatR$ ) y la media.

Para fusionar todas las medidas se podría intersectar todos los subespacios de estados; en [Isler and Bajcsy (2005)] se puede ver un ejemplo de aplicación de éste principio. Una vez lograda la intersección, este será el espacio inicial para la nueva estimación en cada sensor. El paso de fusión de datos no se muestra en la presente tesis.

### Simulaciones y Resultados

Para el ejemplo se utilizaron cuatro nodos sensores, dos de ángulo y dos de rango; y un nodo base para la cobertura de un espacio de estados de 20 m. por 20 m. En la simulación se ubicaron los sensores y el objeto a localizar como muestra la Fig. 4.5. Los sensores de ángulo se representan con rectángulos y se encuentran en las coordenadas  $[-10 \ 0]$  y  $[0 \ -10]$ ; perpendiculares uno respecto del otro. Los sensores de rango se encuentran ubicados en las coordenadas  $[-5 \ 5]$  y  $[5 \ 5]$ . El objeto a detectar se ubica en  $[4 \ 0]$ . Se adoptó  $N_s = 500$ . Todos los nodos comunican la información a la base, donde se trasladan y rotan para representarlos en el espacio de estados.



**Figura 4.5:** Esquema de la ubicación de los nodos y el objeto para el ejemplo de localización en 2D simulado acotando en coordenadas cartesianas.

Se simulan las observaciones de los sensores. En (4.9) se presenta el modelo para generar las observaciones del sensor de ángulo y en (4.10) el modelo para generar las observaciones logarítmicas del sensor de rango; donde  $v$  es la incertidumbre del modelo;  $x$  e  $y$  son la ubicación correspondiente en el espacio de estados.

$$z = \arctan(y/x) + v \quad (4.9)$$

$$z = \log\left(\frac{1}{\sqrt{x^2 + y^2}} + v\right) \quad (4.10)$$

A continuación se utiliza el modelo de aproximación del sensor de ángulo (4.11) para el cálculo de la predicción de la observación de ángulo y el modelo de aproximación del sensor de rango (4.12) para el cálculo de la predicción de la observación de rango, sobre cada partícula.

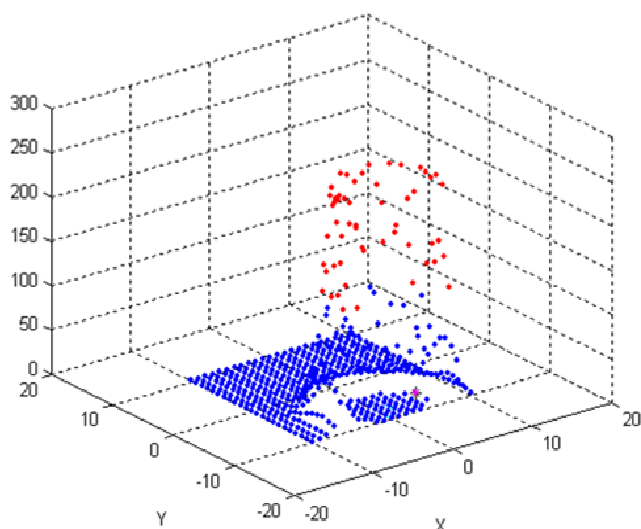
$$h = \arctan(y/x) \quad (4.11)$$

$$h = -\log(\max(x, y)) - \log\left(\left(\frac{\min(x, y)}{\max(x, y)}\right)^2 + 1\right) \quad (4.12)$$

El logaritmo es utilizado para comprimir el rango dinámico de las medidas, facilitando luego la elección de  $CotaP$ . Finalmente los pesos se calculan con (4.3), teniendo en cuenta, para el caso del sensor de rango, que los valores de  $R$ ,  $z$  y  $h_i$  deben expresarse en logaritmo base diez.

Se supone que los nodos reciben, inicialmente, solo las cotas del espacio de estados a estimar en coordenadas cartesianas donde procede como el *Nodo* de la Fig. 4.4. Al principio se realizan varias medidas y estimaciones dentro de cada nodo sensor antes de enviar el resultado a la base. Esto es conveniente solo en la primera iteración, cuando el espacio de estados común equivale al espacio total, ya que al no tener información previa se logra reducir el subespacio aún midiendo sobre un mismo nodo, pero cuando ya se tiene información previa, no se obtienen cambios significativos en las sucesivas iteraciones.

Una vez realizada la medición y extraídas las cotas, éstas son comunicadas a la base donde se arman y trasladan los rectángulos para luego obtener un espacio de estados



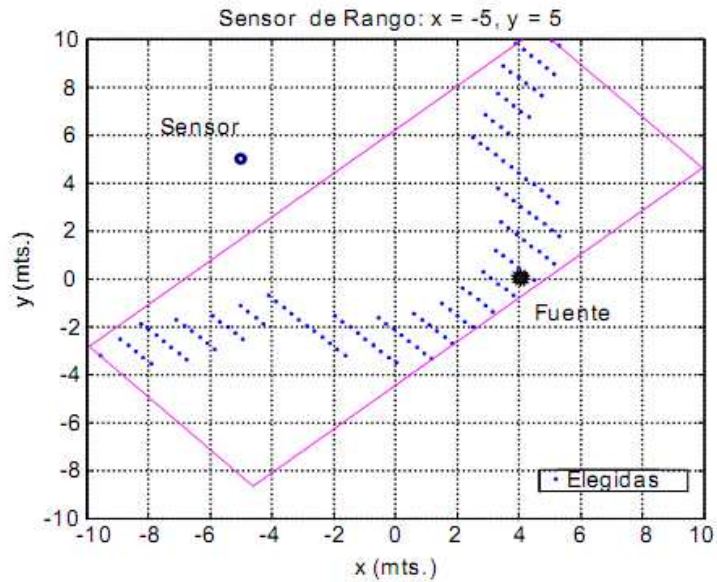
**Figura 4.6:** Estimación del sensor de rango donde se detalla el peso de las partículas dentro del área estimada, en rojo las partículas seleccionadas dentro de la misma.

menor que representa el estimado deseado.

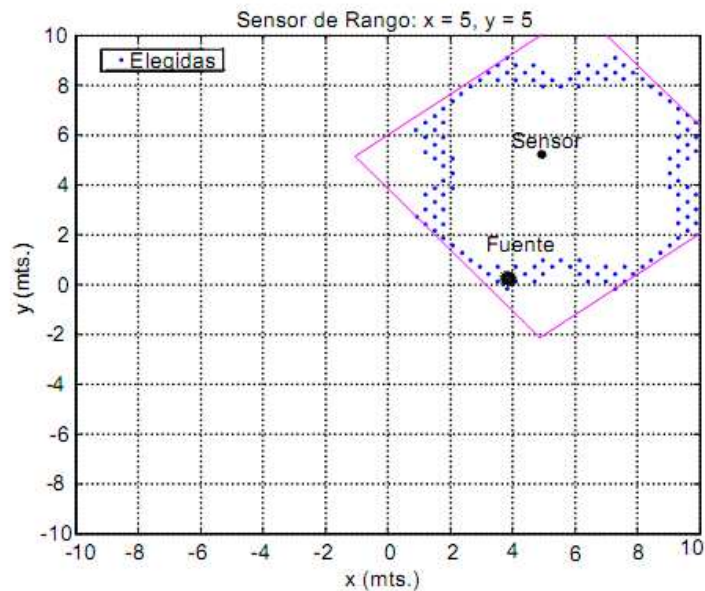
Para entender mejor las figuras a continuación, en la Fig. 4.6 se pueden ver las partículas en el plano XY y sobre el eje Z sus pesos, resultado de una medición de un sensor de rango. Las partículas en rojo son las elegidas por superar  $CotaP$ , y en azul el resto. Se puede ver que el área más probable tiene forma anular.

El resultado de la selección se puede ver en las Fig. 4.7, Fig. 4.8 y Fig. 4.9; donde se pueden apreciar las partículas elegidas en cada nodo sensor y los rectángulos que las encierran formados en la base con las cotas recibidas y llevados a la ubicación correspondiente en el espacio de estados real. Se debe recordar que la base, cuando la red esté en funcionamiento, solo recibirá las cotas; las partículas se comunican y se muestran solo a modo ilustrativo para la simulación. En la práctica, si se debieran comunicar las partículas se requeriría de gran capacidad de comunicación.

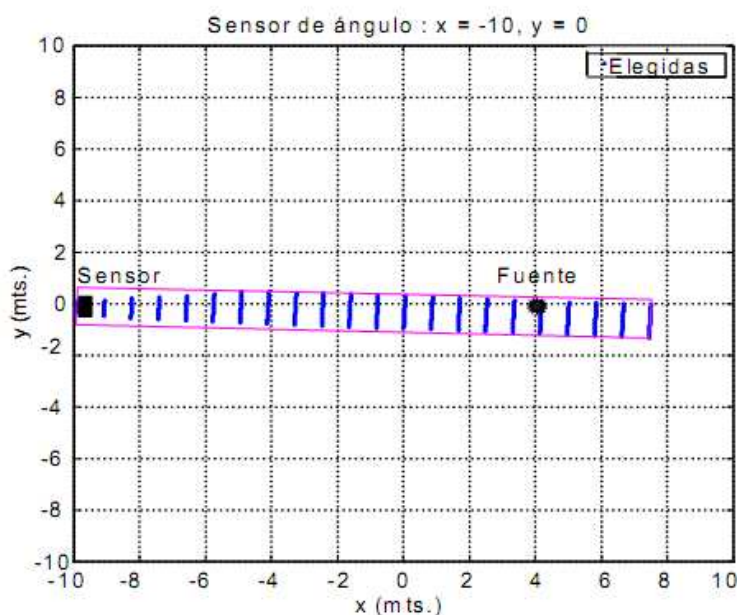
El número de partículas elegidas y la forma del espacio en que se encuentran dependen de la incertidumbre del sensor y la cota de los pesos; si bien  $R$  está dado por el sensor, la cota se debe elegir de modo que siempre se seleccione un mínimo número de partículas para no obrar de sobreconfidente (ver la ecuación (4.7) y Alg. 3). En la



**Figura 4.7:** En azul se representan las partículas seleccionadas y rectángulo magenta representa el subespacio estimado resultante del sensor de rango ubicado en  $[-5 \ 5]$ .



**Figura 4.8:** En azul se representan las partículas seleccionadas y rectángulo magenta representa el subespacio estimado resultante del sensor de rango ubicado en  $[5 \ 5]$ .



**Figura 4.9:** En azul se representan las partículas seleccionadas y rectángulo magenta representa el subespacio estimado resultante del sensor de ángulo ubicado en  $[-10\ 0]$ .

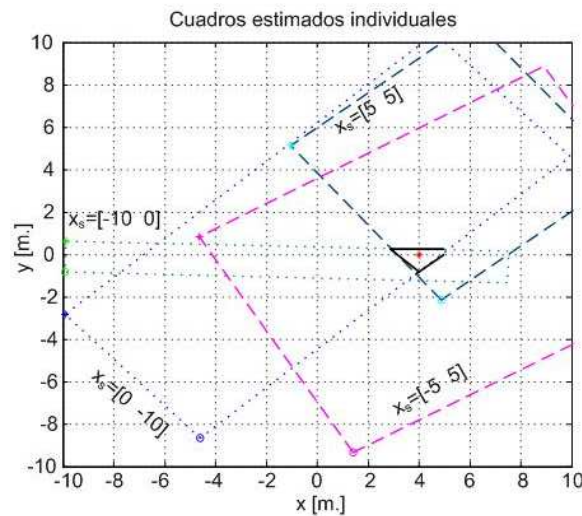
simulación se puede observar que el objeto se halla dentro del rectángulo estimado; entre las partículas seleccionadas. En el caso presentado la cota se eligió basándose en un número determinado de partículas supervivientes,  $N_{min}$ . La cota así elegida resulta diferente para cada nodo.

En la Fig. 4.10 se ve el resultado de las estimaciones de los nodos; en línea llena se marcó la intersección de los rectángulos, se puede observar que el objeto está contenida en ella.

## Conclusiones

Como se pudo apreciar en las simulaciones, el espacio de estados estimado resultante depende de muchos factores. Algunos de ellos tienen que ver con la definición propia del problema, como por ejemplo la incertidumbre en las observaciones. Otros, son parámetros de ajuste propios del filtro BPF como  $CotaP$  y  $N_s$ . El algoritmo del BPF fue implementado en forma simple con el fin de demostrar algunas particularidades de su funcionamiento. Especialmente cómo influyen las cotas o la generación de partículas





**Figura 4.10:** En negro se presenta el espacio de estados resultante de la fusión (intersección) de los estimados de los sensores.

en el espacio resultante. Así, en el caso de observaciones de rango y ángulo como el presentado, las cotas rectangulares, aunque simples de determinar, sobredimensionan la incertidumbre resultante. Si se toman por ejemplo observaciones únicamente de rango con ruido aditivo Gaussiano, la pdf resultante será anular en el espacio cartesiano y la región rectangular que acota encerrará zonas de alta probabilidad con otras muy bajas, conteniendo incluso al sensor (Fig. 4.8). En este caso sería más natural acotar en coordenadas polares. Para reducir al mínimo la comunicación y el procesamiento es posible transmitir de nodo a nodo sólo las cotas de la región más probable. Otro elemento importante es entonces la generación de las partículas en las regiones definidas por las cotas, resultado de una estimación previa. Una opción simple y de baja carga de cómputo es generar partículas sobre una grilla. Sin embargo es evidente que generarlas con una distribución uniforme es preferible ya que no dependen de otros parámetros como el paso de la grilla, etc. Es posible mejorar la aproximación de la pdf *a priori* si además de las cotas se transmite algún tipo de información adicional como los valores más probables.



### 4.2.3. Filtro de Partículas Acotado en Coordenadas Polares

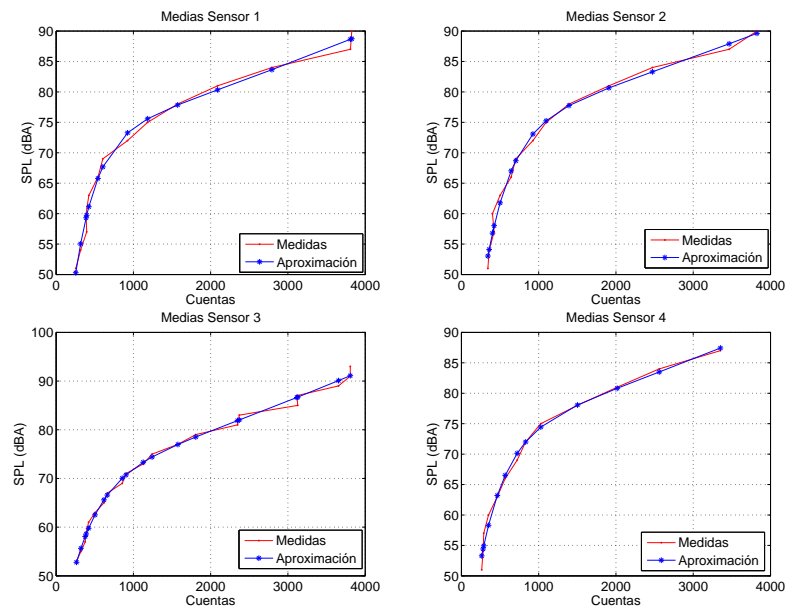
A continuación se presentan dos aplicaciones de localización y seguimiento. La primera de una fuente sonora a partir de sensores de presión acústica donde se instalaron los sensores en una red de nodos distribuida en un ambiente exterior. La segunda de una fuente de alta reflectividad a partir de sensores laser donde se instalaron los sensores en una red de nodos distribuida en un ambiente interior.

El objetivo es analizar el desempeño del BPF en estas aplicaciones y evaluar su comportamiento ante variaciones en los parámetros de ajuste del filtro. La estimación resultante será en forma de cotas en coordenadas polares que, como se muestra en la sección anterior, es más adecuado para representar la varianza real.

Se estudiará el impacto de los parámetros del BPF a partir de un análisis estadístico de diversos casos y aplicaciones, cómo la distribución de los nodos, las características del modelo y de las observaciones, esto es si éstas son de rango o ángulo o ambos. En todos los casos se transmiten solo las cotas y las partículas se generan en el nuevo nodo con una distribución uniforme con pesos iguales  $w_i = 1/N_s$ .

#### Mediciones en un Ambiente Exterior. Localización de una Fuente Sonora.

Se configuró una red de sensores con nodos MicaUNS y cada nodo fue conectado a un Sensor de Nivel de Presión de Sonora diseñado y elaborado en la Universidad Nacional del Sur. La observación se basa en la medición de señales de nivel de presión sonora (dBA) utilizando micrófonos omnidireccionales para sensar la señal de una fuente de sonido consistente en un tono de 200Hz. y se utilizó en los ensayos. El sensor de nivel de presión sonora (dBA) fue diseñado para localizar una fuente emisora de una señal acústica de 200 Hertz. Los sensores fueron previamente testeados y calibrados. En la Fig. 4.11 se ve la curva medida y la aproximación de cada sensor; mas detalles se pueden encontrar en [Sondon et al. (2007a)] y [Sondon et al. (2007b)]. También se realizó una caracterización de la fuente sonora y ambos modelos se combinaron para obtener aquel que relaciona los dBA con el rango.



**Figura 4.11:** Curvas medidas de los sensores dBA y su aproximación. Cuentas es el valor medido con el conversor analógico-digital del nodo MicaUNS y dBA es la presión sonora emitida por una fuente medida con un micrófono patrón.

Se considera que las observaciones responden a un modelo de ruido aditivo Gaussiano y de media cero. Los pesos se calculan a partir de la estimación de la observación con este modelo y de la observación misma. Con estos se seleccionan las partículas mas pesadas calculando la  $CotaP$  como un porcentaje del peso máximo.

Una vez obtenida la medición en dBA se calculan los pesos y se seleccionan las partículas mas pesadas. La  $CotaP$  se calcula a partir del peso máximo; a partir de este se toman dos desviaciones ( $R$  en el modelo de observación) y se establece la cota en consecuencia.

Como ya se analizó en la Sección 4.1 una buena estimación inicial de la  $CotaP$  es del 70% del peso máximo, siendo esto función de la precisión de los sensores. En caso de que el número total de partículas seleccionadas sea muy bajo (menor a un mínimo número de partículas sobrevivientes,  $N_{min}$ ), se reduce la cota de peso y se vuelve a realizar la comparación, de modo que el estimado sea representativo del estado.  $N_{min}$  se elige en función del número mínimo de partículas a partir del cual habría una pérdida severa de información o no se lograría representar correctamente la pdf [Carpenter et al.

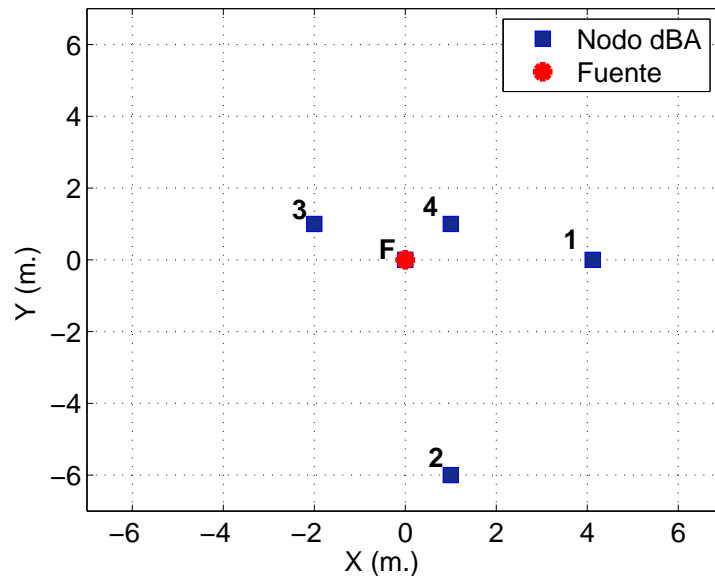
**Cuadro 4.1:** Ubicación de los nodos y la fuente para localización en exteriores. Se tomo como origen de coordenadas la posición de la fuente.

ID Nodo	Posición X (m.)	Posición Y (m.)
F	0	0
1	4.12	0
2	1	-6
3	-2	1
4	1	1

(1999)]. En este caso,  $N_{min}$  se elige de 15 partículas debido a la incertidumbre de los modelos y los sensores, y debido a que en ensayos previos se observó que 15 partículas son representativas del estado. Los sensores de nivel de presión sonora dan información de rango del nodo a la fuente, es por ello que las áreas que contienen a las partículas elegidas serán anulares. Cada anillo estimado por cada sensor es interceptado por el siguiente y define un nuevo espacio de estados. En esta aplicación de localización la estimación se va acotando en forma consecutiva a medida que llega una nueva medición; el algoritmo finaliza cuando la estimación tiene la calidad aceptable o no hay mejoras en la incertidumbre. Para este análisis, una vez resuelta la estimación, cuando se llega al punto en que el área estimada no varía, se puede continuar estimando sobre dicha área o bien, comenzar una nueva corrida partiendo del espacio total de estados. En un problema de seguimiento se debe actualizar el área estimada con el modelo de movimiento antes de comenzar la estimación.

Los sensores se ubicaron como muestra la Fig. 4.12; los números representan la identificación (ID) de los nodos y su ubicación se detalla en la Tabla 4.1. Como fuente (F) se utilizó un parlante con un tono de 200 Hertz.

Las mediciones experimentales en un ambiente exterior abierto se procesaron para reducir los efectos de las ráfagas de viento en los micrófonos. El espacio de estados a estimar es el espacio X-Y. Las mediciones llegan a la computadora y son almacenadas en una matriz, el orden de arriba es el orden en que el algoritmo procederá a realizar las estimaciones fuera de línea. El algoritmo toma los datos de un nodo, genera las partículas dentro de las cotas que corresponda (si es la primer medición o el comienzo de una nueva corrida el espacio total, sino el espacio formado por aquellas que encierran



**Figura 4.12:** Configuración de la SN para localización de una fuente sonora de 200 Hz, en un espacio de estados de 2D. Con un cuadrado azul se representa la posición de los nodos de la red de sensores y con un círculo rojo la de la fuente.

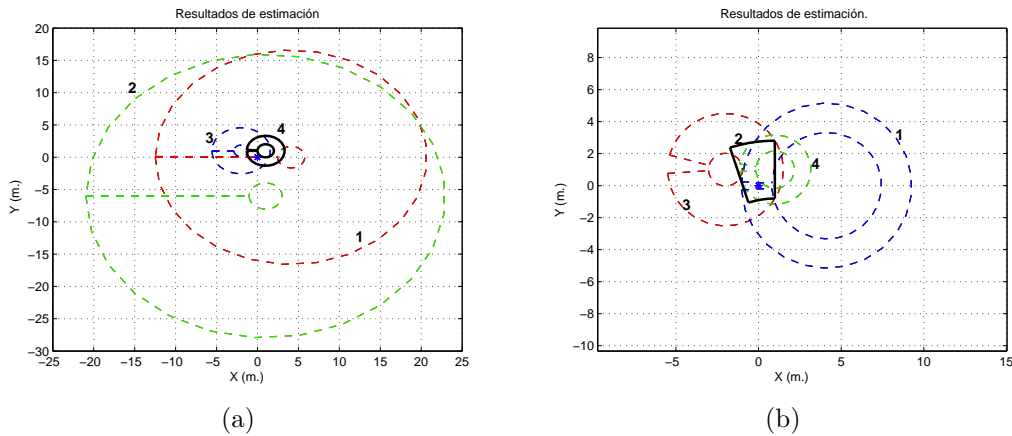
el estimado anterior), calcula los pesos de las partículas y *CotaP*, realiza la selección de partículas conservando las mas pesadas y extrae las cotas que las encierran. Una vez obtenidas, estas se utilizan como espacio de estados inicial para la siguiente medida de nodo que arriba a la computadoras.

En la Fig. 4.13 se pueden ver los resultados de la localización, cada nodo realiza una estimación y la comunica al siguiente, la tercera es la que se considera estimación resultante (en la figura en línea llena). Los sensores de nivel de presión sonora dan información de rango del nodo a la fuente, es por ello que las áreas que contienen a las partículas elegidas son anulares. Cada anillo estimado por cada sensor es interceptado por el siguiente y define un nuevo espacio de estados. En un problema de localización la estimación se va acotando en forma consecutiva a medida que llega una nueva medición; el algoritmo finaliza cuando la estimación tiene la calidad aceptable o no hay mejoras en la incertidumbre. Para este análisis, una vez resuelta la estimación, cuando se llega al punto en que el área estimada no varía, se puede continuar estimando sobre dicha área o bien, comenzar una nueva estimación, iniciando en el espacio total de estados.

En un problema de seguimiento se debe actualizar el espacio de estados inicial con el modelo de movimiento antes de comenzar la estimación hasta lograr un resultado aceptable.

En línea discontinua se muestran las estimaciones consecutivas hasta la resultante donde el número cercano a cada área representa la ID del nodo que la realizó. El orden de los sensores que realizaron las estimaciones en cada corrida es 2-1-3-4 (Fig. 4.13(a)) y 1-3-4-2 (Fig. 4.13(b)). Como se puede apreciar en forma cualitativa, la forma y tamaño final del área resultante estará relacionada con el orden en el que se procese la información.

Es posible incorporar un algoritmo que seleccione el orden en el que se procesa la información (próximo nodo) en función de parámetros tales como la medida que incorpora mayor información, u otra similar [Kaplan (2006)]. Esto sin embargo está fuera del objetivo de esta tesis y por lo tanto las simulaciones presentan secuencias arbitrarias.



**Figura 4.13:** Estimación secuencial de la ubicación de la fuente (asterisco), (a) en orden 2-1-3-4 y (b) orden 1-3-4-2 de los nodos. El estimado final se presenta en línea completa y en línea discontinua las estimaciones intermedias.

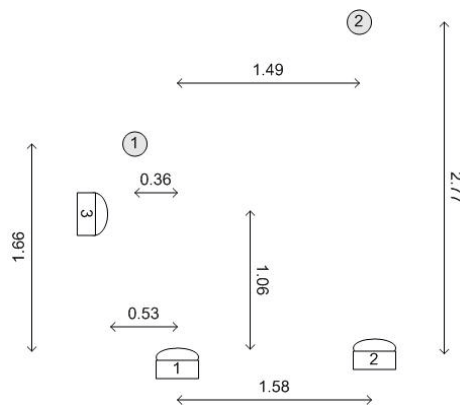
### Mediciones en un Ambiente Interior. Localización y Seguimiento de un Objeto.

Se realizaron una serie de ensayos con el objeto de analizar el desempeño de el BPF ante variaciones de algunos de los parámetros de ajuste. Para esto se plantean

tanto el problema de localización como de seguimiento en ambientes interiores. Para el primer caso se presenta un análisis del tiempo de procesamiento del algoritmo. Se utilizó como sensor en los nodos un escáner láser y para los análisis se empleó únicamente la información de rango. Los objetos a localizar y/o seguir eran conos con cinta reflectiva de modo que sean fácilmente distinguibles a partir del nivel de reflectancia. La fusión nodo a nodo se realiza fuera de línea en una computadora.

**Localización.** En una primera instancia se realizó un ensayo de localización en un ambiente interior. Se ubicaron dos objetos a localizar y el láser en la forma que muestra la Fig. 4.14.

La Tabla 4.2 muestra la ubicación de los nodos y los objetos a localizar con el *Nodo 1* como origen de coordenadas; con el objeto de poder comparar los resultados del algoritmo y corroborar que la solución contiene al objeto real.



**Figura 4.14:** Ubicación de los objetos a localizar (círculos) y sensores láser en el ensayo de localización en un ambiente interior. Las medidas están dadas en metros.

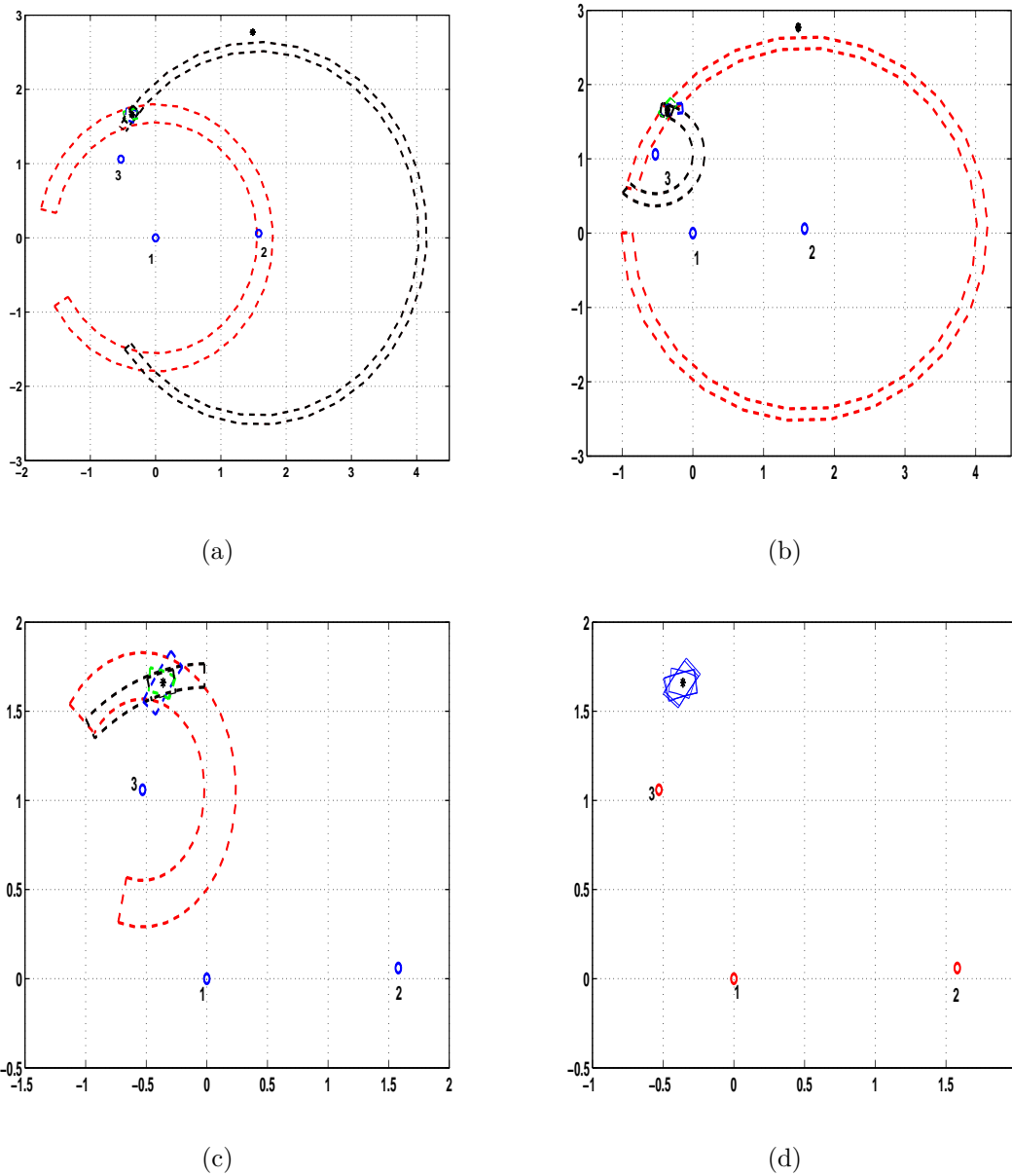
Una vez analizadas las mediciones en diferentes posiciones se eligieron las que pertenecen a las posiciones 1, 2 y 3 del láser debido a que aportan mejores resultados de reflectancia de los objetos a localizar. Se dividieron las mediciones por objeto. Las Fig. 4.15 y Fig. 4.16 presentan las estimaciones del *Objeto 1* y *Objeto 2* respectivamente, utilizando el BPF. En ambas se pueden ver las estimaciones consecutivas

Nodo	X (m.)	Y (m.)
1	0	0
2	1.58	0.06
5	-0.53	1.06
Objeto 1	-0.36	1.66
Objeto 2	1.49	2.77

**Cuadro 4.2:** Ubicación de los Sensores y los objetos a localizar respecto del Nodo 1

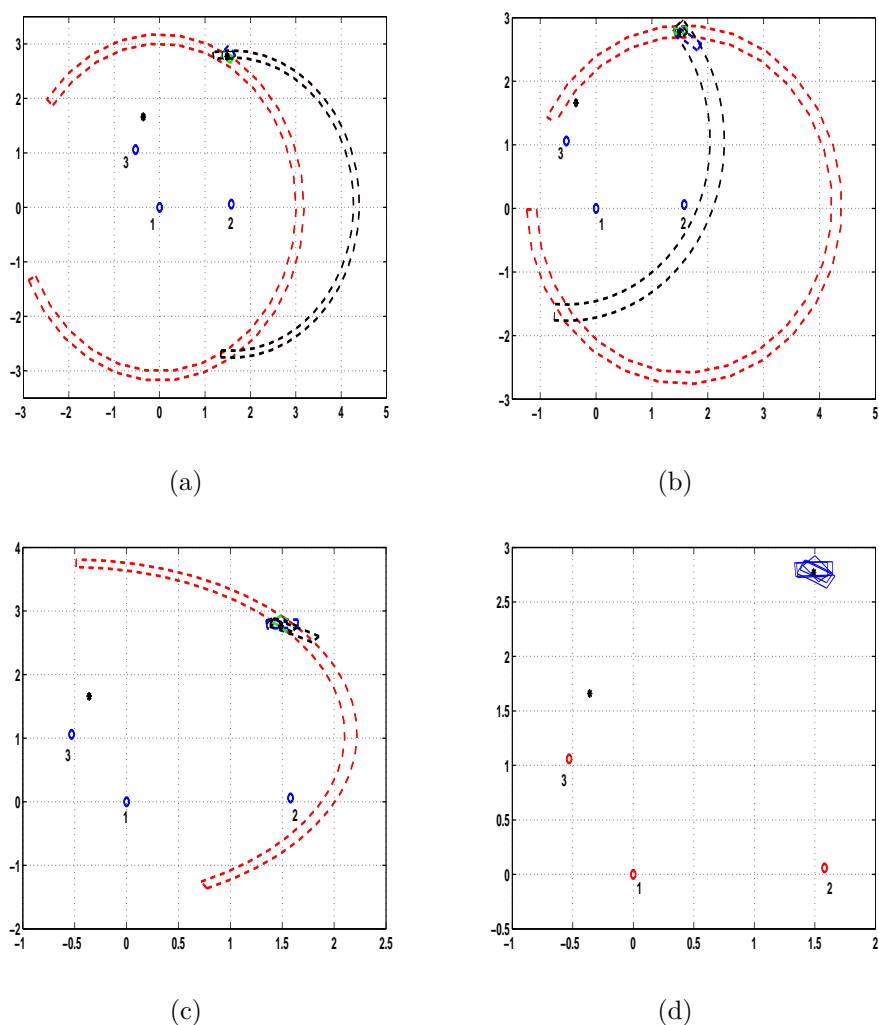
(línea discontinua) hasta llegar a una estimación final por corrida (línea completa). Para apreciar el efecto que tiene la selección del nodo inicial (y posiblemente el orden de estimación), en las Fig. 4.15 (a) a (c) se muestra una corrida iniciándola en cada uno de los sensores. El ultimo gráfico muestra las tres estimaciones finales juntas. Como se puede apreciar cualitativamente, todas las soluciones son similares y contienen a la fuente real.

De ambas figuras es interesante resaltar el funcionamiento cualitativo del algoritmo. Tomando por ejemplo la Fig. 4.15, la primera medición del nodo genera una región anular rodeándolo y conteniendo la fuente. En la figura esta región se observa como un sector anular ya que el área inicial donde existían partículas era el rectángulo definido por  $[-3,-4]$  y  $[5,4]$ . Estas cotas se pasan al nodo 2 quien genera partículas en forma uniforme dentro de ellas. El resultado al procesar la medición son dos áreas simétricas, una donde esta la fuente y otra en  $[1.5,-2.5]$  aproximadamente. Como la cota resultante debe encerrar ambas hipótesis se genera un sector anular como se ve en la figura. Finalmente, el nodo 3 genera partículas en este sector y al pesar por su medida se obtiene el sector final que contiene a la fuente solamente. La misma progresión puede verse en la Fig. 4.15 (b), la Fig. 4.16 (a) y (b) pero no en la Fig. 4.15 (c) y la Fig. 4.16 (c) donde por la geometría de la ubicación de los nodos y la fuente, no existe la ambigüedad en la segunda medida.



**Figura 4.15:** Localización del *Objeto 1* utilizando el BPF. En círculos rojos se representan los nodos sensores y con asteriscos negros los objetos a localizar; la estimación final se presenta en línea azul completa y las estimaciones consecutivas anteriores a la final en trazo de guiones rojo. En (a) iniciando la estimación en el nodo 1, en (b) en el 2 y en (c) en el 3. En (d) se muestran las estimaciones finales de cada corrida secuencial.





**Figura 4.16:** Localización del *Objeto 2* utilizando el BPF. En círculos azules se representan los nodos sensores y con asteriscos negros los objetos a localizar; la estimación final se presenta en línea azul completa y las estimaciones consecutivas anteriores a la final en trazo de guiones rojo. En (a) iniciando la estimación en el nodo 1, en (b) en el 2 y en (c) en el 3. En (d) se muestran las estimaciones finales de cada corrida secuencial.

**Análisis de los resultados** Se realizaron ensayos (corridas) variando los parámetros detallados al principio de la sección de modo de ver la influencia de los parámetros del BPF; como ser el número de partículas inicial ( $N_s$ ), el número mínimo de partículas sobrevivientes ( $N_{min}$ ), el porcentaje del peso máximo ( $P$ ), además del efecto del número estimaciones consecutivas. La intención es, a partir de dichos ensayos, proponer una estrategia para la elección de los valores óptimos para el buen desempeño del BPF.

Esta práctica puede ser implementada para sintonizar el funcionamiento del BPF en una red de sensores, para una aplicación y estructura de red determinadas.

En la Tabla 4.3 se detalla en el orden que se grafican los resultados de cada ensayo.

El número mínimo de partículas ( $N_{min}$ ) es como se dijo el número mínimo de aquellas que representa la distribución. Este número no es posible determinarlo sin conocer exactamente la distribución. Se presenta entonces este análisis fuera de línea para tener una idea su efecto y valor. Esto implica no tener restricciones de cálculo o almacenamiento, y es posible generar muchas más partículas que las necesarias.  $CotaP$  se obtiene como un porcentaje del peso máximo de todas las partículas y con esta cota se seleccionan las sobrevivientes y comparan con  $N_{min}$ . En el caso que el número sea mayor o igual, se obtienen las cotas con aquellas sobrevivientes y se pasa a la siguiente estimación o corrida según corresponda. En el caso que no se alcance  $N_{min}$ , se reduce la  $CotaP$  para lograr ese número mínimo.

El número mínimo de partículas ( $N_{min}$ ) se fue variando para observar hasta que punto el objeto de interés es contenido totalmente en la estimación (dentro de las cotas de la estimación final). El número de estimaciones consecutivas se eligió experimentalmente evaluando el estimado resultante, 3 fue el número de iteraciones del algoritmo que aportaron una estimación final aceptable, con 5 estimaciones consecutivas el resultado mejora, y con 7 estimaciones consecutivas se observó cualitativamente que no había una gran mejora respecto a los resultados con 5 iteraciones. Esto es natural dado que se está básicamente triangulando medidas de rango en dos dimensiones. Para formalizar el comportamiento del BPF ante las variaciones que muestra la Tabla 4.3, para la red y aplicación ya detalladas, a continuación se muestran los resultados para 500 corridas del algoritmo en cada ensayo.

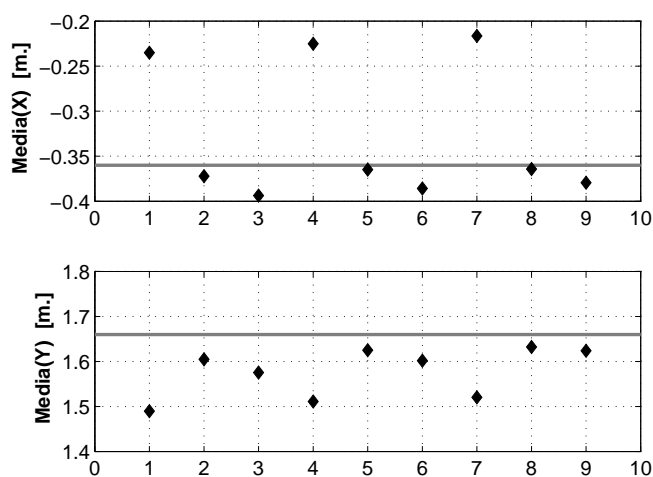
Los primeros ensayos se realizaron generando 1000 partículas ( $N_s$ ) al comienzo de cada estimación y los resultados se muestran en las Fig. 4.17, 4.18 y 4.19. El eje  $X$  de los gráficos representa el número de ensayo.

En la Fig. 4.17, la Fig. 4.20 y la Fig. 4.23 se muestra una comparación entre la posición real en  $X$  e  $Y$  en línea llena y para cada ensayo el promedio de los 500 valores

Ensayo N°	N° Mín de Part.	N° Estimac. consecutivas
1	5	3
2	5	5
3	5	7
4	10	3
5	10	5
6	10	7
7	15	3
8	15	5
9	15	7

**Cuadro 4.3:** Correspondencia entre Barras y Rombos con cada Ensayo

medios de las partículas seleccionadas en la última estimación de cada corrida, representados con rombos. En la Fig. 4.18, la Fig. 4.21 y la Fig. 4.24 se grafica, con una barra por cada ensayo, el promedio de la Desviación Estándar (STD: Standard Deviation) de las 500 corridas calculada a partir de la Desviación Estándar de las partículas que conforman la estimación resultante de cada corrida.



**Figura 4.17:** En las ordenadas se representa el número de ensayo, y en las abscisas la media de las partículas resultantes en estimaciones finales con 1000 partículas iniciales en [m.]. Rara cada ensayo se representa con rombos el promedio de los 500 valores medios de las partículas seleccionadas en la última estimación de cada corrida. En línea llena se grafica el valor real.

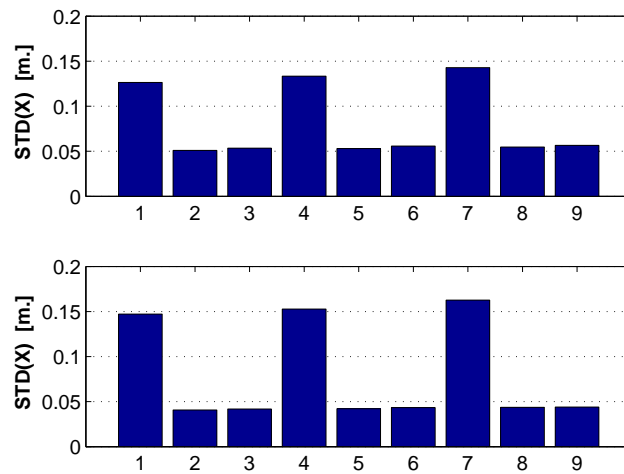
En la figura fig. 4.17, donde se presenta la posición real del objeto y su estimación, se puede apreciar que para un número de iteraciones mayor a 5 el error promedio entre la estimación y su valor real es siempre menor al 10 % (ensayos 2,3,5,6,8,9). En base a esto, se puede concluir que 5 es un número mínimo de iteraciones necesarias para este ejercicio de localización. Entre 5 y 7 iteraciones, hay un incremento leve en el error, y esto se debe al orden en que se procesa la información y a la geometría del problema. De la Fig. 4.17 se puede concluir que los ensayos que más se acercan al objeto son los de 5 iteraciones (ensayos número 2, 5 y 8 sobre el eje X).

La medida de *Área* que se calcula representa la calidad de la estimación resultante, en este caso relacionada con el tamaño. No es una medida convencional, sino que se define en esta tesis como *Área* al producto de las diferencias entre las cotas de ángulo y rango, y se calcula con (4.13). Esta es una medida mas conveniente para comparar que un área en metros cuadrados debido a la forma en que se toman las cotas.

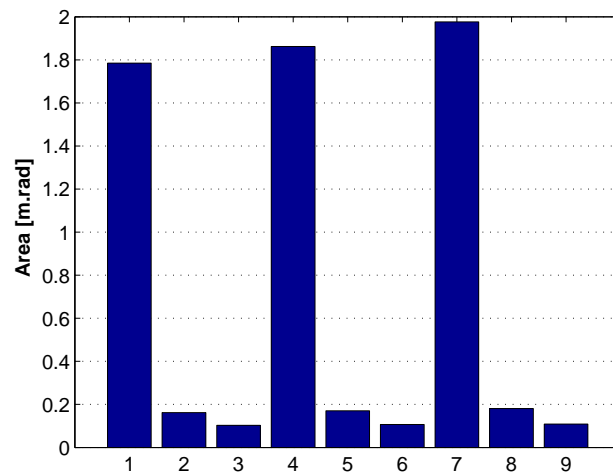
$$\text{Área} = (\text{Rango Máx} - \text{Rango Mín}) \times (\text{Ángulo Máx} - \text{Ángulo Mín}).[\text{m.rad}] \quad (4.13)$$

En las Fig. 4.18 y Fig. 4.19 se puede observar claramente que el STD y el *área* se reducen a medida que aumenta el número de iteraciones dentro de cada corrida. El cambio más importante es entre 3 y 5 iteraciones. En este caso particular de una estimación en dos dimensiones a partir de la información de rango, se podría seleccionar como condición de parada del algoritmo las 5 iteraciones, pues ya no se producen grandes cambios al aumentarlas. Esto es relevante dado que cuanto menos iteraciones se realicen, mas rápido se llega al estimado resultante y se consume menor potencia en los nodos; de ahí que es importante poder establecer un número óptimo de las mismas.

Algo mas a notar es que a medida que el Mínimo Número de Partículas Sobrevivientes aumenta, aumenta el STD y el *área* (ensayos 1,4 y 7 por ejemplo, en las figuras Fig. 4.18 y Fig. 4.19); lo que es razonable debido a que al aumentar  $N_{min}$  se esta modelando una reducción de la precisión del sensor; por esa razón aumenta el *área* estimada y el STD.



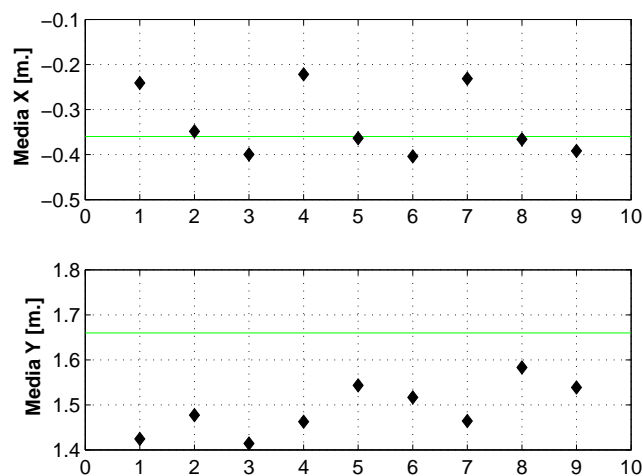
**Figura 4.18:** En las ordenadas se representa el número de ensayo, y en las abscisas la varianza STD de las partículas resultantes en estimaciones finales en [m.] con 1000 partículas iniciales. Resultado de promediar la varianza STD a de 500 corridas en cada ensayo.



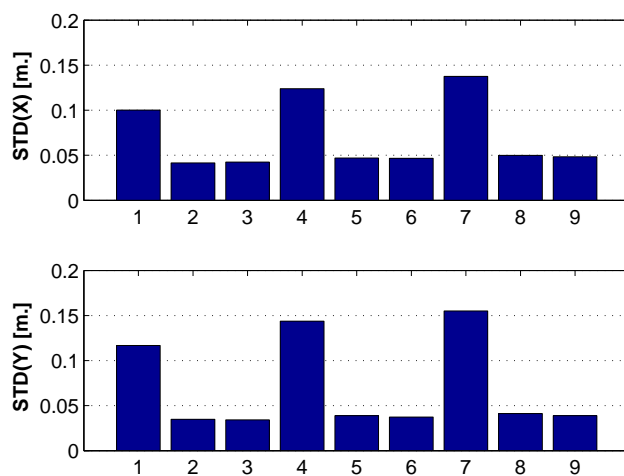
**Figura 4.19:** El eje X corresponde al número de ensayo, y las barras al *área* aproximada de las estimaciones finales con 1000 partículas iniciales [m.rad]. Resultado de promediar el *área* de 500 corridas en cada ensayo.

Se realizó luego el mismo procedimiento pero generando 500 y 200 partículas al comienzo de cada estimación y los resultados se muestran en las Fig. 4.20, 4.21 y 4.22 para 500 partículas iniciales y en las Fig. 4.23, 4.24 y 4.25 para 200 partículas iniciales. Se pueden ver en cada caso los mismos efectos observados para 1000 Partículas Iniciales.

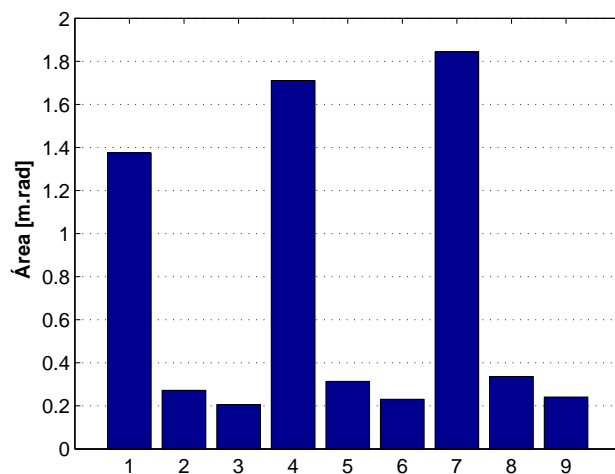
De las Fig. 4.20 y Fig. 4.23 se deduce lo mismo que para los ensayos de 1000 partículas generadas inicialmente; aquellos con 5 iteraciones mínimo aproximan suficientemente la posición real del objeto. Como se puede apreciar en las figuras, no hay gran mejora en hacer 7 iteraciones en lugar de 5; esto se puede observar en las gráficas de STD y de *área*; sin embargo hacer menos iteraciones permite llegar a la estimación en menos tiempo.



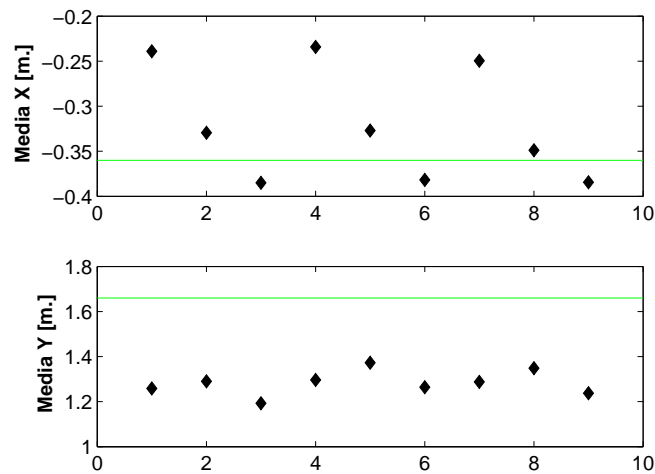
**Figura 4.20:** En las ordenadas se representa el número de ensayo, y en las abscisas la media de las partículas resultantes en estimaciones finales con 500 partículas iniciales en [m.]. Rara cada ensayo se representa con rombos el promedio de los 500 valores medios de las partículas seleccionadas en la última estimación de cada corrida. En línea llena se grafica el valor real.



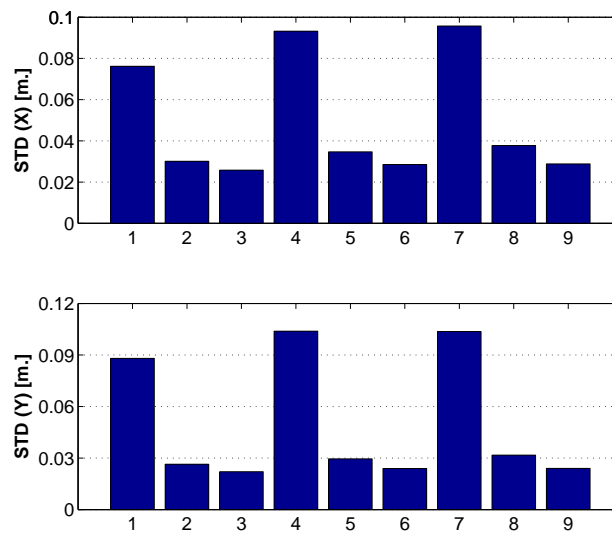
**Figura 4.21:** En las ordenadas se representa el número de ensayo, y en las abscisas la varianza STD de las partículas resultantes en estimaciones finales en [m.] con 500 partículas iniciales. Resultado de promediar la varianza STD a de 500 corridas en cada ensayo.



**Figura 4.22:** El eje X corresponde al número de ensayo, y las barras al *área* aproximada de las estimaciones finales con 500 partículas iniciales [m.rad]. Resultado de promediar el *área* de 500 corridas en cada ensayo.

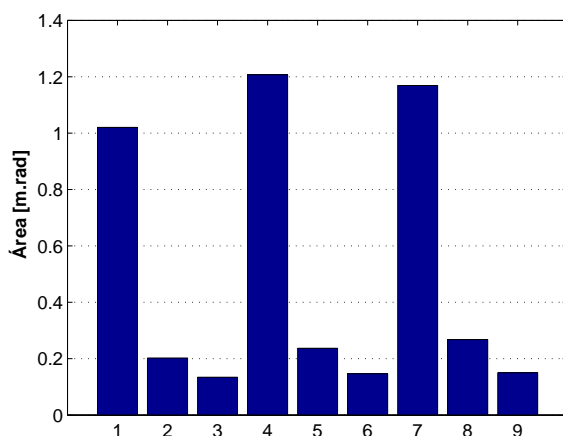


**Figura 4.23:** En las ordenadas se representa el número de ensayo, y en las abscisas la media de las partículas resultantes en estimaciones finales con 200 partículas iniciales en [m.]. Para cada ensayo se representa con rombos el promedio de los 500 valores medios de las partículas seleccionadas en la última estimación de cada corrida. En línea llena se grafica el valor real.



**Figura 4.24:** En las ordenadas se representa el número de ensayo, y en las abscisas la varianza STD de las partículas resultantes en estimaciones finales en [m.] con 200 partículas iniciales. Resultado de promediar la varianza STD a de 500 corridas en cada ensayo.





**Figura 4.25:** El eje X corresponde al número de ensayo, y las barras al *área* aproximada de las estimaciones finales con 200 partículas iniciales [m.rad]. Resultado de promediar el *área* de 500 corridas en cada ensayo.

A continuación se muestran los resultados de los ensayos para 5 iteraciones del algoritmo en cada corrida, hasta llegar al estimado resultante. En este caso los parámetros que se variaron fueron el número inicial de partículas ( $N_s$ ), y el número mínimo de partículas sobrevivientes ( $N_{min}$ ); según se lista en la Tabla 4.4. El objetivo es establecer un compromiso entre la capacidad de procesamiento, el almacenamiento del nodo y el tiempo versus la precisión de la estimación.

El número inicial de partículas,  $N_s$ , se fue aumentando desde un mínimo que asegura la representación de la pdf buscada en el problema de localización en dos dimensiones. Este se obtuvo a partir de las experimentaciones al igual que el máximo que se determinó cuantitativamente a partir de cómo era mejorado el estimado con cada variación.

El número mínimo de partículas se estableció en función del resultado de la estimación y si esta contenía o no al objeto de interés. Un número muy bajo puede no representar la pdf buscada aunque contenga al objeto [Carpenter et al. (1999)]. Por eso aquí el compromiso es elevar  $N_{min}$  para lograr una mejor representación contra el

Ensayo N°	N° de Part. Inic. $N_s$	N° Mín de Part. $N_{min}$
1	200	5
2	500	5
3	1000	5
4	200	10
5	500	10
6	1000	10
7	200	15
8	500	15
9	1000	15

**Cuadro 4.4:** Correspondencia entre Ensayos y el parámetro a variar en cada uno. Su ubicación en el eje X de los gráficos que siguen esta dada por el número de ensayo. Los ensayos se realizaron para 5 iteraciones en cada corrida del BPF.

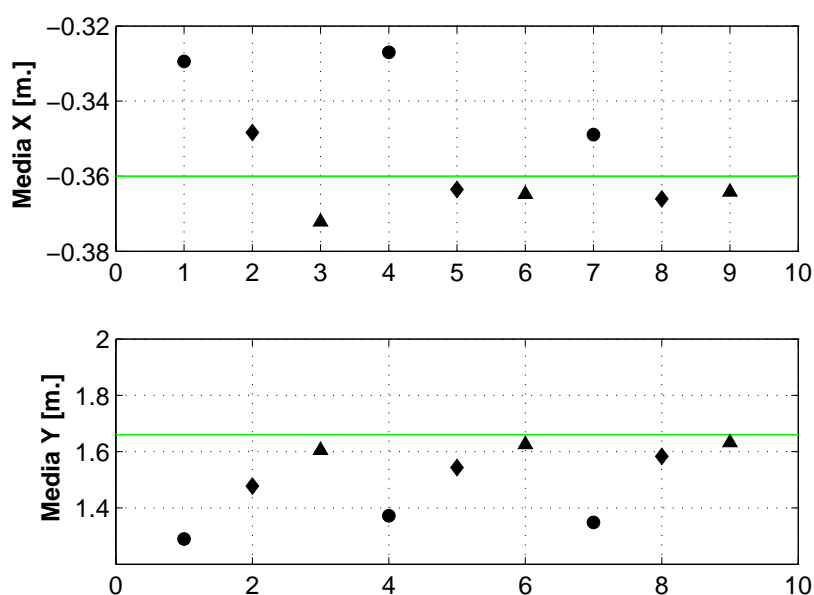
tiempo en el que se obtiene el estimado.

En la Fig. 4.26 se pueden ver en círculos, rombos y triángulos los ensayos para un Número Mínimo de Partículas Sobrevivientes en la etapa de selección de 5; 10 y 15 respectivamente. Cada una de estas marcas representa los promedios de las medias de las partículas que forman las estimaciones resultantes en cada una de las 500 corridas. Para referencia se muestra con una línea punteada la posición real del objeto. También se puede apreciar en la figura que al aumentar  $N_s$  se logra una mejor aproximación que es parte del compromiso buscado.

También se puede apreciar observando en conjunto a las Fig. 4.26 y Fig. 4.27, que al ser  $N_s$  mas grande se logra una mejor estimación pero a la vez la varianza resultante es mayor. Esto se debe a que al haber más número de partículas disponibles, un número también superior sobreviven al umbral y como consecuencia aumenta el área que las encierra. El valor mas bajo de  $N_{min}$  (cinco partículas) permite lograr un resultado sin problemas de empobrecimiento de muestras y una buena localización del objeto. Es importante recalcar que esto se debe fundamentalmente a la precisión del sensor y por ende a la varianza de la medición; que en este caso es pequeña.

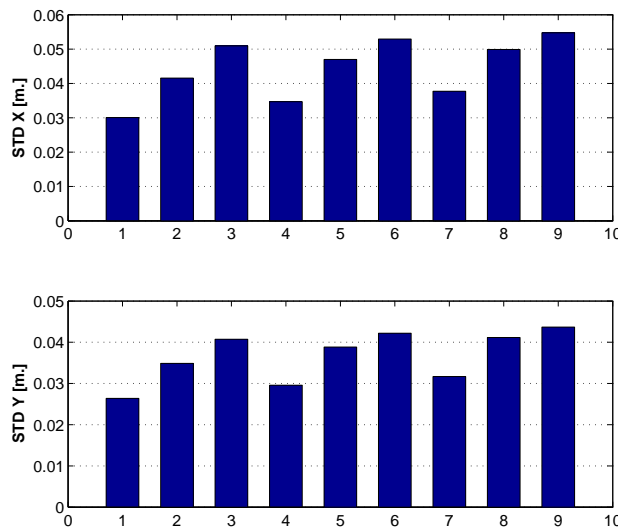
Media		Std		Área apróx. [m.rad]
X [m.]	Y[m.]	X [m.]	Y[m.]	
-0.34835	1.4775	0.041565	0.034882	0.27258

**Cuadro 4.5:** Resultados para el ensayo con los parámetros más convenientes, la generación inicial de 500 partículas, con 5 iteraciones consecutivas y 5 partículas como el mínimo a conservar.



**Figura 4.26:** En las ordenadas se representa el número de ensayo, y en las abscisas la media de las partículas resultantes en estimaciones finales luego de 5 iteraciones, en [m.]. Rara cada ensayo se representa con rombos el promedio de los 500 valores medios de las partículas seleccionadas en la última estimación de cada corrida. En línea llena se grafica el valor real. Se representan con círculos los ensayos para un Número Mínimo de Partículas Sobrevivientes de 5; con rombos pertenecen a 10 Partículas Sobrevivientes Mínimo y con triángulos a 15 Partículas Sobrevivientes Mínimo en la etapa de selección.

Una buena selección de parámetros para la aplicación presentada y teniendo en cuenta lo visto hasta el momento sería la generación inicial de 500 partículas, con 5 iteraciones consecutivas y 5 partículas como el mínimo a conservar. Para este caso se obtienen las soluciones que se muestran en la Tabla 4.5.

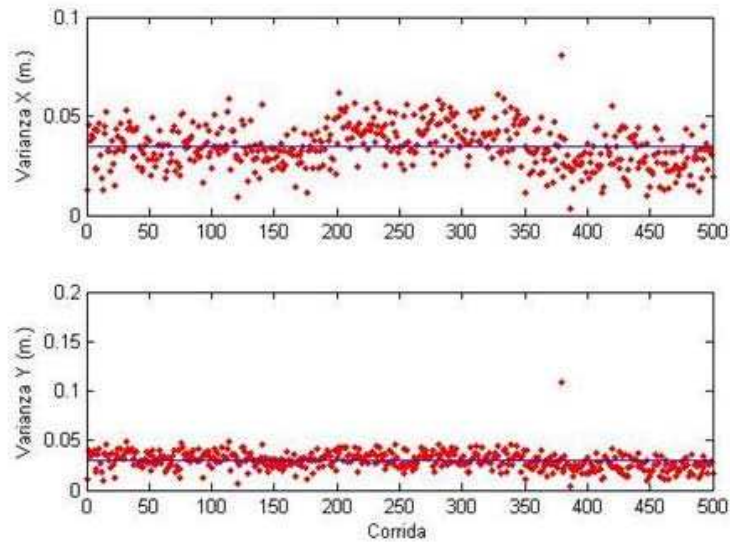


**Figura 4.27:** En las ordenadas se representa el número de ensayo, y en las abscisas la varianza STD de las partículas resultantes en estimaciones finales luego de 5 iteraciones, en [m.]. Resultado de promediar la varianza STD a de 500 corridas en cada ensayo.

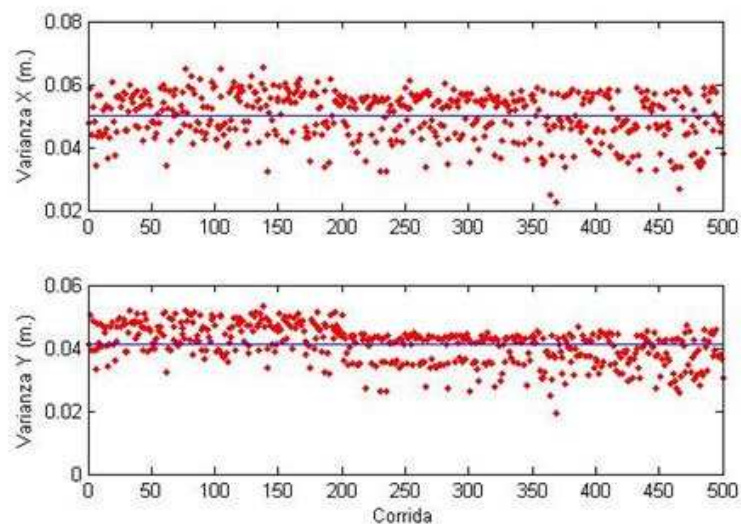
Es importante recalcar, como se aclaró al principio de este análisis, que la selección y estudio de los parámetros del BPF solo sirven para la presente aplicación. Una parte esencial de la sintonía para su funcionamiento es seguir este procedimiento antes de dejar la red operativa. El objetivo es explotar en forma óptima la potencialidad de la red, así como posiblemente la ubicación de los nodos, la selección de los sensores, el terreno, el tráfico de paquetes de RF, etcétera.

Una particularidad importante es como se modifica la varianza de la estimación en las diferentes corridas. Si esta se mantiene cercana a la varianza media en todos los casos, significaría que en todas las corridas se llega a un área resultante similar bajo las mismas condiciones de configuración de la red y del BPF. En la Fig. 4.28 se ve la varianza de las partículas seleccionadas en la estimación resultante de las 500 corridas del BPF, para  $N_s = 500$ , 5 iteraciones consecutivas y  $N_{min} = 10$ . En la Fig. 4.29 se ve lo mismo para  $N_s = 500$ , 5 iteraciones consecutivas y  $N_{min} = 15$  y en la Fig. 4.30 nuevamente el mismo resultado para  $N_s = 1000$ , 5 iteraciones consecutivas y  $N_{min} = 10$ . En las tres figuras se aprecia cualitativamente que, salvo en uno o dos casos

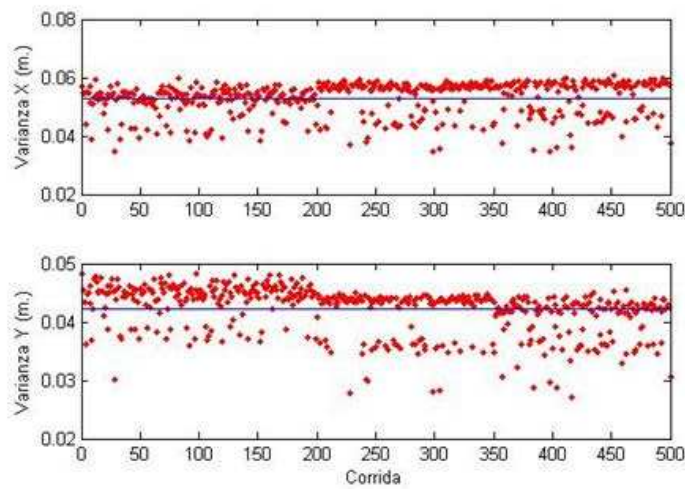
de 500, la varianza esta cercana a la varianza media de las corridas, lo que es deseable.



**Figura 4.28:** En las ordenadas se representa la varianza STD de las partículas resultantes en estimaciones finales en [m.] para  $N_s = 500$ , 5 iteraciones consecutivas y  $N_{min} = 10$ . Resultado de cada una de las 500 corridas del BPF.

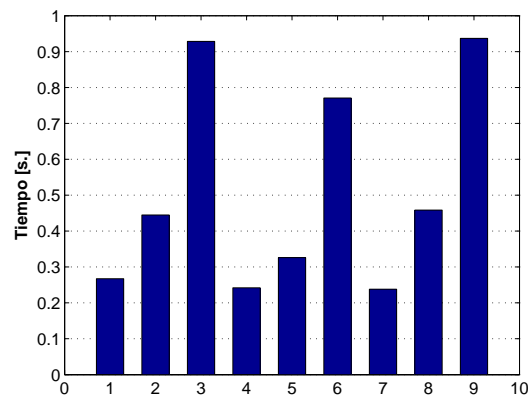


**Figura 4.29:** En las ordenadas se representa la varianza STD de las partículas resultantes en estimaciones finales en [m.] para  $N_s = 500$ , 5 iteraciones consecutivas y  $N_{min} = 15$ . Resultado de cada una de las 500 corridas del BPF.



**Figura 4.30:** En las ordenadas se representa la varianza STD de las partículas resultantes en estimaciones finales en [m.] para  $N_s = 1000$ , 5 iteraciones consecutivas y  $N_{min} = 10$ . Resultado de cada una de las 500 corridas del BPF.

**Análisis de Tiempo en Localización** Para realizar una medida comparativa de tiempos, se realizó una marca temporal cada vez que se terminaba de correr el algoritmo en cada ensayo; en la Fig. 4.31 las barras corresponden a los ensayos descritos en la Tabla 4.4 y revelan diferencias importantes de tiempos que están directamente relacionadas al número de partículas generadas inicialmente. A medida que  $N_s$  aumenta; el proceso de generación de partículas, de procesar sus pesos y compararlas con las cotas aumentan y por lo tanto el tiempo de ejecución.

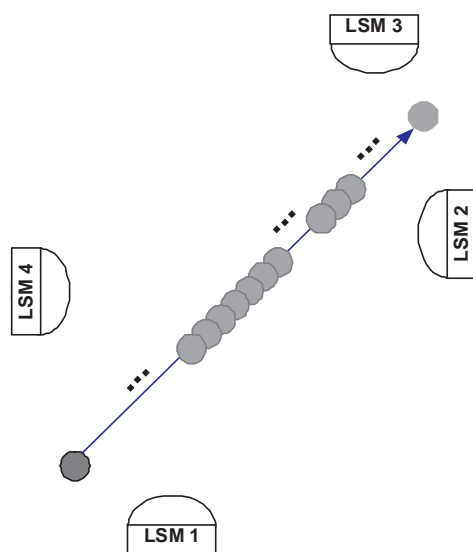


**Figura 4.31:** Tiempo entre corridas del algoritmo para 5 iteraciones por estimación.

### Seguimiento.

Se denomina seguimiento a la localización en el tiempo de un objeto en movimiento; básicamente el procedimiento es el mismo, solo que cada estimación (las cotas en rango y ángulo) es actualizada con el modelo de desplazamiento (dinámico) del objeto antes de ser utilizada en el siguiente paso. Se considera para la aplicación un objeto que se mueve a  $0,01 \text{ m/seg.}$  en ambas coordenadas X e Y, es decir se mueve en diagonal respecto de los ejes.

Para observar cualitativamente el desempeño del BPF para el caso del seguimiento de una fuente, se realizó la siguiente experiencia. Se tomaron observaciones provenientes de un sensor láser y el objeto a seguir se armó en base a un cono cubierto con cinta reflectiva montado sobre un robot Pioneer [Pioneer] que seguía una trayectoria recta a velocidad constante. Con un único sensor laser ubicado en distintos puntos, se simularon cada una de las posiciones de los nodos. Para cada posición se hizo recorrer al robot la misma trayectoria y se sincronizaron las medidas al reloj del Laser. Las observaciones en cada sensor se tomaron cada un segundo, sin embargo no en todos los instantes se obtuvieron rebotes provenientes del objeto.

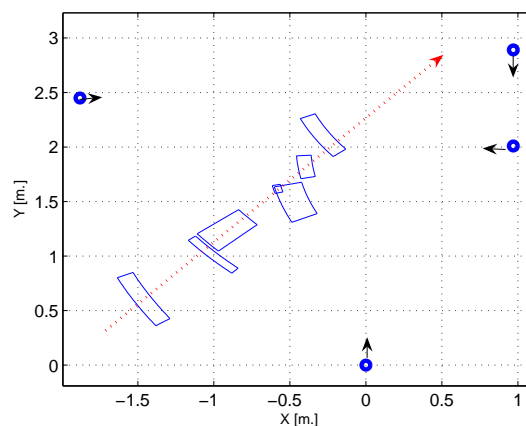


**Figura 4.32:** Gráfica cualitativa de la trayectoria del objeto a localizar y sensores en el ensayo de localización de un objeto en movimiento.

Nodo	X (m.)	Y (m.)
1	0	0
2	0.97	2.01
3	0.97	2.89
4	-1.83	2.45

**Cuadro 4.6:** Ubicación de los Sensores respecto del Nodo 1

En la Fig. 4.32 se muestra un esquema con la ubicación de los nodos láser y la trayectoria del objeto (círculo) en forma cualitativa; en la Tabla 4.6 se detallan las distancias tomando el *Nodo 1* como el origen de coordenadas. En la Fig. 4.33 se muestra el resultado del proceso de estimación. Los círculos azules representan la posición del laser y la flecha la orientación del mismo. La flecha discontinua roja indica la trayectoria promedio seguida por el robot (recuérdese que para obtener la medición de cada sensor, el robot volvía a recorrer la recta y no se puede asegurar que siempre fuera la misma). Las áreas encerradas con líneas continuas azules representan el resultado de la estimación luego de que los cuatro nodos hayan procesado sus medidas, en instantes de tiempo consecutivos. A pesar de las imperfecciones de la experiencia, se puede advertir que el resultado de la aplicación del algoritmo es satisfactorio. Las áreas, luego de procesar la información de todos los nodos, encierran siempre a la posición real del objeto en ese instante de tiempo.



**Figura 4.33:** Ubicación de los objetos a localizar y sensores en el ensayo con objeto en movimiento.



#### 4.2.4. Implementación sobre Nodos de una Red de Sensores

A continuación se presentan los resultados y análisis de la implementación total del BPF sobre el microcontrolador en los nodos de una red de sensores concreta. En las secciones anteriores se presentó un análisis del funcionamiento del BPF donde las observaciones eran procesadas en un nodo de una red de sensores y los resultados enviados a una computadora para la fusión. Aquí se verá que, aún con las limitaciones que introduce la arquitectura de los nodos, es posible realizar la fusión de datos y llegar a un estimado procesando los datos íntegramente sobre la red.

##### **Algoritmo del Filtro de Partículas Acotado en el nodo.**

A continuación se presentará la implementación del Filtro de Partículas Acotado sobre un nodo sensor donde se tendrán en cuenta las limitaciones de procesamiento, almacenamiento y comunicación de datos que el nodo posee. El proceso es el siguiente, inicialmente  $N_s$  partículas son distribuidas sobre la totalidad del espacio de estados de interés. Se adopta un esquema de fusión nodo a nodo, en oposición a uno centralizado como se hizo en secciones anteriores. En dicho esquema un sensor actúa como líder comenzando las estimaciones y seleccionando el siguiente nodo sin ninguna estrategia específica. Esto quedará fuera del alcance de esta tesis habiendo algunos autores que exploran estrategias tales como seleccionar el siguiente nodo en base a cuan informativa puede ser su medida relacionada con la estimación propia [Kaplan (2006)]. Un esquema de la red propuesta se muestra en la Fig. 4.34, donde  $R$  es la incertidumbre del sensor;  $CotaP$  el umbral de peso (ver (4.7)), para la comparación y selección de las partículas;  $Xs_{previa}$  es la posición del último sensor y  $Xs$  es la posición del sensor actual. La única información a comunicar es la posición del sensor y los límites extraídos en la estimación. Cuando llega la información de las cotas desde el sensor anterior, el sensor actual (o también en el caso particular del primer sensor, con una configuración inicial de los estados) distribuye las partículas en el área acotada y realiza una medición y actualización de los pesos. Como resultado se acota el área que contiene las partículas mas pesadas y retenidas en el proceso de comparación con el umbral. De esta forma,

el espacio de estados es reducido al espacio ocupado por las partículas elegidas.

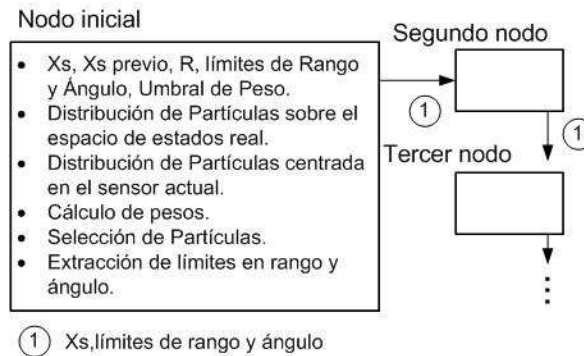


Figura 4.34: Procedimiento del BPF dentro del nodo.

### Funcionamiento del Algoritmo en el Nodo.

Para iniciar la estimación, la información inicial llega desde un nodo inicial o previo y consiste en la posición del nodo y las cotas de ángulo y rango (solo 6 parámetros de doble precisión a ser transmitidos en un único paquete de RF). Una vez que la información llega, el nodo comienza su propio procedimiento de estimación y luego lo comunica al nodo siguiente. Una computadora conectada a uno de los nodos recibe los datos del resultado de la estimación para almacenarlos y graficar los estimados y las partículas, sin intervenir en el procesamiento. El procedimiento de estimación en detalle comienza con una lectura del sensor (observación, medición); las partículas son generadas dentro de las cotas recibidas en coordenadas polares y transformadas a cartesianas para poder trasladarlas relativas al nuevo sensor (utilizando la ubicación del sensor anterior); y vueltas a transformar a coordenadas polares. Cuando una observación es efectuada, se calcula la estimación de la observación sobre cada partícula; y se actualiza su peso. Se compara el ángulo y rango de las partículas cuyos pesos exceden  $CotaP$  con las cotas de rango y ángulo iniciales o previas para actualizarlas de ser necesario.

Dadas las capacidades limitadas de los nodos de la red es necesario implementar un generador de números aleatorios para obtener localmente las partículas iniciales de cada iteración. Se implementó un generador de números aleatorios llamado  $Ran$  [ $Ran$ ] y se modificó en dos formas para obtener distribuciones totalmente uniformes. Primero

se lee el contador del reloj del nodo y se rota para obtener un número que no sea monótonamente creciente, dicho número se suma a la semilla. En segundo lugar, cada número generado se utiliza como semilla para la próxima generación.

Para obtener una partícula se genera un número aleatorio y se opera para obtener un número entre las cotas de rango. Luego para dicho rango se genera un nuevo número aleatorio que es operado para que caiga dentro de las cotas de ángulo. Una vez generada la partícula dentro de las cotas comunicadas por el nodo anterior se traslada en referencia al nodo actual.

En la Fig. 4.35 se muestra un diagrama de la implementación del algoritmo sobre el nodo; donde los bloques grises son los procedimientos que no son propios del BPF; como la interacción con otros nodos y la medición del sensor. En este ensayo no se conectaron sensores a los nodos, sino que las medidas se simularon fuera de línea. El objeto de esto, es permitir analizar el desempeño del filtro mas allá de los modelos del sensor que agregan sus incertezas.

Las tareas propias del algoritmo enunciadas en al Fig. 4.35 son las siguientes:

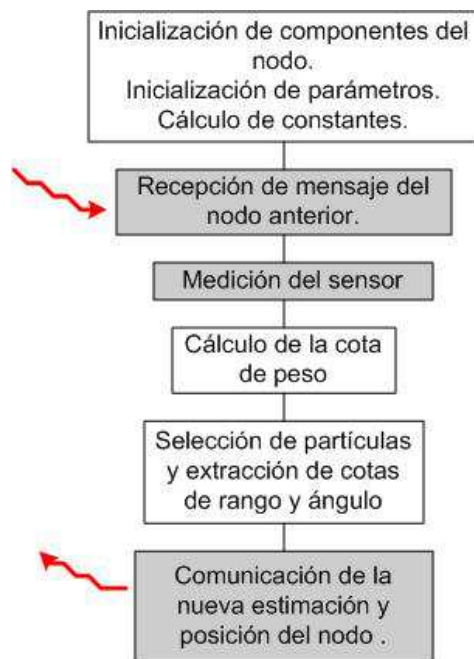
- Inicialización de parámetros: Los parámetros  $A_{min}$  (ángulo mínimo),  $A_{max}$  (ángulo máximo),  $R_{min}$  (rango mínimo) y  $R_{max}$  (rango máximo) representan las cotas en rango y ángulo que determinan la región de mayor probabilidad. Al final de una iteración del algoritmo, definirán la estimación. Su inicialización se hace para asegurar que la primer partícula que sobreviva al proceso de selección con  $CotaP$ , los modifique. Por ejemplo, si el alcance en rango de un sensor son 10 metros,  $R_{min}$  será 20 metros y  $R_{max}$  0 metros. De esta forma, se asegura que la región acotada esté vacía y se vaya formando con el proceso de supervivencia. Si la primer partícula que sobrevive tiene un rango de 5 metros, tanto  $R_{min}$  como  $R_{max}$  será de 5 metros. Si una segunda partícula sobreviviente tiene 2 metros,  $R_{max}$  no se modificará pero si  $R_{min}$  a 2 metros. De igual forma se actualizan las cotas de ángulo y este proceso sigue hasta el final de la iteración.
- Medición del sensor: En el caso analizado en esta sección se simula la medida

de un sensor de rango. Fuera de línea, partir de la posición del sensor se genera una observación de rango a la que se modela como gaussiana [The MathWorks (1994-2010)]. Se calcula el valor del rango (correspondiente a cada sensor) más el ruido; generado con (4.14) donde  $r$  es el rango,  $v$  el ruido del sensor (en este caso 0.05 m.) y  $rand(n, m)$  es una función que devuelve una matriz de  $n \times m$  muestras con distribución Gaussiana de media cero y varianza 1.

$$z = r + v.rand(1000, 1) \quad (4.14)$$

Se calculan 1000 valores con media el rango y varianza conocida (5 cm.) y en el nodo se elige entre estas muestras en forma aleatoria una que representará la lectura del sensor.

- Cálculo de la Cota de Peso ( $CotaP$ ): Se genera un número inicial de partículas,  $M$ , se calcula su peso y se selecciona el peso máximo.  $CotaP$  resultará entonces un porcentaje de este peso máximo y las partículas generadas son descartadas sin utilizarse en el proceso de estimación.
- Selección de partículas y extracción de cotas de ángulo y rango: En esta etapa del algoritmo, cuando una partícula cuyo peso  $w_i \geq CotaP$ , se incrementa un contador de partículas  $k$  en uno y se actualizan los parámetros  $A_{min}$ ,  $A_{max}$ ,  $R_{min}$ ,  $R_{max}$  con el ángulo y rango de la nueva partícula. Si  $w_i < CotaP$  se descarta la partícula. Este proceso se repite con todas las partículas ( $N_s$ ) hasta que el mínimo número de partículas,  $N_{min}$ , se cumple ( $k \geq N_{min}$ ). Esto se hace, comparando el rango y el ángulo de la partícula con los parámetros. Si la partícula es mayor a los máximos, se modifican estos de acuerdo su valor. Si es menor, se modifican lo mínimos.



**Figura 4.35:** Comportamiento del nodo.

En la Fig. 4.36 se especifica mas detalladamente el procedimiento realizado dentro de cada nodo. En primer lugar se generan  $M$  partículas, dentro de las cotas recibidas y debidamente trasladadas, de las que se extrae el peso máximo y se calcula  $CotaP$ . Se define el número de partículas mínimo que se desea seleccionar,  $N_{min}$  y se inicializa un contador de partículas seleccionadas  $k = 0$ . También se inicializan los parámetros de las cotas de la región mas probable de manera que la primer partícula generada las modifique (como se explicó anteriormente). Comienza la etapa de estimación, se generan  $N_s$  partículas dentro de las cotas recibidas, se calcula el peso y se compara con  $CotaP$ . Las partículas cuyo  $w_i$  supera  $CotaP$  son utilizadas para actualizar las cotas de la región mas probable e incrementan  $k$  en uno, el resto son descartadas sin producir cambios en  $k$  o en los parámetros de las cotas. Si al terminar la selección  $k$  es menor que el número mínimo requerido de partículas seleccionadas, se reduce  $CotaP$ , se inicializan los parámetros y se comienza nuevamente la selección siendo que las partículas anteriores solo modificaban las cotas pero no eran almacenadas en memoria. Una vez que  $k$  alcanza el valor  $N_{min}$  se termina el procedimiento y se comunican las cotas de la región mas probable al nodo siguiente o bien a la computadora.

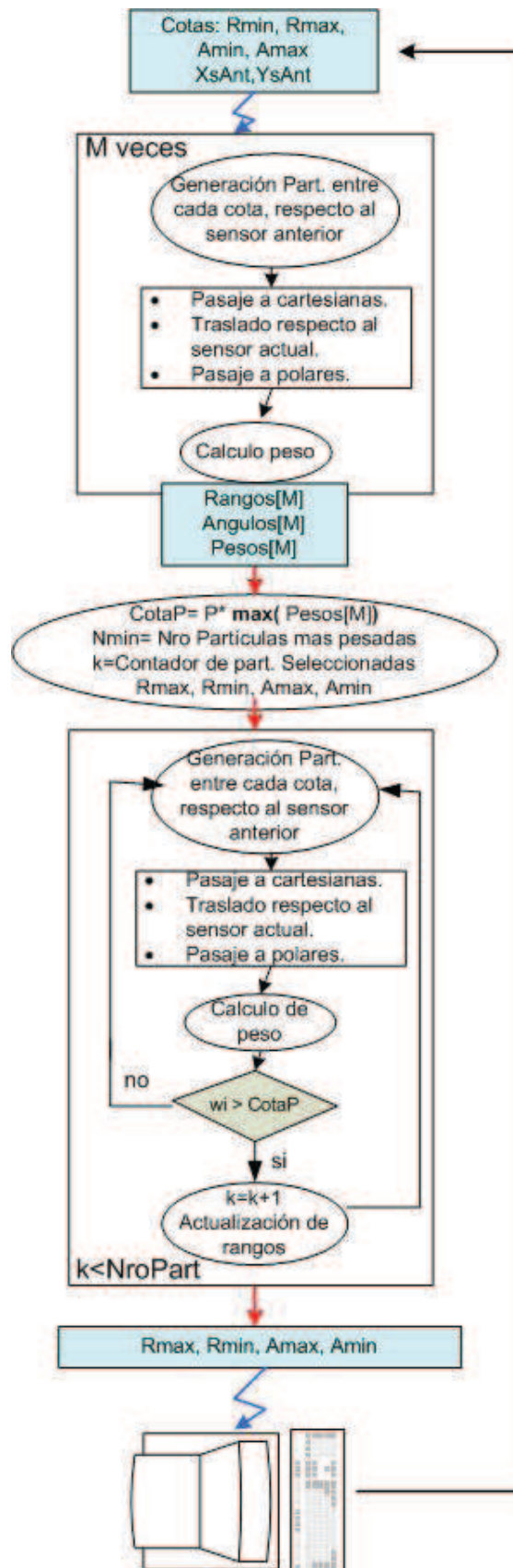


Figura 4.36: Detalle del algoritmo dentro de los nodos.

Nodo	X (m.)	Y (m.)	z (m.)
1	0	0	-
2	1	1	1.998
3	4	-1	4.121
4	-1	3	3.1699

**Cuadro 4.7:** Ubicación y orden de estimación de los sensores con el *Nodo1* como origen de coordenadas,  $z$  corresponde a la observación de rango de cada nodo.

### Ensayo del algoritmo en el nodo.

Se ensaya una red de sensores programando el algoritmo presentado en la sección anterior sobre nodos MicaUNS y aplicando un esquema de fusión nodo a nodo. Los nodos sensores realizan el procesamiento en el orden y la ubicación especificada en la Tabla 4.7. Se realizan los sucesivos ensayos manteniendo la configuración de la red, el orden de procesamiento y el valor medido.

El proceso para encontrar un valor local (en cada nodo) de  $CotaP$  que sea función de las cotas recibidas y la observación hecha por el nodo se realiza en forma dinámica, dado que la capacidad de almacenamiento de datos es limitada. Es decir, se genera una partícula, se extraen las propiedades de interés (en este caso el peso) y se la destruye. Se genera una nueva partícula y se repite el proceso actualizando el valor del peso máximo.  $M$  es el número de partículas generadas con el fin de encontrar el peso máximo.

Para cada ensayo se realizaron 20 corridas del algoritmo BPF sobre las cuales se evaluó, sobre las partículas de la estimación resultante de cada corrida, el promedio de la media y la varianza de ellas. Para analizar comparativamente el número total de partículas generadas en cada ensayo, se mantuvo  $N_{min}$  constante en 15 partículas y el  $N_s$  de cada ensayo se calcula como la media de estos valores. Además se calculó el área resultante con (4.13).

La intención es variar  $M$  con el fin de encontrar el peso máximo; de ahí elegir  $CotaP$  como un porcentaje del máximo peso y poder ver los efectos de dicha variación en el estimado resultante.

Se realizaron 18 ensayos combinando la variación de los siguientes parámetros:

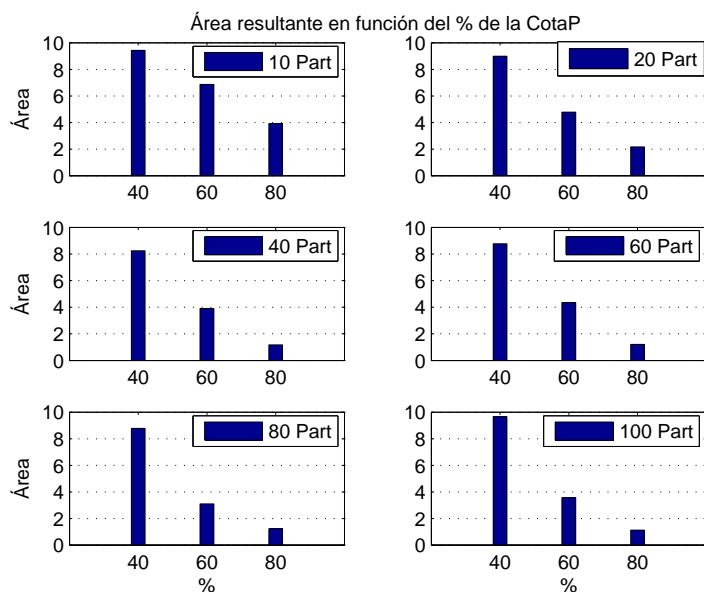
- El número de partículas  $M$ : 10, 20, 40, 60, 80 y 100 sobre las cuales elijo la de peso máximo para luego calcular  $CotaP$ .
  
- El porcentaje del peso máximo,  $P$ , que se adopta por  $CotaP$ : 40, 60 y 80%. De esta forma se varía la exigencia en la selección de partículas. En el caso de estos ensayos el sensor es muy preciso y permite utilizar una cota del 80% del peso máximo; de todas maneras esto permite observar qué pasaría si los sensores no fueran tan precisos.

Una corrida consta de tres estimaciones consecutivas, en el orden (2-3-4) de los nodos. La tercer estimación consecutiva es denominada *estimación resultante*.

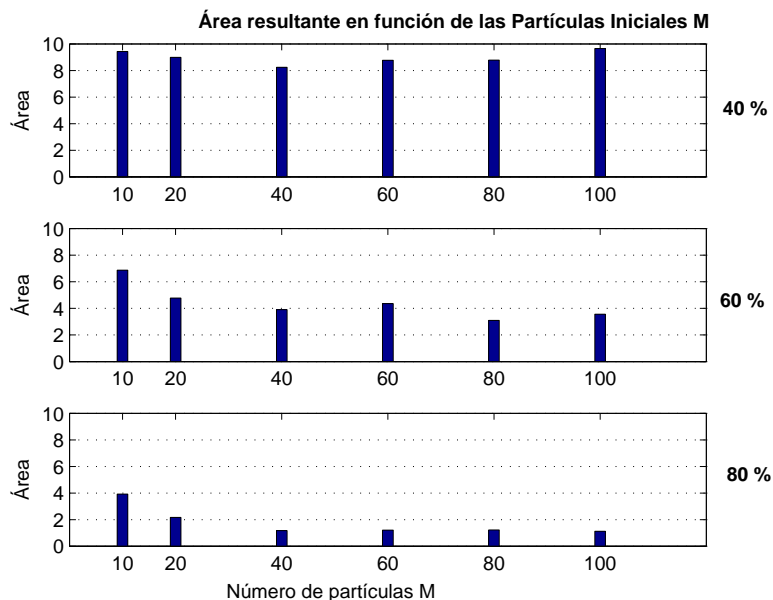
En la Fig. 4.37 se ve el *área* de la estimación resultante como función del porcentaje  $P$  del peso máximo que se toma como cota de peso. Arriba sobre la derecha se indica el número de partículas iniciales,  $M$ , generadas para elegir el peso de la mas pesada. En la Fig. 4.38 se muestran en forma diferente los mismos resultados que en la Fig. 4.37. Allí se ve el *área* de la estimación resultante como función de  $M$ . Se ve que a medida que el porcentaje ( $P$ ) del peso máximo aumenta, disminuye el *área*; esto ocurre porque se está siendo exigente en la selección de las partículas suponiendo que la precisión del sensor lo permite.

En la Fig. 4.38 se aprecia que para un determinado  $P$ , aun variando  $M$  sustancialmente para elegir el peso máximo, los resultados del algoritmo no son afectados en forma importante. Esto implica que generándose las partículas en forma aleatoria, la más pesada se genera entre las partículas iniciales.





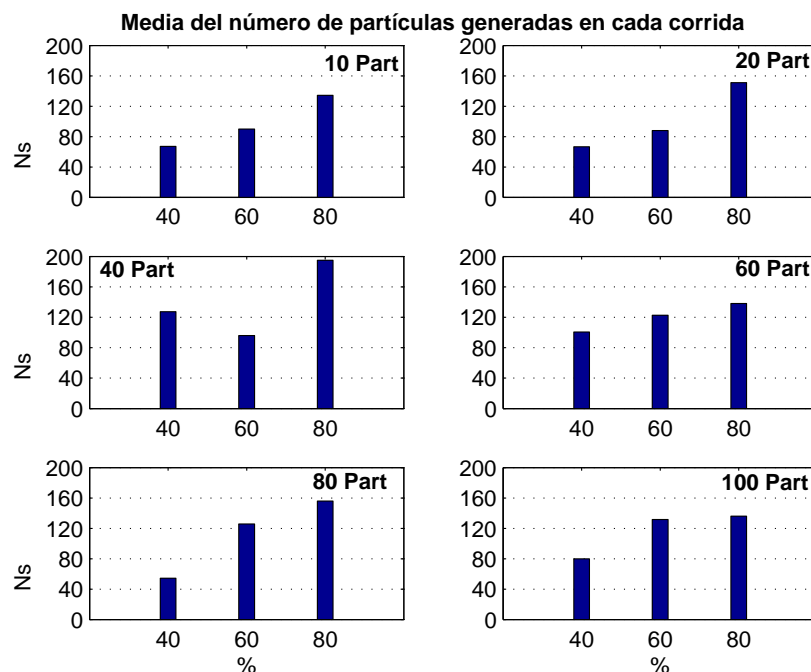
**Figura 4.37:** Área de la estimación resultante para 40, 60 y 80% del peso máximo para la selección de las partículas. Arriba sobre la derecha se indica el número de partículas iniciales  $M$ , generadas para tomar el peso máximo utilizado para el cálculo de  $CotaP$ .



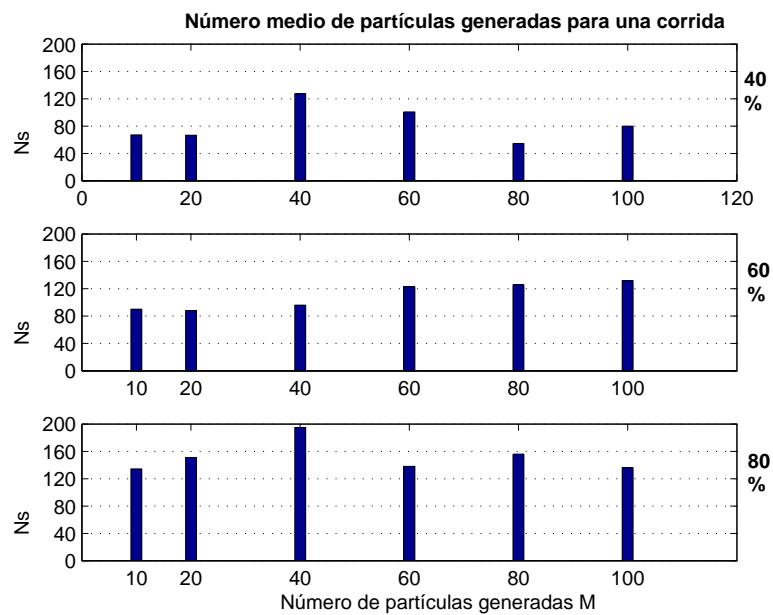
**Figura 4.38:** Área de la estimación resultante para los diferentes números de partículas iniciales  $M$ , para extraer el peso máximo. A la derecha de cada gráfico se indica el porcentaje del peso máximo con que se calcula  $CotaP$ .

En las Fig. 4.39 y Fig. 4.40 se muestra el número medio de partículas generadas por corrida en función de los porcentajes  $P$  y del número de partículas iniciales  $M$ . Se recuerda que el BPF implementado en el nodo no tiene un  $N_s$  definido, sino que se generan las partículas hasta lograr cumplir con  $N_{min}$  de ellas seleccionadas. A medida que aumenta el porcentaje, aumenta  $CotaP$  haciendo mas exigente la selección; esto hace que se reduzca el área más probable, haciendo necesaria la generación de un mayor número de partículas para lograr cumplir con las 15 mínimas requeridas para terminar la estimación, como puede apreciarse en las figuras.

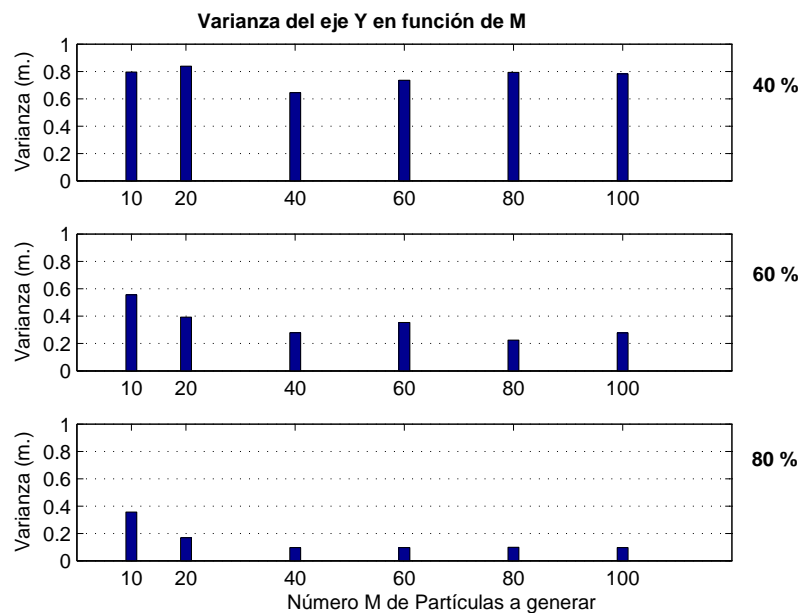
En la Fig. 4.41 se muestra la varianza de las partículas seleccionadas en la estimación resultante sobre el eje Y para diferentes porcentajes del peso máximo en el cálculo de  $CotaP$ . Se puede observar que la varianza se reduce notablemente al aumentar el porcentaje en el cálculo de  $CotaP$  lo que se condice con lo visto en las figuras anteriores (Fig. 4.39 y Fig. 4.40), donde a medida que aumenta el porcentaje, aumenta  $CotaP$  haciendo mas exigente la selección y reduciendo el área más probable.



**Figura 4.39:** Número medio de partículas generadas  $N_s$  en cada corrida para 40, 60 y 80% del peso máximo como  $CotaP$ . Arriba sobre la derecha se indica el número de partículas iniciales  $M$ , para la extracción del peso máximo.

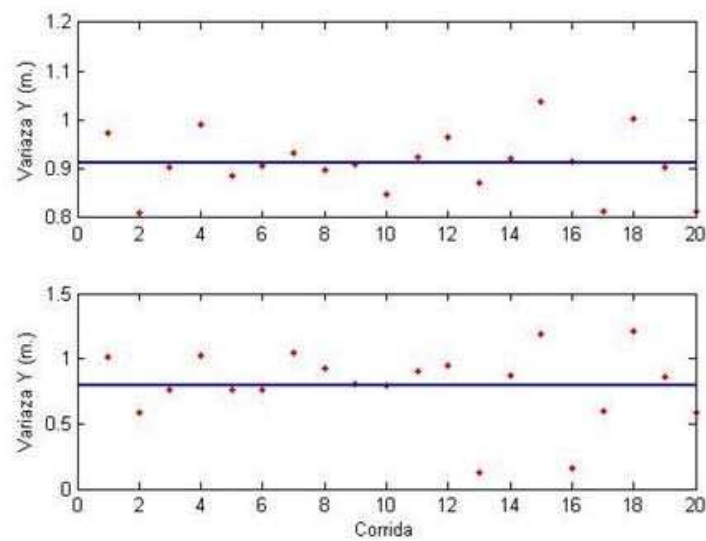


**Figura 4.40:** Media del número de partículas generadas  $N_s$ , en una corrida en función de las partículas iniciales  $M$ , generadas para extraer el peso máximo. A la derecha de cada gráfico se indica el porcentaje del peso máximo con que se calcula  $CotaP$ .

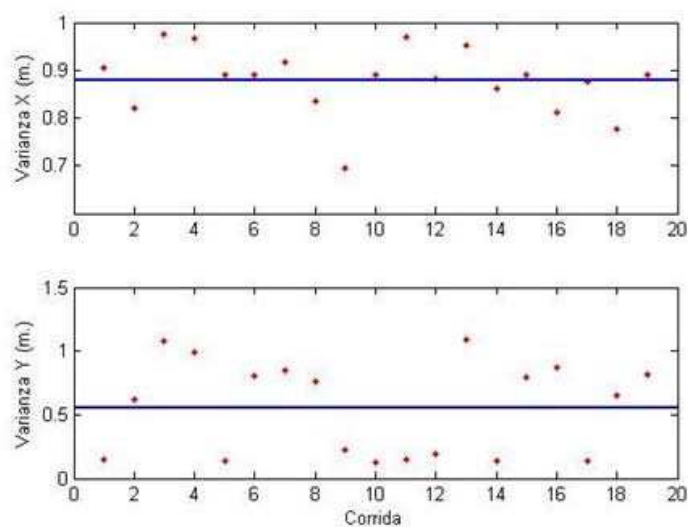


**Figura 4.41:** Varianza de las partículas seleccionadas en la estimación resultante sobre el eje Y respecto a  $M$ , para diferentes porcentajes del peso máximo para el cálculo de  $CotaP$ .

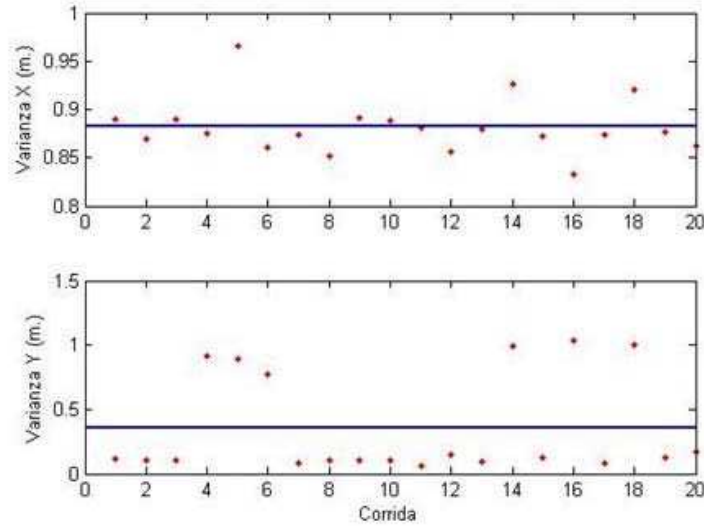
En las Fig. 4.42, Fig. 4.43 y Fig. 4.44 se muestra la varianza de las partículas seleccionadas en la estimación resultante en cada corrida para  $M = 10$  partículas y  $P$  de 40 %, 60 % y 80 % respectivamente. Se puede apreciar que las varianzas se encuentran cerca de la varianza media de las 20 corridas. Además la dispersión respecto de la media es mayor cuando menor es  $P$ , lo que coincide con lo observado en la Fig. 4.41.



**Figura 4.42:** Varianza de las partículas seleccionadas en la estimación resultante sobre el eje X e Y del espacio de estados para  $M = 10$  y  $P = 40\%$ .

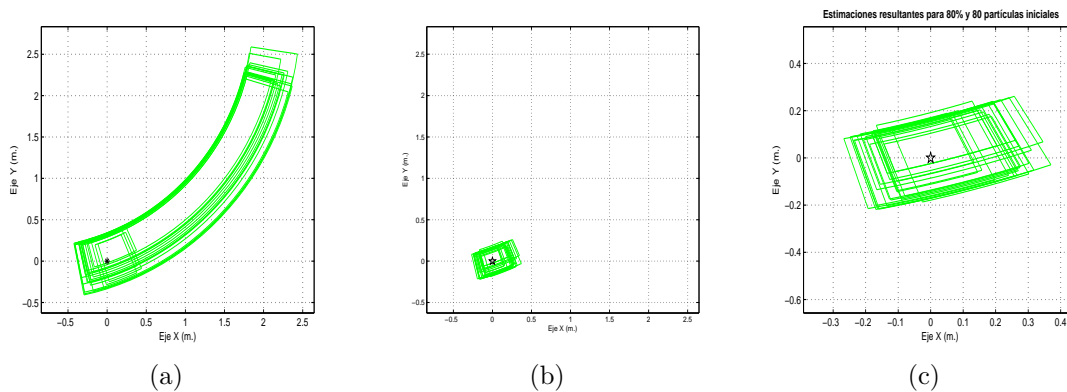


**Figura 4.43:** Varianza de las partículas seleccionadas en la estimación resultante sobre el eje X e Y del espacio de estados para  $M = 10$  y  $P = 60\%$ .



**Figura 4.44:** Varianza de las partículas seleccionadas en la estimación resultante sobre el eje X e Y del espacio de estados para  $M = 10$  y  $P = 80\%$ .

En la Fig. 4.45 se pueden ver las estimaciones resultantes de las 20 corridas del algoritmo para el caso particular de  $M = 80$ . En la Fig. 4.45(a) se utiliza una cota de peso del 40% del peso máximo, mientras que en la Fig. 4.45(b) se utiliza una cota del 80% del peso máximo. Se ve claramente la forma en que se reduce el área más probable. La Fig. 4.45(c) es una ampliación de la Fig. 4.45(b).



**Figura 4.45:** Estimaciones resultantes de las 20 corridas del algoritmo para 80 partículas iniciales de donde se selecciona el mayor peso. En la Fig. 4.45(a) se utiliza una cota de peso del 40% del peso máximo, mientras que en la Fig. 4.45(b) se utiliza una cota del 80% del peso máximo como  $CotaP$ . La Fig. 4.45(c) es una ampliación de la Fig. 4.45(b)

Lo que se pudo apreciar con estos ensayos es que a medida que el porcentaje ( $P$ ) del peso máximo aumenta, disminuye el *área* y la varianza notablemente, y que el hecho de elegir el peso máximo entre más o menos partículas ( $M$ ) no genera un comportamiento destacable del algoritmo. En base a lo visto se pueden elegir los parámetros óptimos para la aplicación en estudio, este tipo de ensayos debería realizarse antes de dejar una red de sensores en funcionamiento con el BPF.

Algo importante que se podría agregar al algoritmo con mínimo procesamiento es el cálculo del área transmitida en la estimación anterior y el cálculo del área de la estimación actual. De esta manera se podría realizar la comparación de ellas y, en caso de no encontrar gran diferencia, se podría decir que la observación del sensor actual no aporta información a la estimación. Esto cumpliría el mismo rol que el número de partículas efectivo en los Filtro de Partículas convencionales [Arulampalam et al. (2002)]. Un parámetro que permite la misma observación en los PF es la variación de  $N_{eff}$ .

### Casos Especiales

La distribución de los nodos de una red de sensores en el espacio, respecto del objeto a localizar o seguir, afecta la calidad de la estimación. En esta sección se analiza el comportamiento del BPF en algunos de estos casos; además se analiza cómo se puede mejorar la estimación con algunas modificaciones simples al filtro. En los casos que se presentan a continuación las redes de sensores están formadas por nodos con medición de rango en los cuales se implementa el BPF, para localización de un objeto.

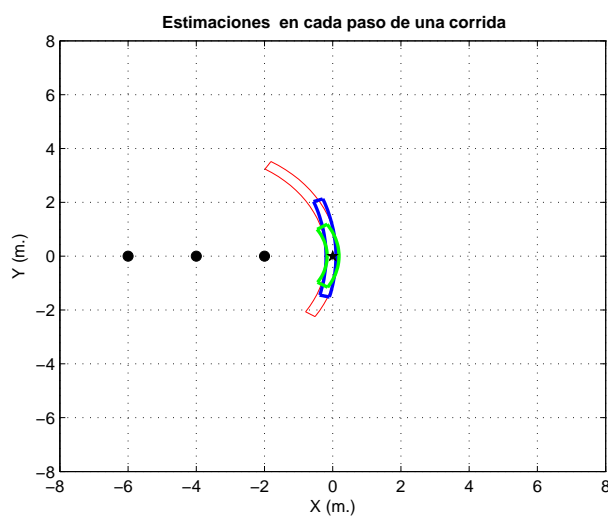
**Caso 1.** Se plantea una configuración de red donde los sensores están alineados. Esto provoca que los anillos de las cotas del estimado no se corten perpendicularmente en ningún momento. Para este caso se seleccionaron las partículas cuya cota supere el 60% del peso máximo entre los pesos de  $M = 80$  partículas inicialmente generadas. Una vez que se encuentran  $N_{min} = 15$  partículas que superan dicha cota, se extraen las cotas de rango y ángulo que las encierran. En la Tabla 4.8 se detalla la ubicación

Nodo	X (m.)	Y (m.)	z (m.)
1	0	0	-
2	-4	0	4.0499
3	-6	0	6.0311
4	-2	0	1.9474

**Cuadro 4.8:** Ubicación y orden de estimación de los sensores con el Nodo 1 como origen de coordenadas.

de los sensores.

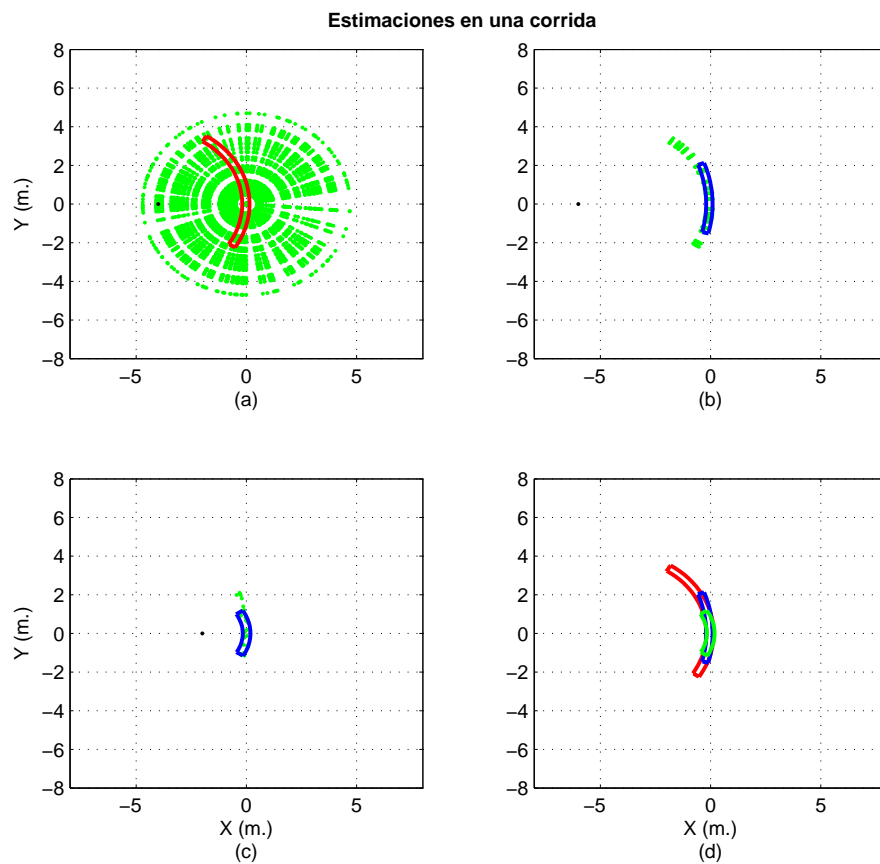
En la Fig. 4.46 se pueden apreciar las sucesivas iteraciones hasta llegar a la estimación resultante. Se simbolizan con círculos los sensores y con una estrella el objeto a localizar, que se encuentra en el origen de coordenadas.



**Figura 4.46:** Estimaciones para un caso especial. Se representan con círculos los sensores y con una estrella el objeto a localizar, que está en el origen. Encerrando al objeto a localizar se encuentran las tres áreas estimadas, en rojo el resultado de la primer estimación, en azul el de la segunda y en verde la estimación resultante.

La Fig. 4.47 muestra en detalle cómo actúa cada nodo. En cada subfigura se muestran las partículas generadas dentro de las cotas iniciales totales (Fig. 4.47 (a)) o dentro de las cotas que recibió del sensor anterior (Fig. 4.47 (b), (c)), y en línea llena las nuevas

cotas que encierran las partículas seleccionadas por tener un peso superior a  $CotaP$  (Fig. 4.47(a), (b), (c)). La Fig. 4.47(d) muestra las áreas mas probables obtenidas en cada estimación de la corrida. Puede deducirse de la figura que sin importar el número de nodos alineados que realicen estimaciones consecutivas, la estimación final no mejorará demasiado.

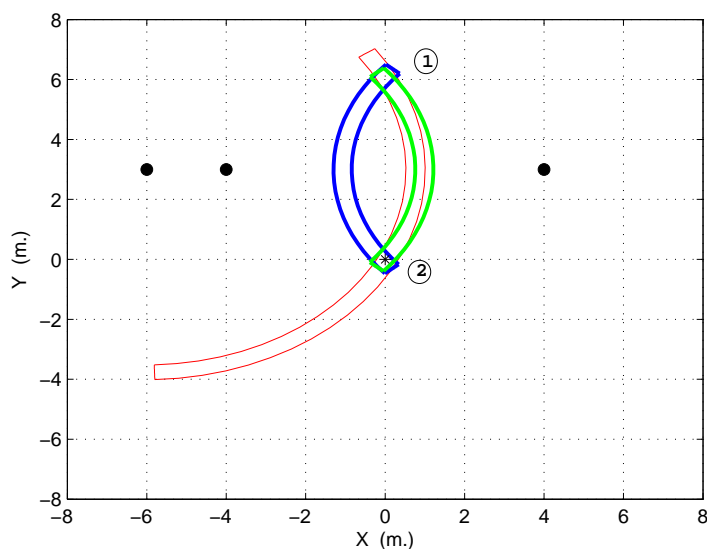


**Figura 4.47:** Pasos en una corrida, en verde se encuentran las partículas generadas inicialmente, sobre las que se realiza la estimación; en negro el nodo que la realiza: (a) Primer estimación sobre el espacio total de estados, resultado en rojo, (b) Segunda estimación, resultado en azul, (c) Estimación resultante, resultado en azul, (d) Las tres áreas estimadas, en rojo el resultado de la primer estimación, en azul el de la segunda y en verde la estimación resultante.

**Caso 2: Múltiples hipótesis.** En este caso se presenta una red de sensores de rango para localización de un objeto en la que los nodos están alineados y el objeto a localizar se encuentra a tres metros de distancia, perpendicular a ellos.

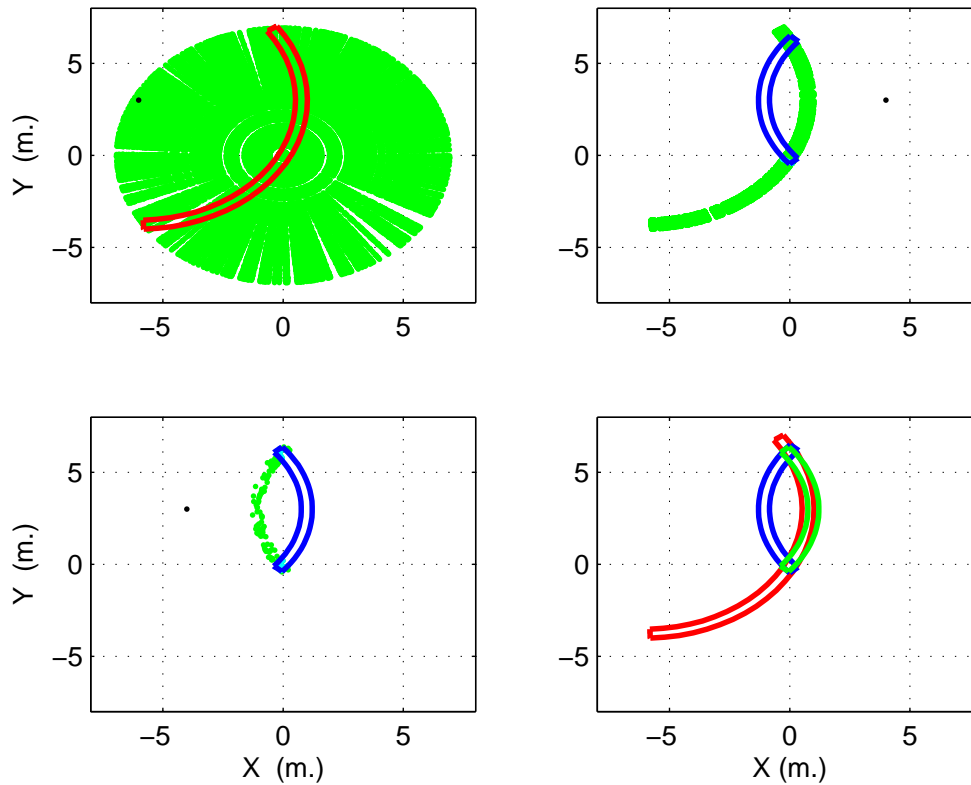


La Fig. 4.48 muestra en detalle las áreas de las estimaciones resultantes de una corrida del algoritmo en la red. Los círculos negros corresponden a los sensores y la estrella negra al objeto a localizar. La primera estimación se representa en rojo, la segunda en azul y la final en verde. Como se puede observar, las áreas estimadas no se reducen y no lo harán, debido a la configuración de la red.



**Figura 4.48:** Áreas estimadas. Los círculos negros corresponden a los sensores y la estrella negra al objeto a localizar. La primer estimación se representa en rojo, la segunda en azul y la final en verde.

La Fig. 4.49 muestra una corrida de estimaciones de la red; en cada caso se muestran las partículas generadas dentro de las cotas iniciales totales (Fig. 4.49 (a)) o dentro de las cotas que pasó el sensor anterior (Fig. 4.49 (b), (c)), y en línea llena el área determinada por la nueva estimación, es decir, las nuevas cotas que encierran las partículas seleccionadas por tener un peso superior a  $CotaP$  (Fig. 4.49 (a), (b), (c)). La Fig. 4.49(d) muestra las áreas mas probables obtenidas en cada estimación de la corrida.



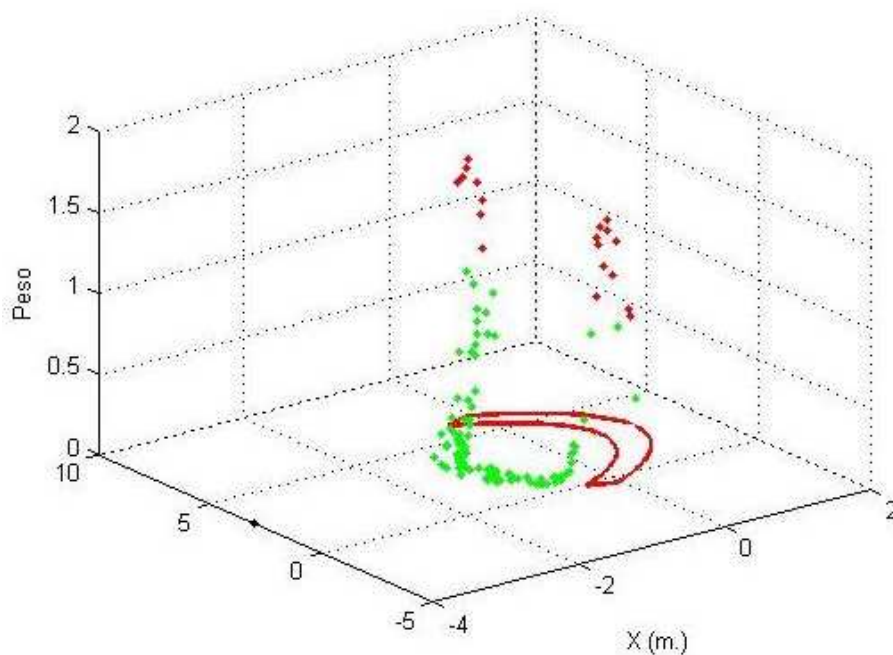
**Figura 4.49:** Corrida de estimaciones. Los gráficos muestran las partículas generadas dentro de las cotas iniciales totales (arriba izquierda) o dentro de las cotas que pasó el sensor anterior (arriba derecha, abajo izquierda), y en línea llena el área determinada por la nueva estimación. El último gráfico (abajo derecha) muestra las áreas mas probables anteriormente obtenidas.

La situación que se plantea aquí es que se transmite un único conjunto de cotas y el resultado de la estimación posee mas de una hipótesis probable, es un caso de múltiples hipótesis. Como se ve en la Fig. 4.48, con la distribución de los sensores no se puede discriminar si el objeto se encuentra en la posición 1 ó 2 que es donde se intersectan las sucesivas estimaciones. Esto se ve mas claro en la Fig. 4.50, donde se grafica en el eje Z el valor del peso. Así, se pone de manifiesto la ubicación de las partículas mas probables. Al encerrar estas en una sola cota se generan siempre zonas como las mostradas en las Fig. 4.48 (b) y (c), donde además de las partículas mas probables, se incluyen regiones de baja probabilidad.

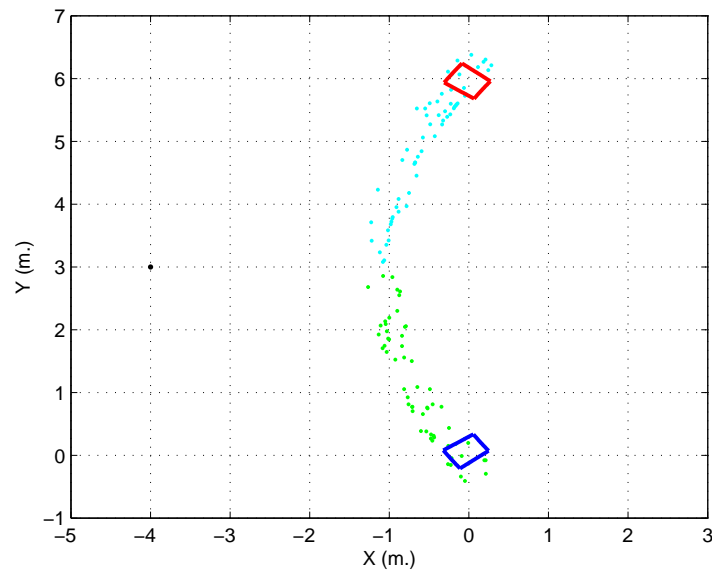
La solución a esto es transmitir mas de un conjunto de cotas. Esto es simple de

administrar e incluir en el BPF. En la Fig. 4.51 se muestra el resultado de la estimación con esta estrategia. Si bien se mantienen las dos hipótesis posibles, la estimación no contiene zonas de baja probabilidad dentro del área resultante.

El algoritmo general plantea la posibilidad de transmitir todos los elementos que describen la distribución. Aunque todo el análisis previo se hizo a partir de la generación de la mínima información a transmitir, este ejemplo demuestra la adaptabilidad del BPF a las condiciones de la aplicación. Sin cambiar el esquema general es posible solucionar en forma eficiente cualquier problema de localización y/o seguimiento.



**Figura 4.50:** Partículas con sus pesos respectivos, las partículas verdes son las de peso inferior a  $CotaP$ , las rojas las seleccionadas. En línea llena se muestra el área resultante utilizando una única cota. Se puede apreciar que existen dos áreas con alta probabilidad, aunque una sola es la que contiene el objeto (múltiples hipótesis) se ve claramente que hay dos zonas más probables.



**Figura 4.51:** Resultado de analizar la estimación con dos conjuntos de cotas, se divide el area previa resultante en 2 y se generan las partículas dentro de cada área (partículas verdes y azules). Luego se realiza la selección y se acotan las partículas seleccionadas de cada mitad, resultando en las áreas en línea llena azul y roja.

### 4.3. Contribuciones

En este capítulo se presentaron las principales contribuciones de esta tesis. En resumen se resalta:

- Se probó que es posible la implementación de un algoritmo con mínimos requerimientos de almacenamiento y/o transmisión de datos entre iteraciones [Sañudo et al. (2007)]. Lo único que se requiere transmitir, en su versión mas simple, son las cotas dentro de las cuales se hallan las partículas con mayor probabilidad de contener el estado de interés; y de ser necesario propiedades estadísticas de ellas.
- Se probó que es posible lograr una convergencia suficientemente rápida para las aplicaciones presentadas, aún cuando se parte en cada iteración de una distribución uniforme dentro del área estimada en el paso anterior. Además, el hecho de que el espacio de estados se va reduciendo, hace que se pueda ir reduciendo el número de partículas generadas al comienzo de las iteraciones.

- Se probó que es posible la representación de densidades de probabilidad arbitrarias; ya que, como se pudo ver en las aplicaciones, a medida que se va modificando el espacio de estados y/ó el sensor (en posición y/ó medida y/ó calidad), la pdf dentro de la misma no posee características Gaussianas.
- Se probó que es posible evitar el empobrecimiento de muestras, ya que en cada selección de partículas se adapta la cota de peso de forma de conservar un número mínimo de ellas; un número que puede variar en cada nodo. La cota es adaptada de modo que su valor permita conservar un número mínimo de partículas, pero sin ser demasiado conservativo para no perder información. Aunque no hay una forma cerrada para estos parámetros es posible adaptar el proceso de estimación experimentalmente.
- Se logró combinar la ventaja de poder representar pdf arbitrarias del Filtro de Partículas con las limitaciones de una red de sensores [Sañudo et al. (2011)]. Para esto se realiza una criteriosa selección de partículas y se acota el espacio de aquellas seleccionadas; de modo que toda la información esta contenida en las cotas; siendo este el único dato a transmitir para una nueva estimación (en el caso de mínimo requerimiento de comunicación).
- Se puede analizar el aporte de cada observación analizando, con mínimo procesamiento, la diferencia entre las áreas de la región estimada anterior y actual.
- El número de partículas iniciales  $N_s$  se puede adaptar a medida que el área d3e la región estimada se reduce.

#### 4.4. Conclusiones

En este capítulo se presenta el algoritmo BPF partiendo del concepto general hasta el detalle de su implementación en una red de sensores real. Luego se detallan los sensores utilizados en las aplicaciones prácticas del BPF y desarrollan aplicaciones de

localización de fuentes u objetos y el seguimiento de un objeto. Se realiza un estudio estadístico de la influencia de la variación de diferentes parámetros de ajuste del BPF para una configuración de red y aplicación específicas. Se presenta luego una aplicación de localización de un objeto con el BPF implementado íntegramente sobre una plataforma de nodo comercial y nuevamente se evalúa el desempeño del mismo ante la variación de los parámetros de ajuste. Se extraen observaciones, elementos importantes y conclusiones que ayudan en la elección óptima de estos parámetros y en la correcta implementación del BPF para cada caso específico. Finalmente se listan las contribuciones que se logran con el filtro propuesto.

# Capítulo 5

## Conclusiones

### 5.1. Resumen del trabajo

Las Redes de Sensores son el paradigma de medida distribuida, en lugar de utilizar sensores locales precisos, se utiliza un conjunto de dispositivos denominados nodos a los que se conectan sensores menos precisos, pero que permiten obtener mas información del lugar a sensar debido a su distribución espacial. Una de las aplicaciones mas comunes de las redes es la localización y seguimiento de fuentes u eventos; donde las estimaciones de los estados se basan en medidas ruidosas aportadas por los sensores y en la información obtenida a priori. Dichas aplicaciones involucran generalmente modelos no lineales y estimaciones resultantes no Gaussianas. En la presente tesis se presenta un Algoritmo de Filtro de Partículas Acotado especialmente pensado para ser utilizado en Redes de Sensores e implementado en los nodos. Se decidió trabajar sobre el Filtro de Partículas ya que es un filtro no lineal que permite trabajar con funciones de probabilidad no Gaussianas, modelos de observación y proceso no lineales y permite múltiples hipótesis. Hay determinadas cosas que se tuvieron en cuenta a la hora de desarrollar la solución propuesta; primero el hecho de que para ser implementado se debe cumplir con todas las restricciones de procesamiento, almacenamiento y comunicación que presentan los nodos; segundo el problema de empobrecimiento de muestras y de convergencia que presenta el Filtro de Partículas original; tercero el procesamiento que implica el remuestreo y la forma de resumir la información de la estimación de modo de transmitir la menor cantidad de datos posible. Se probó experimentalmente que el

filtro converge a un estimado representativo del estado con muy poco procesamiento y sin necesidad de almacenar ni transmitir partículas.

## 5.2. Contribuciones

En la presente tesis se probó que es posible la implementación de un algoritmo con mínimos requerimientos de almacenamiento y/o transmisión de datos entre iteraciones ([Sañudo et al. (2007)], [Sañudo et al. (2011)]), ya que lo único que se requiere son las cotas dentro de las cuales se hallan las partículas con mayor probabilidad de ser el estado de interés, y de ser necesario se pueden comunicar otros datos de interés de dichas partículas. Se probó que se logra una convergencia relativamente rápida a un estimado, aún cuando se parte en cada iteración de la generación de partículas dentro del área estimada en el paso anterior. Además, el hecho de que el espacio de estados se va reduciendo, hace que pueda ir reduciendo el número de partículas generadas al comienzo de las iteraciones.

También se probó que es posible representar funciones de probabilidad totalmente arbitrarias, lo que es necesario ya que a medida que se va modificando el espacio de estados y/ó el sensor (en posición y/ó medida y/ó calidad), la pdf dentro del mismo adquiere características que no son posibles de representar como Gaussianas. Además se probó que es posible evitar el empobrecimiento de muestras si en cada selección de partículas se adapta la cota de peso de forma de conservar un número mínimo de estas, teniendo en cuenta que en cada iteración el número de partículas generadas inicialmente puede variarse. La cota es adaptada en base a un compromiso entre conservar un número mínimo de partículas y ser demasiado conservativo para no perder información.

Se logra combinar la ventaja del Filtro de Partículas para representar pdf arbitrarias con la selección criteriosa de partículas (que evita el empobrecimiento de muestras) a partir de cotas que contengan la información para la siguiente iteración, siendo este el único dato a transmitir entre nodos en una implementación mínima del BPF.

Con mínimo procesamiento se puede analizar el aporte de cada observación analizando la diferencia entre las áreas de la región estimada anterior y actual. De esta



forma es posible calificar si un sensor aporta o no información nueva a la estimación. Además puede adaptarse  $N_s$  a medida que el área de la región estimada se reduce. Si es la primer iteración, partiendo del espacio total de estados, el número inicial de partículas generadas  $N_s$  será mayor que en el caso que ya se hayan realizado varias iteraciones y se realice la estimación sobre un subespacio reducido.

Finalmente, y también con mínimo procesamiento agregado, se puede extender la representación de la estimación a varias áreas resultantes, para abordar casos de multi-hipótesis.

### 5.3. Líneas futuras

En la tesis se mostraron resultados de la implementación del filtro sobre nodos comerciales, por software, pero debido a la necesidad de reducir la energía al mínimo, sería muy interesante la implementación del algoritmo en un circuito integrado. Se demostró que las funciones empleadas son básicas y el almacenamiento necesario mínimo, de modo que no se está muy lejos de la posible implementación.

Hasta ahora, algunas arquitecturas han sido propuestas en la literatura ([Hong et al. (2004)], [Hong et al. (2005)]), con implementaciones en FPGA de hasta 10,000 partículas. Los algoritmos en tiempo real explorados en [Kwok et al. (2004)] pueden ser útiles para varios miles de partículas solamente. Recurriendo a un circuito integrado dedicado, se pretende lograr filtros de partícula más rápidos y con más partículas. Un cálculo aproximado indica que utilizando un proceso 3D actualmente disponible con 2Gb de memoria de DRAM en la pastilla, un Filtro de Partículas con varios millones de partículas podría ser factible.

Para una aplicación concreta, el algoritmo presentado debe complementarse algoritmos de sincronización de la red, de localización de los nodos dentro de la misma y de selección del nodo vecino mas informativo. Se puede extender el algoritmo para detección de múltiples fuentes.

# Bibliografía

Lms 200/211/221/291 laser measurement systems. URL  
[sicktoolbox.sourceforge.net/docs/sick-lms-technical-description.pdf](http://sicktoolbox.sourceforge.net/docs/sick-lms-technical-description.pdf).

Sick. *www.sick.com*.

F. Abdallah, A. Gning, and P. Bonnifait. Box particle filtering for nonlinear state estimation using interval analysis. *Automatica*, 44:807–815, 2008.

K. Akkaya and M. Younis. A survey on routing protocols for wireless sensor networks. *Ad Hoc Networks*, (3):325–349, 2005.

I. F. Akyldiz, T. Melodia, and K. R. Chowdhury. Wireless multimedia sensor networks: Applications and testbeds. *Proceedings of the IEEE*, 96(10):1588–1605, 2008.

D. Alspach and H. Sorenson. Recursive bayesian estimation using gaussian sum approximations. *Automatica*, 7:465–479, 1971.

D. Alspach and H. Sorenson. Nonlinear bayesian estimation using gaussian sum approximations. *Automatic Control, IEEE Transactions on*, 17(4):439 – 448, 1972. ISSN 0018-9286.

A. Arora, P. Dutta, S. Bapat, V. Kulathumani, H. Zhang, V. Naik, V. Mittal, H. Cao, M. Demirbas, M. Gouda, Y. Choi, T. Herman, S. Kulkarni, U. Arumugam, M. Nesterenko, A. Vora, and M. Miyashita.

- M. S. Arulampalam, S. Maskell, N. Gordon, and T. Clapp. A tutorial on particle filters for online nonlinear/non-gaussian bayesian tracking. *Signal Processing, IEEE Transactions on*, 50(2):174–188, 2002. URL <http://dx.doi.org/10.1109/78.978374>.
- S. Bhowmik and C. Roy. Comparison of estimation techniques using kalman filter and grid-based filter for linear and non-linear system. In *Proceedings of the International Conference on Computing: Theory and Applications*, pages 516–520, Washington, DC, USA, 2007. IEEE Computer Society. ISBN 0-7695-2770-1. URL <http://portal.acm.org/citation.cfm?id=1260199.1260291>.
- D. Blatt and A. O. Hero. Energy-based sensor network source localization via projection onto convex sets. *IEEE Transaction on Signal Processing*, 54:3614–3619, 2006.
- M. Bolić, P. M. Djurić, and S. Hong. New resampling algorithms for particle filters. *Acoustics, Speech, and Signal Processing Proceedings ICASSP on IEEE International Conference*, 2(2):589–592, 2003.
- G. E. P. Box, W. G. Hunter, J. S. Hunter, and W. G. Hunter. *Statistics for Experimenters: An Introduction to Design, Data Analysis, and Model Building*. John Wiley & Sons, June 1978. ISBN 0471093157. URL <http://www.amazon.com/exec/obidos/redirect?tag=citeulike07-20&path=ASIN/0471093157>.
- E. Brewer, D. Culler, D. Gay, P. Levis, R. von Behren, and M. Welsh. nesc: A programming language for deeply networked systems. URL <http://nesc.sourceforge.net/>.
- F. Campillo and V. Rossi. Convolution particle filters for parameter estimation in general state-space models. *ResearchReport5939,INRIA,062006*, 2006.
- J. Carpenter, P. Clifford, and P. Feamhead. Improved particle filter for nonlinear problems, radar, sonar and navigation. *IEEE Proceedings*, 146(1), 1999.

- P. A. Challenges, J. A. Stankovic, Q. Cao, T. Doan, L. Fang, Z. He, R. Kiran, S. Lin, S. Son, R. Stoleru, and A. Wood. Wireless sensor networks for in-home healthcare. In *in HCMDSS*, pages 2–3, 2005. URL <http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.60.5768>.
- K. Chebrolu, B. Raman, N. Mishra, P. Valiveti, and R. Kumar. Brimon: A sensor network system for railway bridge monitoring. *MobiSys*, 2008. URL [www.cs.toronto.edu/~delara/courses/csc2228/papers/brimon.pdf](http://www.cs.toronto.edu/~delara/courses/csc2228/papers/brimon.pdf).
- J. Chen, S. Kher, and A. Somani. Distributed fault detection of wireless sensor networks. *Dependability Issues in Wireless Ad hoc Networks and Sensor networks (DIWANS'06)*, 2006.
- Z. Chen. Bayesian filtering: From kalman filters to particle filters, and beyond. *Technical Report, Adaptive Systems Laboratory, McMaster University*, 2003.
- C.-Y. Chong and S. P. Kumar. Sensor networks: evolution, opportunities, and challenges. *Proceedings of the IEEE*, 91(8):1247–1256, 2003. URL <http://dx.doi.org/10.1109/JPROC.2003.814918>.
- M. Coates. Data fusion in decentralized sensing networks. *4th International Conference on Information Fusion, Montreal, Canada*, 2001.
- F. Daum. Nonlinear filters: Beyond the kalman filter. *IEEE Aerospace and Electronics Systems Magazine*, 20(8):57–69, August 2005.
- R. Douc, E. Moulines, and J. Olsson. On the auxiliary particle filter. *Technical Report 0709.3448v1, arXiv*, 2007. URL <http://arxiv.org/abs/0709.3448>.
- H. Durrant-Whyte and T. Bailey. Simultaneous localization and mapping: part i. *IEEE Robotics and Automation Magazine*, 13(2):99–108, 2006.
- H. Durrant-Whyte and M. Stevens. Data fusion in decentralized sensing networks. *4th International Conference on Information Fusion, Montreal, Canada*, 2001.

- D. Estrin, R. Govindan, J. Heidemann, and S. Kumar. Next century challenges: Scalable coordination in sensor networks. *Proceedings International Conference Mobile Computing and Networking (MOBICOM)*, pages 263–270, 1999.
- D. Fox. Adapting the sample size in particle filters through kld-sampling. *International Journal of Robotics Research*, 22, 2003. URL <http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.5.6774>.
- D. Gay, P. Levis, R. von Behren, M. Welsh, E. Brewer, and D. Culler. The nesc language: A holistic approach to networked embedded systems. *Proceedings of Programming Language Design and Implementation (PLDI) 2003*, 2003.
- H. Gharavi and S. Kumar. Special issue on sensor networks and applications. *Proceeding of IEEE Second Edition*, 91(8):1151–1153, 2003.
- A. Gning, F. Abdallah, and P. Bonnifait. A new estimation method for multisensor fusion by using interval analysis and particle filtering. *IEEE International Conference on Robotics and Automation*, pages 3844–3849, 2007.
- N. J. Gordon, D. J. Salmond, and A. F. M. Smith. Novel approach to nonlinear/non-gaussian bayesian state estimation. *Radar and Signal Processing, IEE Proceedings F*, 140(2):107–113, August 2002a. URL [http://ieeexplore.ieee.org/xpls/abs\\_all.jsp?arnumber=210672](http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=210672).
- N. J. Gordon, D. J. Salmond, and A. F. M. Smith. Novel approach to nonlinear/non-gaussian bayesian state estimation. *Radar and Signal Processing, IEE Proceedings F*, 140(2):107–113, August 2002b. URL [http://ieeexplore.ieee.org/xpls/abs\\_all.jsp?arnumber=210672](http://ieeexplore.ieee.org/xpls/abs_all.jsp?arnumber=210672).
- H. Green. Tech wave 2: The sensor revolution. *Business Week*, 2003.
- T. Haenselmann, T. King, M. Busse, W. Effelsberg, and M. Fuchs. *Emerging Directions in Embedded and Ubiquitous Computing*, volume 4809 of *Lecture Notes in Computer Science*. Springer Berlin / Heidelberg, copyright 2007 edition, 2007. ISBN 978-3-540-77089-3.

- J. Handschin. Monte carlo techniques for prediction and filtering of non-linear stochastic processes. *Automatica*, 6(4):555 – 563, 1970. ISSN 0005-1098. URL <http://www.sciencedirect.com/science/article/B6V21-47X16R1-3B/2/2e31c5f79d17f1dc6d18859deaa42e0d>.
- T. He, C. Huang, B. Blum, J. Stankovic, and T. Abdelzaher. Range-free localization schemes for large scale sensor networks. *Proc. IEEE Mobicom 2003*, pages 81 – 95, 2003.
- W. R. Heinzelman, A. Chandrakasan, and H. Balakrishnan. Energy-efficient communication protocol for wireless microsensor networks. *33rd Hawaii International Conference on System Sciences*, 8, 2000.
- S. Hong, M. Bolic, and P. Djuric. An efficient fixed-point implementation of residual resampling scheme for high-speed particle filters. *Signal Processing Letters, IEEE*, 11:482–485, 2004.
- S. Hong, M. Bolic, and P. Djuric. Resampling algorithms and architectures for distributed particle filters. *IEEE Transactions on Signal Processing*, 53:2442– 2450, 2005.
- A. T. Ihler, J. W. Fisher, and A. S. Willsky. Particle filtering under communications constraints. *Statistical Signal Processing, 2005 IEEE/SP 13th Workshop on*, pages 89–94, 2005. URL <http://dx.doi.org/10.1109/SSP.2005.1628570>.
- V. Isler and R. Bajcsy. The sensor selection problem for bounded uncertainty sensing models, 2005. URL <http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.61.1132>.
- R. E. Kalman and R. S. Bucy. New results in linear filtering and prediction theory. *Transactions of the ASME. Series D, Journal of Basic Engineering*, 83:95–107, 1961.
- L. M. Kaplan. Global node selection for localization in a distributed sensor network. *Aerospace and Electronic Systems, IEEE Transactions on*, 42(1):113–135, 2006. URL <http://dx.doi.org/10.1109/TAES.2006.1603409>.

- A. Kong, J. S. Liu, and W. H. Wong. Sequential imputations and bayesian missing data problems. *Journal of the American Statistical Association*, 89(425):278–288, 1994. ISSN 01621459. URL <http://www.jstor.org/stable/2291224>.
- J. Kotecha and P. Djuric. Gaussian sum particle filtering for dynamic state space models. *Proceedings International Conference in Acoustics, Speech, Signal Processing, Salt Lake City, UT*, 2001.
- J. H. Kotecha and Djuris. Gaussian particle filtering. *IEEE Transaction on signal processing*, 51, 2003.
- S. Kumar, R. Govindan, D. Estrin, and J. Heidemannand. Next century challenges: Scalable coordination in sensor networks. *Proceedings of the ACM/IEEE International Conference on Mobile Computing and Networking, Seattle, Washington, USA*, pages 263–270, 1999.
- S. P. Kumar and C. Chee-Yee. Sensor networks: evolution, opportunities, and challenges. *Proceedings of the IEEE*, 91(8):1247–1256, 2003.
- C. Kwok, D. Fox, and M. Meila. Resampling algorithms and architectures for distributed particle filters. *Proceedings of the IEEE*, 92:469– 484, 2004.
- R. T. Lacoss. Distributed mixed sensor aircraft tracking. *American Control Conference Minneapolis*, 1987.
- H. Lee and H. Aghajan. Vision-enabled node localization in wirelwss sensor networks. *COGNitive systems with Interactive Sensors (COGIS'06), Paris, France*, 2006.
- P. Levis, S. Madden, J. Polastre, R. Szewczyk, K. Whitehouse, A. Woo, D. Gay, J. Hill, M. Welsh, E. Brewer, and D. Culler. Tinyos: An operating system for sensor networks. pages 115–148. 2005. URL [http://dx.doi.org/10.1007/3-540-27139-2\\_7](http://dx.doi.org/10.1007/3-540-27139-2_7).
- Y. L. Li, M. J. White, and S. J. Franke. New fast field programs for anisotropic sound propagation through an atmosphere with a wind velocity profile. *The Journal of the Acoustical Society of America*, 95:718–726, 1994.

- J. S. Liu and R. Chen. Sequential monte carlo methods for dynamic systems. *Journal of the American Statistical Association*, 93(443):1032–1044, 1998. URL [www.jstor.org/stable/2669847](http://www.jstor.org/stable/2669847).
- L. Ljung. Asymptotic behavior of the extended kalman filter as a parameter estimator for linear systems. *Automatic Control, IEEE Transactions on*, 24(1):36–50, 1979. ISSN 0018-9286.
- S. N. MacEachern, M. Clyde, and J. S. Liu. Sequential importance sampling for nonparametric bayes models: The next generation. *The Canadian Journal of Statistics / La Revue Canadienne de Statistique*, 27(2):251–267, 1999. URL [www.jstor.org/stable/3315637](http://www.jstor.org/stable/3315637).
- A. Mainwaring, D. Culler, J. Polastre, R. Szewczyk, and J. Anderson. Wireless sensor networks for habitat monitoring. In *WSNA '02: Proceedings of the 1st ACM international workshop on Wireless sensor networks and applications*, pages 88–97, New York, NY, USA, 2002a. ACM. ISBN 1-58113-589-0. URL <http://dx.doi.org/10.1145/570738.570751>.
- A. Mainwaring, D. Culler, J. Polastre, R. Szewczyk, and J. Anderson. Wireless sensor networks for habitat monitoring. In *WSNA '02: Proceedings of the 1st ACM international workshop on Wireless sensor networks and applications*, pages 88–97, New York, NY, USA, 2002b. ACM. ISBN 1-58113-589-0. URL <http://dx.doi.org/10.1145/570738.570751>.
- B. B. Manjunath, A. Papandreou-Suppappola, C. Chakrabarti, and D. Morrell. Computationally efficient particle filtering using adaptive techniques. *Sensor, Signal and Information Processing Workshop*, 2008.
- M. Marioti, B. Kusy, G. Simon, and A. Lédeczi. The flooding time synchronization protocol. *SynSys04*, 2004.
- F. Mourad, H. Snoussi, F. Abdallah, and C. Richard. Guaranteed boxed localization in manets by interval analysis and constraints propagation techniques. *IEEEGlobeCom*, 2008.



- C. E. Nishimura and D. M. Conlon. Uss dual use: Monitoring whales and earthquakes using sosus. *Marine Technology Society Journal*, 27(4):13–21, 1994.
- S. Park, J. Hwang, K. Rou, and E. Kim. A new particle filter inspired by biological evolution: Genetic filter. *World Academy of Science, Engineering and Technology*, 33:83–87, 2007.
- N. Patwari, A. H. III, M. Perkins, N. Correal, and R. O’Dea. Relative location estimation in wireless sensor networks. *IEEE Trans. Signal Processing*, 51(8):2137 – 2148, 2003.
- Pioneer. URL <http://www.mobilerobots.com>.
- J. Polastre, R. Szewczyk, A. Mainwaring, D. Culler, and J. Anderson. *Wireless Sensor Networks.*, chapter Analysis of Wireless Sensor Networks for Habitat Monitoring, pages 399–423. Number VI. Springer US, copyright 2004 edition, 2006. ISBN 978-0-387-35269-5 (Print) 978-1-4020-7884-2 (Online).
- G. Pottie and W. Kaiser. Wireless integrated network sensors. *Communications of the Association for Computing Machinery*, 43(5), 2004.
- L. R. Rabiner and B. H. Juang. An introduction to hidden markov models. *IEEE Acoustics, Speech, Signal Processing Magazine*, pages 4–16, 1986.
- B. Raman and K. Chebrolu. Censor networks: A critique of "sensor networks" from a systems perspective. *Association for Computing Machinery Special Interest Group on Data Communications (ACM SIGCOMM) Computer Communication Review*, 3(38):75–78, 2008.
- F. Reichenbach, J. Blumenthal, and D. Timmermann. Comparing the efficiency of localization algorithms with the power-error-product (pep). *Second IEEE International Workshop on Wireless Mesh and Ad Hoc Networks (WiMAN)*, 2008.
- B. Sadler. Fundamentals of energy~constrained sensor network systems. *Aerospace and Electronic Systems Magazine, IEEE On Part.2: Tutorials*, 20(8):17–35, 2005.

- E. M. Salomons. Downwind propagation of sound in an atmosphere with realistic sound-speed profile: A semianalytical ray model. *The Journal of the Acoustical Society of America*, 95:2425–2436, 1994.
- S. R. Sañudo, F. Masson, and P. Julian. Bounded state space particle filter for network sensors. *ISCAS 2007 IEEE International Symposium on Circuits and Systems, Sensory Systems I Lecture session C4L-M*, pages 3570–3573, 2007.
- S. R. Sañudo, F. Masson, and P. Julian. Localization and tracking filter algorithm for network sensors. *Enviado a: Journal of Engineering and Technology Research, Academic Journals*, 2011.
- S. Sañudo and F. Masson. Filtro de partículas acotado para fusión de datos en redes de sensores. *Argentinian Robotics Journals (JAR)*, 2006a. URL <http://www.frc.utn.edu.ar/jar2006/docs/Papers/020-jar06.pdf>.
- S. R. Sañudo and F. Masson. Análisis de un algoritmo de sincronismo para nodos de una red de sensores. *AADECA 2006 XXº Congreso Argentino de Control Automático*, (123), 2006b.
- S. Schmid and R. Wattenhofer. Algorithmic models for sensor networks. *Proceedings of the 14th International Workshop on Parallel and Distributed Real-Time Systems*, 2004.
- X. Sheng and Y. Hu. Distributed particle filters for wireless sensor network target tracking. *IEEE International Conference on Acoustics, Speech and Signal Processing Proceedings (ICASSP'05)*, 4:845–848, 2005.
- F. Sivrikaya and B. Yener. Time synchronization in sensor networks: A survey. *IEEE Network*, (4):45 – 50, 2004.
- S. Sondon, G. Stuarts, P. Mandolesi, F. Masson, and P. Julian. Nodo dba para medición de polución acústica con redes de sensores. *Artículo Breve Escuela Argentina de Microelectrónica 2007 (EAMTA07)*, 1:29–32, 2007a.

- S. Sondon, G. Stuarts, P. Mandolesi, F. Masson, and P. Julian. Nodo dba para medición de polución acústica con redes de sensores. *XII Reunión de Trabajo en Procesamiento de la Información y Control (XII RPIC 2007)*, 2007b.
- D. Steere, A. Baptista, D. McNamee, C. Pu, and J. Walpole. Research challenges in environmental observation and forecasting systems. *Proceedings 6th International Conference Mobile Computing and Networking (MOBICOMM)*, pages 292–299, 2000.
- S. Subramaniam, T. Palpanas, D. Papadopoulos, V. Kalogeraki, and D. Gunopulos. Online outlier detection in sensor data using non parametric models. *Very Large Data Base Endowment (VLDB'06), Seoul, Korea*, 2006.
- I. The MathWorks. The mathworks. 1994-2010. URL <http://www.mathworks.com/>.
- S. A. Tunhein and C. AS. Implementation of cmos low cost and low power rf-ics. *Proceeding Wireless Systems Design Conference and Expo*, 2002. San Jose, CA, US.
- T.Zhao and A.Nehorai. Distributed sequential bayesian estimation of a diffusive source in wireless sensor networks. *IEEE Transactions on Signal Processing*, 55(4):1511–1524, 2007.
- E. A. Wan and R. Van Der Merwe. The unscented kalman filter for nonlinear estimation. *Proceedings of Symposium 2000 on Adaptive Systems for Signal Processing, Communication and Control(AS-SPCC) IEEE, LakeLouise, Alberta, Canada*, pages 153–158, 2000.
- B. Week. 21 ideas for the 21st century. *Business Week*, pages 78–167, 1999.
- G. Welch and G. Bishop. An introduction to the kalman filter. Technical report, Chapel Hill, NC, USA, 1995a. URL <http://portal.acm.org/citation.cfm?id=897831>.
- G. Welch and G. Bishop. An introduction to the kalman filter. *University of North Carolina, Department of Computer Science, TR 95-041*, 1995b.

- C.-H. Wu, K.-C. Lee, and Y.-C. Chung. A delaunay triangulation based method for wireless sensor network deployment. In *ICPADS '06: Proceedings of the 12th International Conference on Parallel and Distributed Systems*, pages 253–260, Washington, DC, USA, 2006. IEEE Computer Society. URL <http://dx.doi.org/10.1109/ICPADS.2006.11>.
- J. Yick, B. Mukherjee, and D. Ghosal. Wireless sensor network survey. *Computer Networks*, 12(52):2292–2330, 2008.
- V. S. Zaritskii, V. B. Svetnik, and L. I. Shimelevich. Monte-carlo technique in problems of optimal information processing. *Automation and Remote Control*, 36(12):2015–2022, 1976.