

Construcción de Mapas y Localización Simultánea con Robots Móviles

Tesis de Maestría en Ciencias Computacionales

Víctor Manuel Jáquez Leal

ceyusa@gmail.com

ITESM Campus Cuernavaca Temixco, Morelos, México

Agenda

- Definición del problema.
- Formalización del problema.
- Filtros de partículas.
- Aplicación de los FP en el SLAM.
- Fusión sensorial y exploración.
- Arquitectura de software.

¿Para qué sirven los mapas?

1. Localización.
2. Planeación de movimientos.
3. Evitar obstáculos.
4. Uso humano.
5. Una combinación de cualquiera de los anteriores.

La cartografía robótica

- **Problema:** adquisición de un modelo del ambiente utilizando un robot móvil.
- El robot únicamente cuenta con sus sensores.
- El mapa obtenido es más útil que uno hecho con intervención humana debido a la limitaciones perceptuales.
- Se persigue una verdadera autonomía del robot.

Tipos de mapas

- *Geométricos*: Representan el ambiente por medio de primitivas geométricas.
- *Topológicos*: Representa al espacio como un conjunto de grafos.
- *Híbridos*: Mezclan las representaciones topográficas con información métrica en cada nodo.

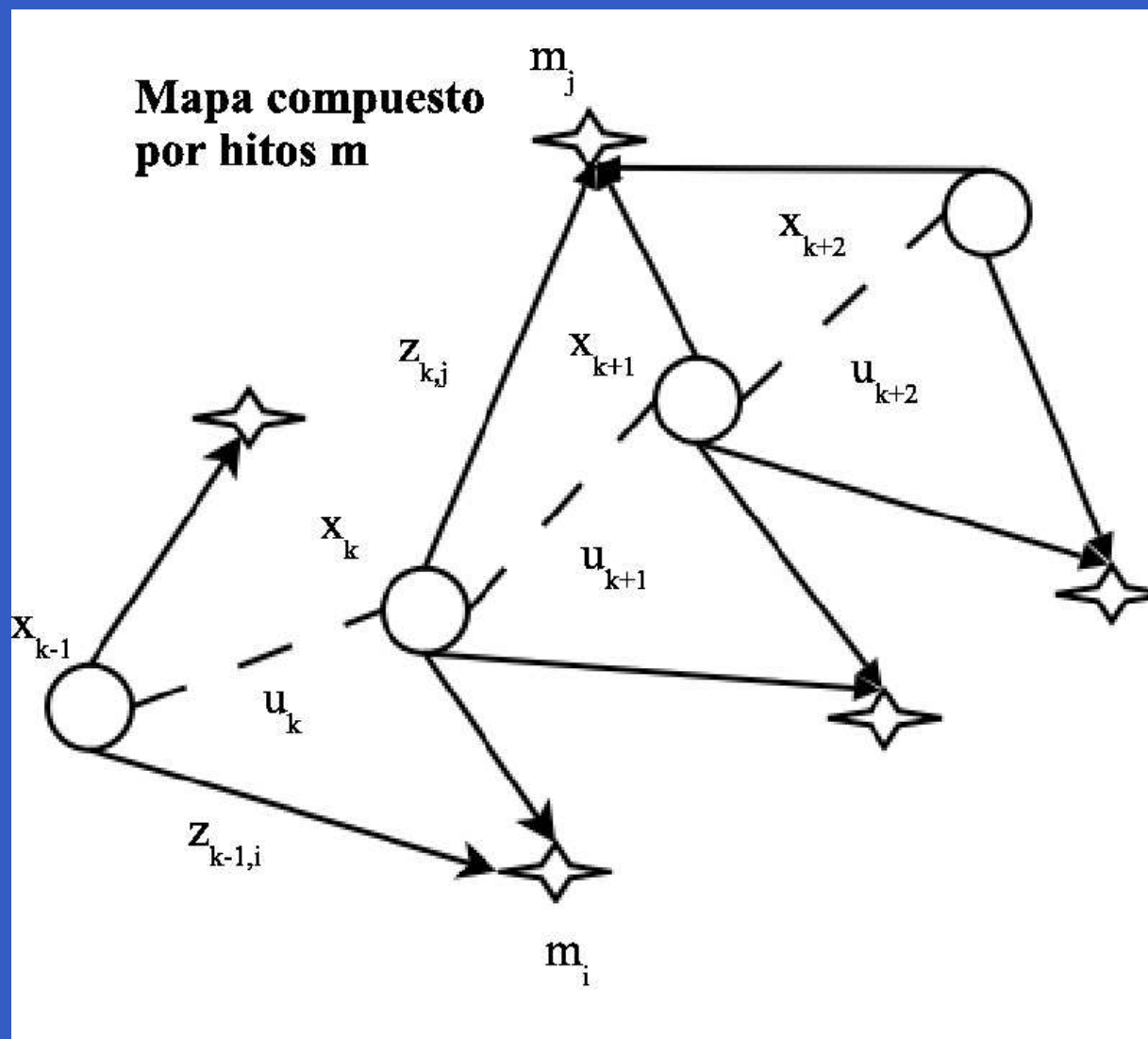
Mapas geométricos

- *Descripción explícita del espacio libre o modelo de rejilla.* El espacio se discretiza en celdas que tienen una probabilidad de ocupación.
- *Descripción explícita de objetos.* Son elementos geométricos dentro del espacio con una posición probabilista.
- *Mapa del terreno.* Expone la altitud de los puntos observados, además de marcar oclusiones existentes.

Limitaciones de los sensores

- Los sensores de distancia tiene percepción limitada.
- El robot debe desplazarse para registrar áreas desconocidas.
- La posición del robot ofrece la posición de observación.

El problema cartográfico



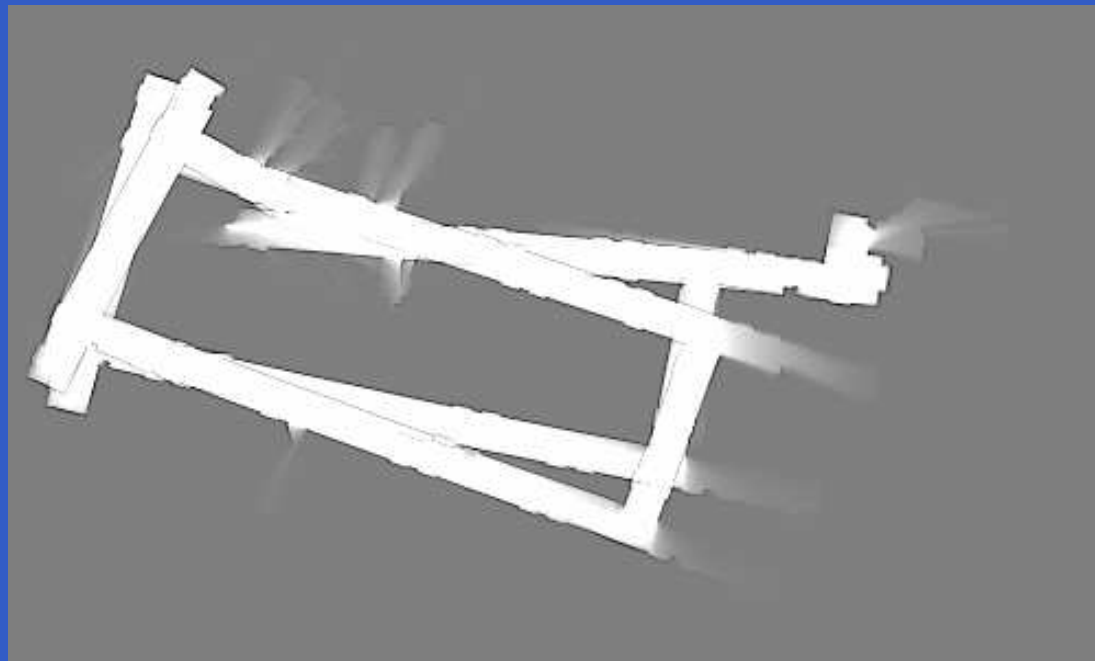
Ruido en los sensores

- Los sensores están sujetos a ruido.
- El error odométrico es acumulativo.

Ruido en los sensores

- Los sensores están sujetos a ruido.
- El error odométrico es acumulativo.

¡Hay que hacer localización del robot!



Cartografía y Localización Simultánea

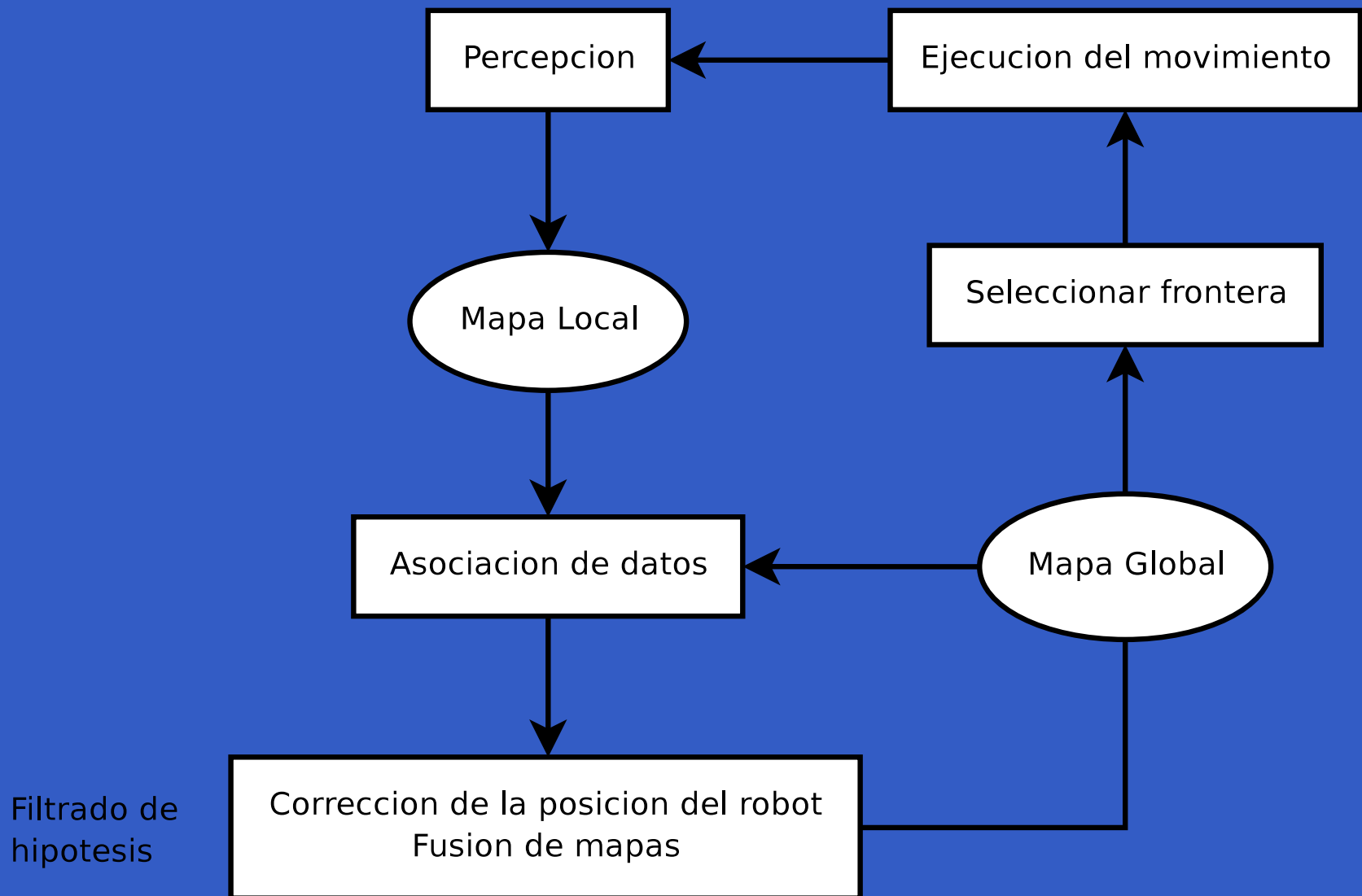
- Para actualizar la parte del mapa visible hay que localizarse.
- Para localizarse se requiere un mapa previo ya que no se confía en la odometría.
- Hay que resolver ambas incógnitas de manera simultánea.

SLAM: Simultaneous Localization And Mapping

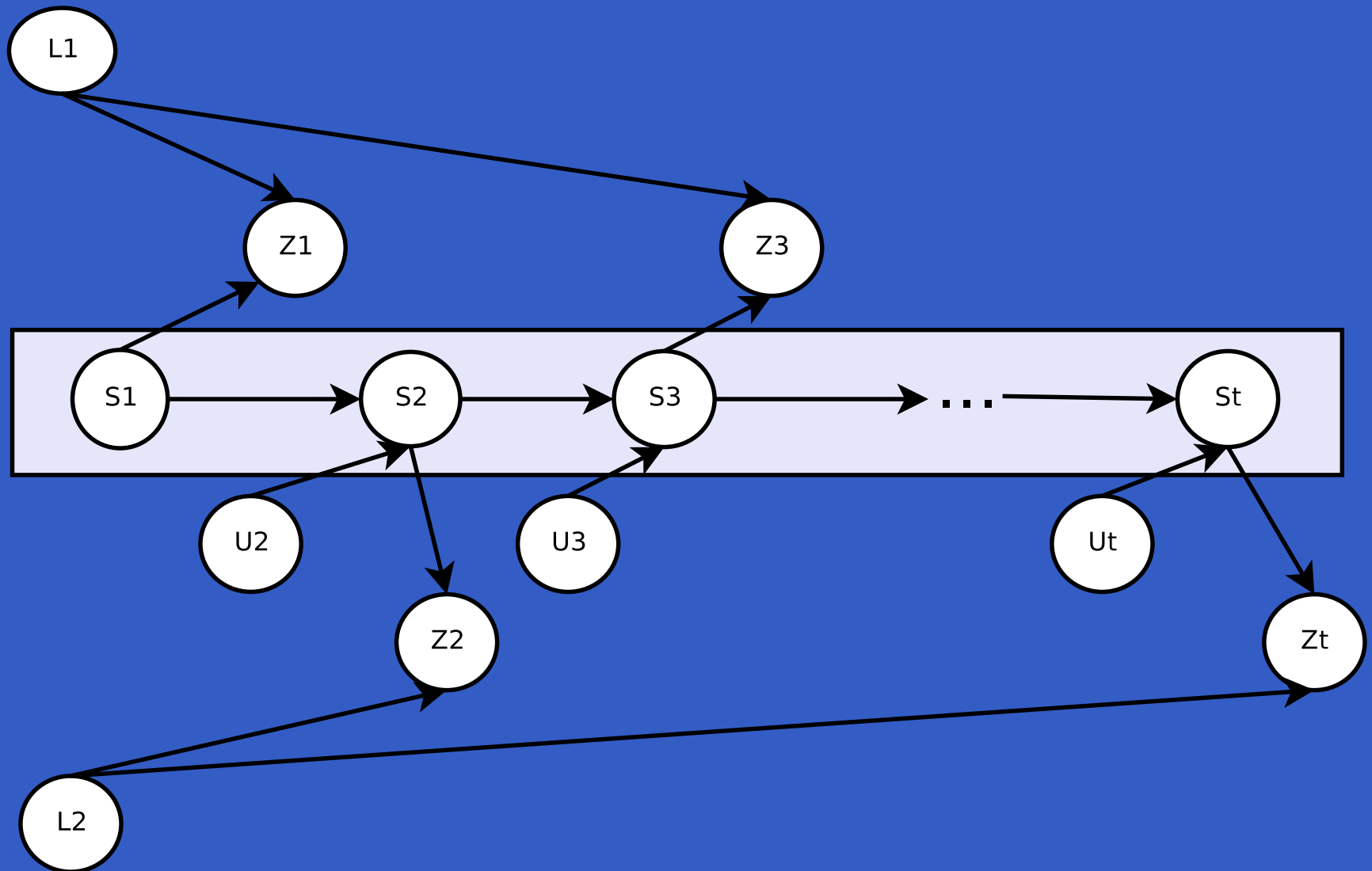
Otros problemas

- Alta dimensionalidad del espacio.
¿Cuántos elementos necesito para describir el espacio?
- Asociación de correspondencias.
¿Esto ya lo vi o es algo nuevo?
- Ambientes dinámicos.
¡Antes había algo ahí!
- Exploración.
¿Y ahora a dónde me voy?

SLAM Incremental



SLAM como DBN



Formalización

Al formularse el problema del SLAM como una DBN, se utiliza un filtro Bayesiano.

$$p(s_t, m | z^t, u^t) = \eta p(z_t | s_t, m) \int p(s_t | u_t, s_{t-1}) p(s_{t-1}, m | z^{t-1}, u^{t-1}) ds_{t-1}$$

A $p(z | s_t, m)$ se le llama *modelo de percepción*

A $p(s | u, s_{t-1})$ se le llama *modelo del movimiento*

Esta ecuación no puede ser implementada tal y como está.

Inferencias de filtrado

- La inferencia exacta con distribuciones continuas o mixtas no existe.
- Se usan algoritmos que buscan aproximaciones.
 - Extended Kalman Filters (EKFs)
 - Sparse Extended Information Filters (SEIFs)
 - Thin Junction Trees
 - **Particle Filters**

Ventajas del filtro de partículas

- No paramétrico (distribuciones no lineales y multimodales)
- Es adaptativo: más partículas al aumentar la densidad probabilista.
- Tienen poco costo computacional - Pueden llegar a $O(M)$.
- Pueden ser aplicados en RTS con algoritmos Any-Time.
- Más fáciles de implementar que los EKF's.

Métodos Monte Carlo

- Asumen aleatoriamente la existencia de valores para algunos nodos en la DBN.
- Usan estos valores aleatorios para inferir la cantidad de los otros nodos.
- Mantienen estadísticas de los valores que toman los nodos.
- Al final las estadísticas dan la respuesta.

Filtro de Partículas

La idea es aproximar el valor del estado a través de un conjunto de partículas ponderadas o muestras:

$$p(x_{0:k} | z_{1:k}) \approx \sum_{i=1}^{N_s} w_k^i \delta(x_{0:k} - x_{0:k}^i)$$

$x_{0:k}^i, \{i = 0, \dots, N_s\}$ Conjunto de puntos aleatorios.

$w_k^i, \{i = 1, \dots, N_s\}$ Pesos asociados a cada punto.

$x_{0:k}, \{x_j, j = 0, \dots, k\}$ Conjunto de estados hasta k .

Ponderación de las partículas

Suponiendo $p(x) \propto \pi(x)$

Dónde

$p(x)$ es una distribución difícil de muestrear.

$\pi(x)$ es una distribución que se puede evaluar.

$q(x)$ es una distribución propuesta.

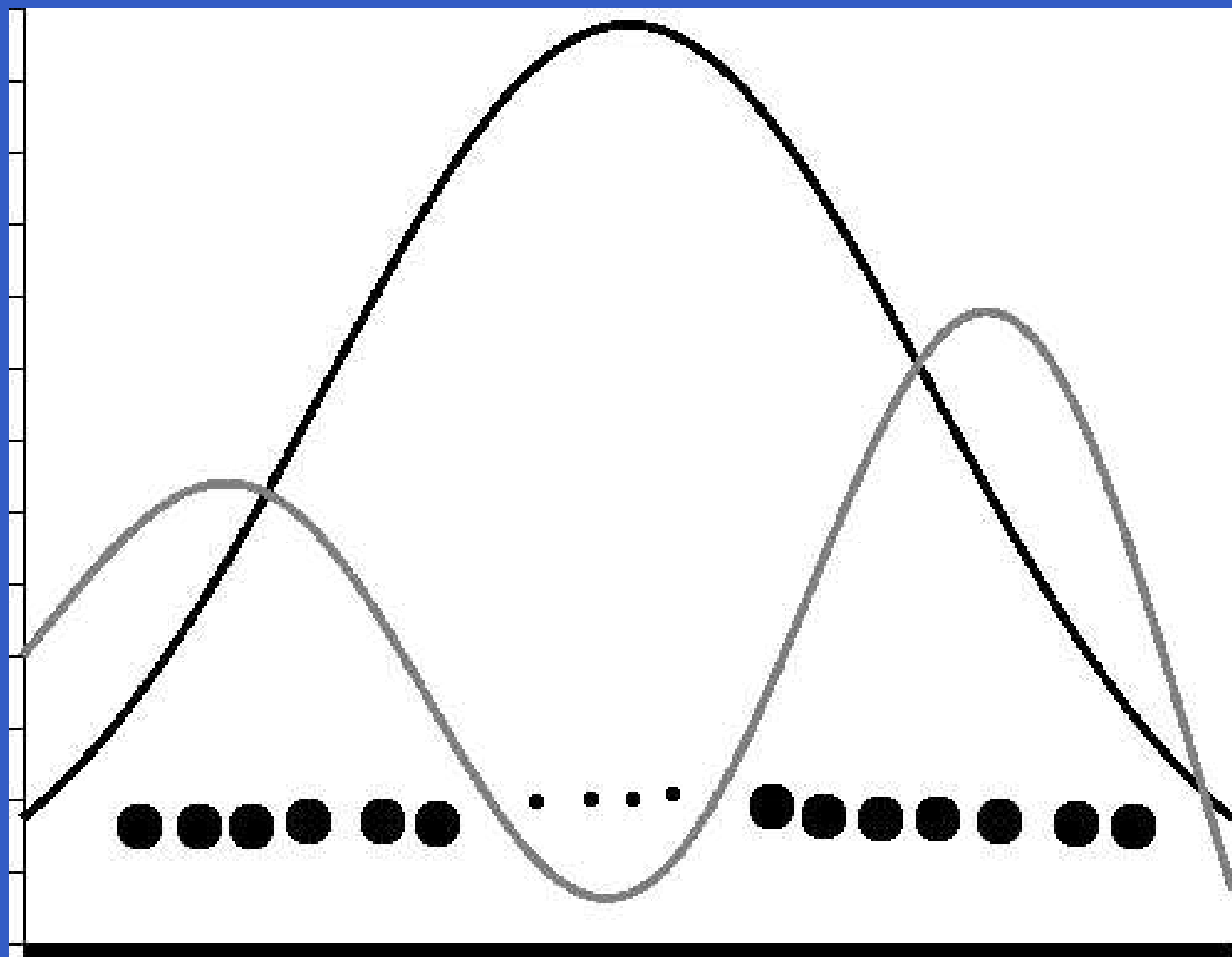
$x^i \sim q(x), \{i = 1, \dots, N_s\}$ son muestras de la distribución propuesta.

Entonces

$$w^i \propto \frac{\pi(x^i)}{q(x^i)}$$

Los pesos están normalizados de tal forma que $\sum_i w_k^i = 1$.

Filtros de partículas



Ponderación de las partículas

El caso secuencial (BDN) se pueden obtener los pesos de las nuevas partículas a partir de sus pesos anteriores.

$$(1) \quad w_k^i \propto w_{k-1}^i \frac{p(z_k | x_k^i) p(x_k^i | x_{k-1}^i)}{q(x_k^i | x_{k-1}^i, z_k)}$$

Algoritmo del filtro de partículas

$$[\{x_k^i, w_k^i\}_{i=1}^{N_s}] = \text{SIS}[\{x_{k-1}^i, w_{k-1}^i\}_{i=1}^{N_s}, z_k]$$

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

Esboza $x_k^i \sim q(x_k \mid x_{k-1}^i, z_k)$

Pondera partícula, w_k^i de acuerdo a la ecuación (1)

end for

Degeneración de la partícula

Después de varias iteraciones todas las partículas, excepto una, tendrán un peso despreciable.

Una medición de dicha degeneración es el tamaño efectivo de la muestra N_{eff} :

$$(2) \quad \widehat{N}_{eff} = \frac{1}{\sum_{i=1}^{N_s} (w_k^i)^2}$$

Un N_{eff} pequeño indica un empobrecimiento severo.

Evitando la degeneración

Hay varios enfoques para evitar el problema de la degeneración:

- Utilizar un N_s muy grande (fuerza bruta). Es muy impráctico.
- La elección de una buena densidad de importancia que minimice $Var(w_k^{*i})$.
- **El remuestreo.**

Remuestreo

- Usar remuestreo siempre que N_{eff} llega a ser menor que un umbral.
- La idea es eliminar partículas que tienen pesos bajos y concentrarse en las partículas con pesos altos.

Remuestreo

- Generar de un nuevo conjunto $\{x_k^{i*}\}_{i=1}^{N_s}$
- Remuestreando, con reemplazo, N_s veces a partir de $p(x_k | z_{1:k})$
- De tal manera que $Pr(x_k^{i*} = x_k^j) = w_k^j$.
- Finalmente los pesos estarán reasignados a $w_k^i = 1/N_s$.

Algoritmo general de filtros de partículas

$$[\{x_k^i, w_k^i\}_{i=1}^{N_s}] = \text{PF}[\{x_{k-1}^i, w_{k-1}^i\}_{i=1}^{N_s}, z_k]$$

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

Esboza $x_k^i \sim q(x_k | x_{k-1}^i, z_k)$

Pondera partícula, w_k^i de acuerdo a (1)

end for

Calcular el peso total: $t = \sum_{i=1}^{N_s} w_k^i$

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

Normalizar: $w_k^i = w_k^i / t$

end for

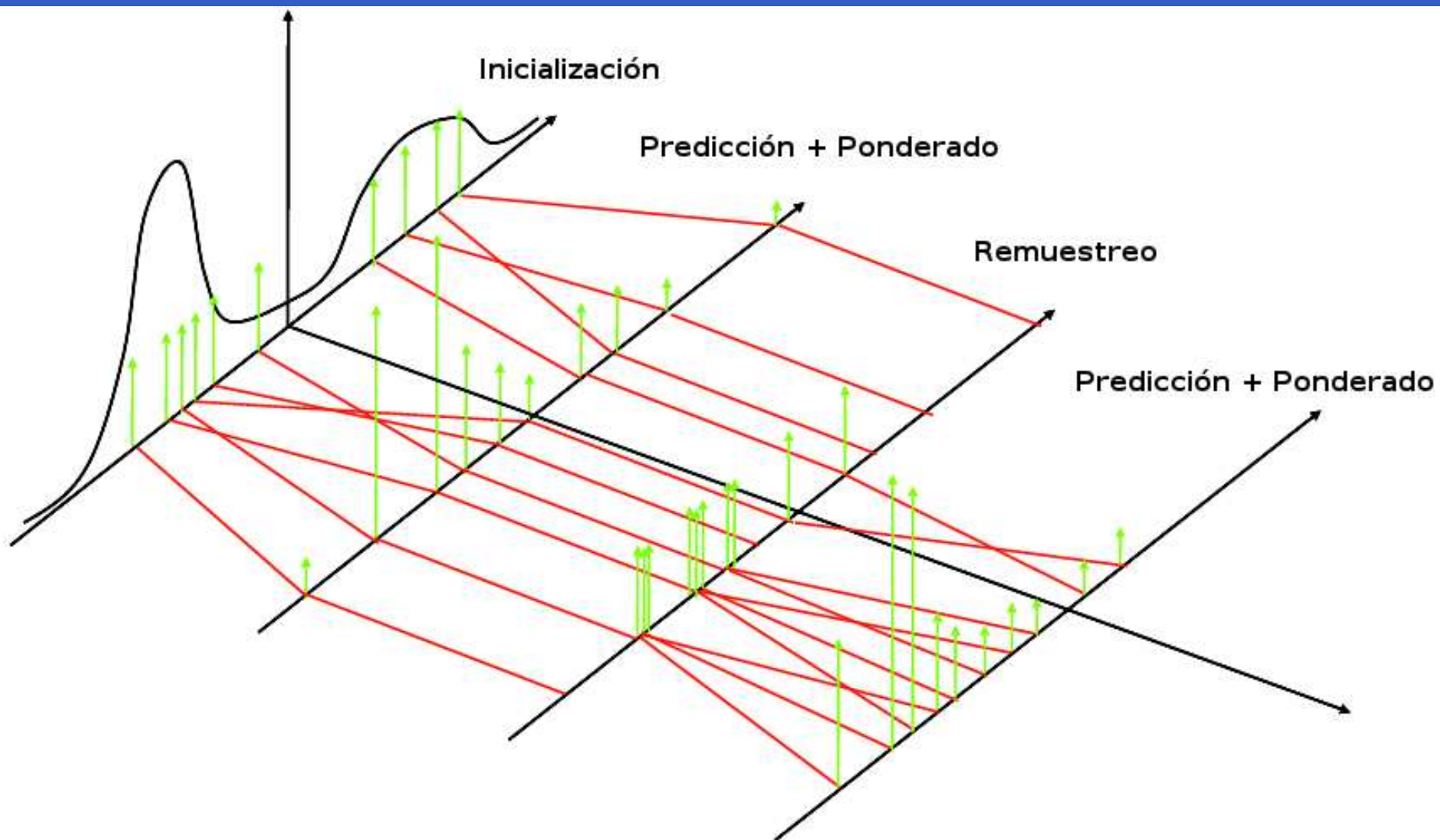
Calcular \widehat{N}_{eff} usando (2)

if $\widehat{N}_{eff} < N_T$ **then**

Remuestrear

end if

Filtros de partículas



FP Rao-Blackwellizados

- Los filtros de partículas pueden ser muy ineficientes en espacios de alta dimensionalidad.
- El *teorema de Rao-Blackwell* dice cómo mejorar un estimador.
- Dividir el espacio de estados en dos subespacios: uno se actualiza analíticamente y el otro con filtros de partículas.
- En una búsqueda de una distribución conjunta $p(m, s \mid u, z)$, se marginaliza una variable m .

Implementando SLAM con PF

La implementación para resolver el problema del SLAM con PF dependen de tres factores:

- Representación del ambiente
- El modelo de movimiento del robot
- El modelo de percepción de los sensores

Rejilla de ocupación

- Es el enfoque más simple.
- La asociación de correspondencias es “natural”.
- Consume mucho espacio en memoria.
- Se pierden detalles de los objetos que componen el ambiente.

Contenido de las partículas

- Posición actual del robot.
- Trayectoria seguida por el robot según la partícula.
- Mapa propuesto por la trayectoria.

De manera global se llevan todas las observaciones.

Modelo de movimiento - Predicción

- Representa la distribución propuesta.
- Estima la trayectoria posterior $p(s_t | s_{t-1}, u^t)$.

Para cada partícula:

$$\Delta pos_t.x = \mathcal{N}(0, k_x * \Delta pos_{t-1}.x)$$

$$\Delta pos_t.y = \mathcal{N}(0, k_y * \Delta pos_{t-1}.y)$$

$$\Delta pos_t.\theta = \mathcal{N}(0, k_\theta * \Delta pos_{t-1}.\theta)$$

donde las k_i son coeficientes ad hoc.

Modelo de observación - Actualización

Cada partícula j esbozada por el modelo de movimiento es ponderada de acuerdo a:

$$w_{t+1}^j = v_{t+1}^j + w_t^j$$

donde

$$v_{t+1}^j \propto p(z_{t+1} \mid s_{t+1}, m_t^j)$$

v_t^j se obtiene de una función de vecindad de cada celda marcada como obstáculo.

Mientras más coheccionados estén las celdas que constituyen un obstáculo, mayor es el valor de v_t^j .

Factible gracias al telémetro láser.

Remuestreo

- Se hace remuestreo cada vez que el N_{eff} es menor a un umbral o han transcurrido un número fijo de intervalos sin remuestreo.
- Los pesos de las partículas se convierten a probabilidades normalizadas.
- Las partículas con menor peso son sustituidas por partículas con mayor peso.
- Al sustituirse sólo se copian las trayectorias y los mapas.
- Los pesos se vuelven 0 para todas las partículas.

Construcción del mapa

- Debido a la marginalización analítica del mapa por el teorema de Rao-Blackwell
- Se utiliza la trayectoria de la partícula con mayor ponderación.
- El mapa se construye utilizando el enfoque histográmico de Borestein.

$$CV_{i,j}(t + 1) = CV_{i,j}(t) + \Delta(t)$$

- Es posible la fusión sensorial con otros sensores (sonar).

Fusión sensorial

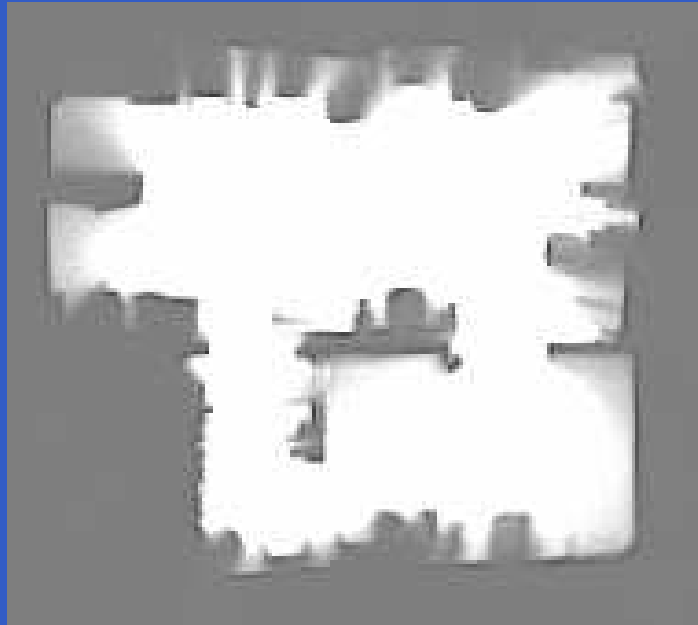
La ocupación (O) de cada celda esta dada por la disyunción entre la ocupación observada por los distintos sensores (S_i):

$$p(O) = p(S_1 \vee S_2 \cdots \vee S_n)$$

Suponiendo que las lecturas de los distintos sensores son mutuamente independientes y exclusivos obtenemos:

$$p(O) = 1 - \prod_{i=1}^n (1 - p(S_i))$$

LSI

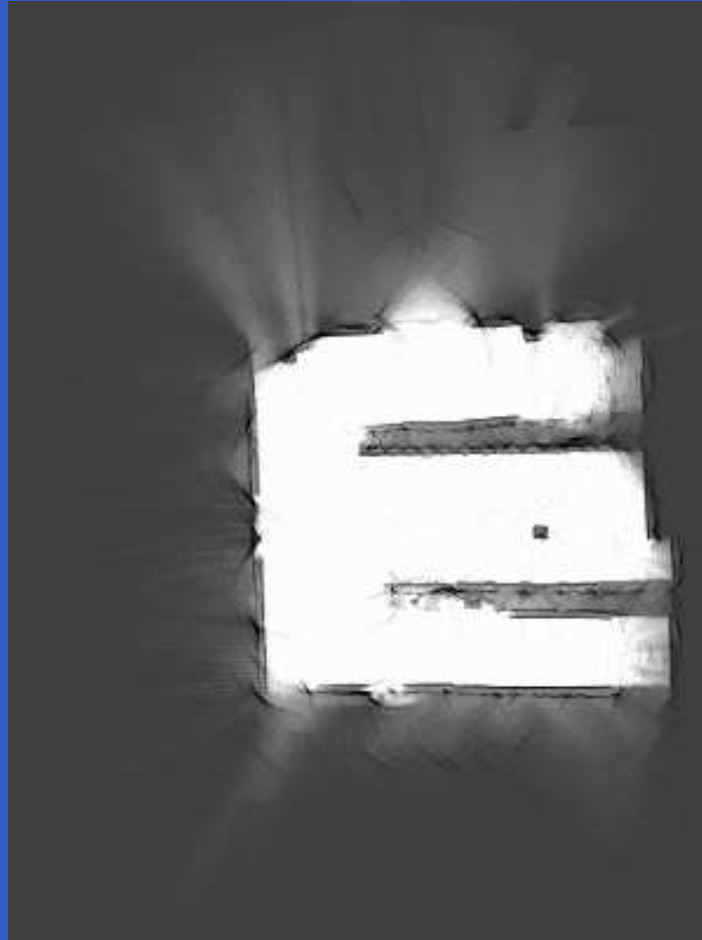


-
-
-

Departamento de matemáticas



CEC



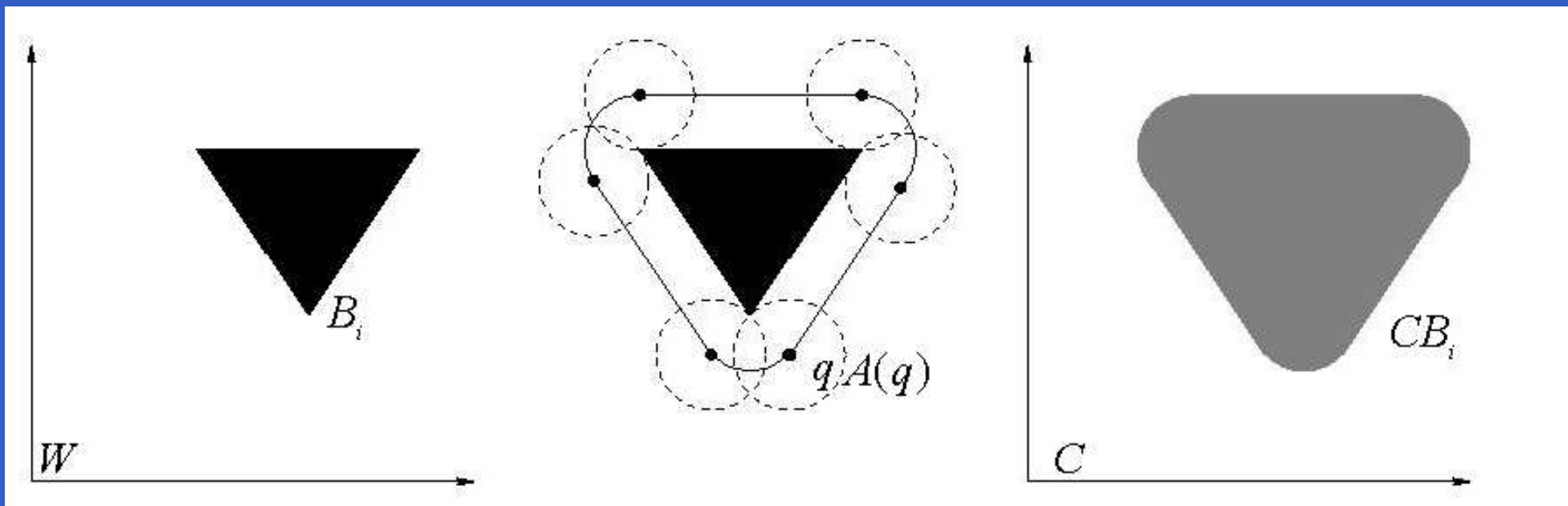
Exploración autónoma

- La resolución de SLAM con filtros de partículas sólo ataca cierta parte del problema general.
- La estrategia de exploración no es asunto del SLAM.
- Características:
 - Enfoque tradicional: ir a la frontera más cercana.
 - Enfoque SLAM con PF: visitar zonas ya conocidas y cerrar circuitos.
 - El mapa se construye lento debido a las restricciones de cómputo.

Espacio de configuraciones

- Una configuración q del robot A es una especificación del estado físico de A con respecto a un marco fijo del ambiente F_w .
- La configuración de un robot móvil A puede representarse completamente como $q = [xy\theta]$.
- El C-Space de A es el espacio \mathcal{C} de todas las posibles configuraciones válidas de A dentro de su ambiente.

Espacio de configuraciones



Algoritmo de exploración

loop

$P \leftarrow$ valor aleatorio uniforme $[0, 1)$

if $P < \epsilon$ **then**

$Angulo \leftarrow$ valor aleatorio entre $[0, \pi/2]$

$Distancia \leftarrow$ obtener distancia en el $Angulo$

else

$MaxDist \leftarrow 0$

for all Laser en el LaserCSpace **do**

$Dist \leftarrow \sqrt{(X \text{ anterior} - X \text{ actual})^2 + (Y \text{ anterior} - Y \text{ anterior})^2} + Laser.Distancia$

if $Dist > MaxDist$ **then**

$Angulo = Laser.Angulo$

$Distancia = Laser.Distancia$

end if

end for

end if

ejecutar giro al $Angulo$

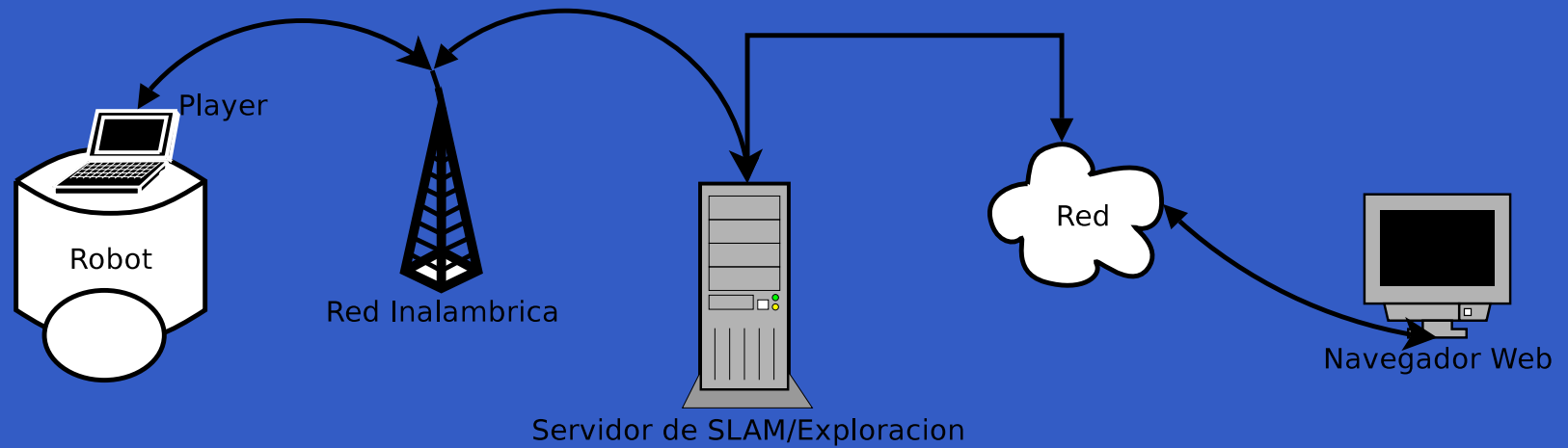
while obstáculos estén lejanos **do**

ejecutar avance con $Distancia$

end while

end loop

Vista general del sistema

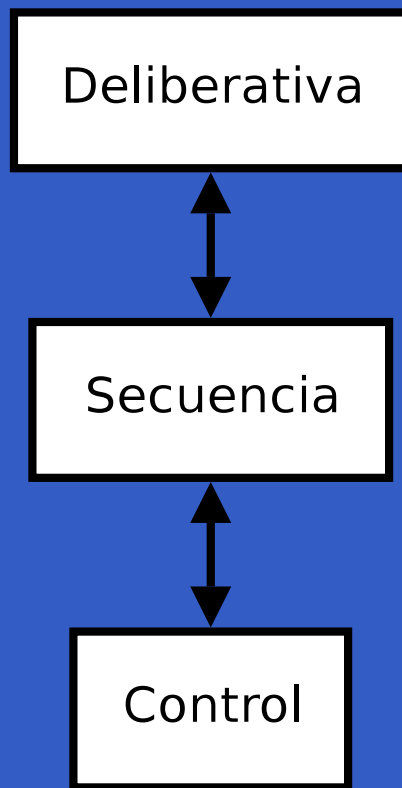


Arquitectura de software

No confundir gimnasia con magnesia.
Son dos vistas a un mismo problema.

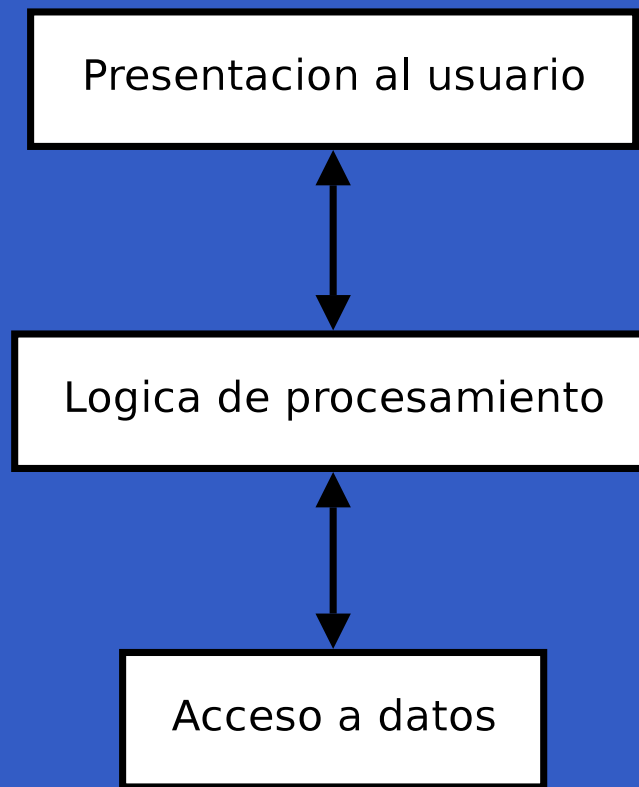
- Arquitectura de tres capas (layers)
Vista lógica y de control del sistema.
- Arquitectura de tres gradas (tiers)
Vista de despliegue y procesamiento.

Arquitectura de tres capas



!Asignatura pendiente!

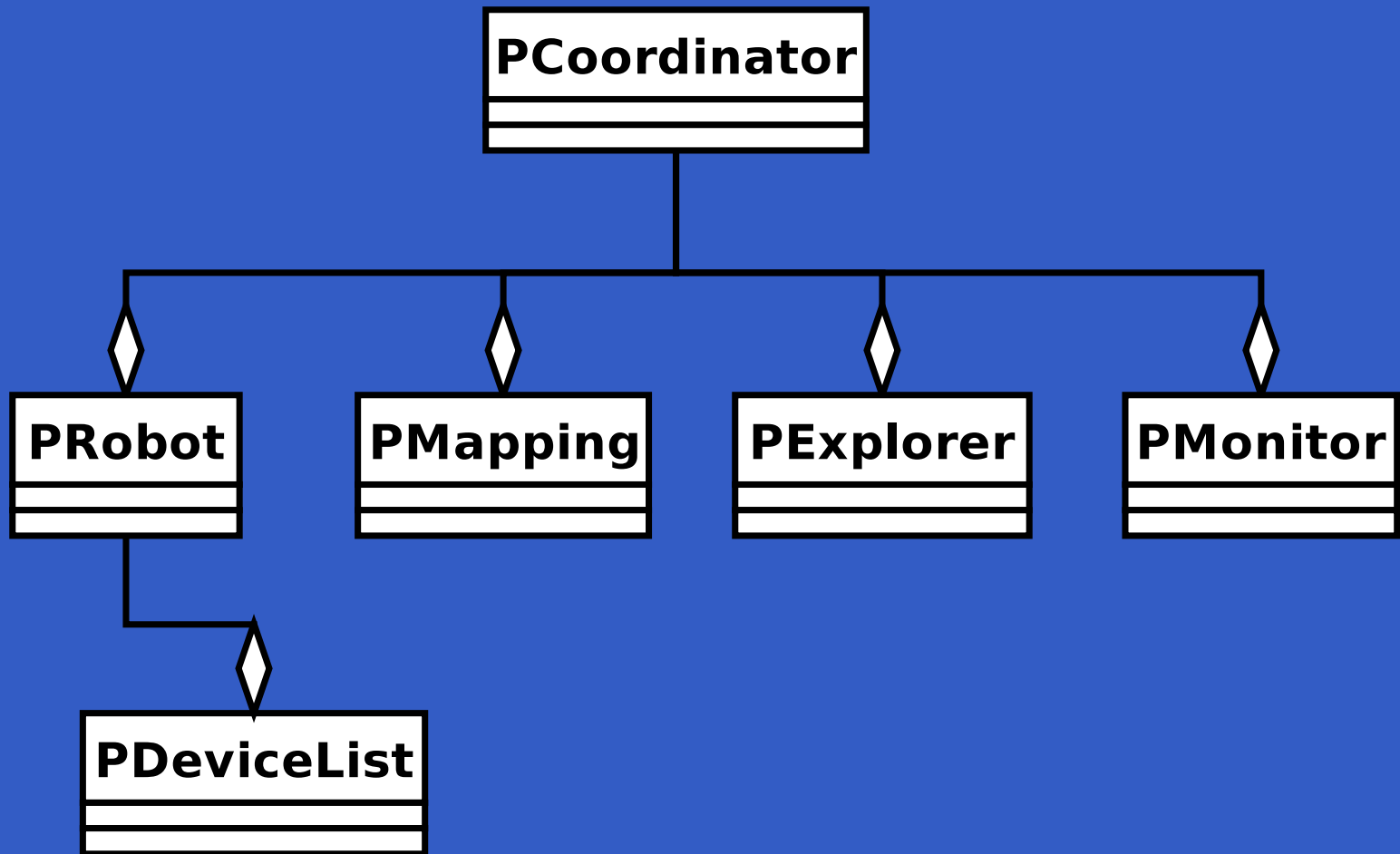
Arquitectura de tres gradas



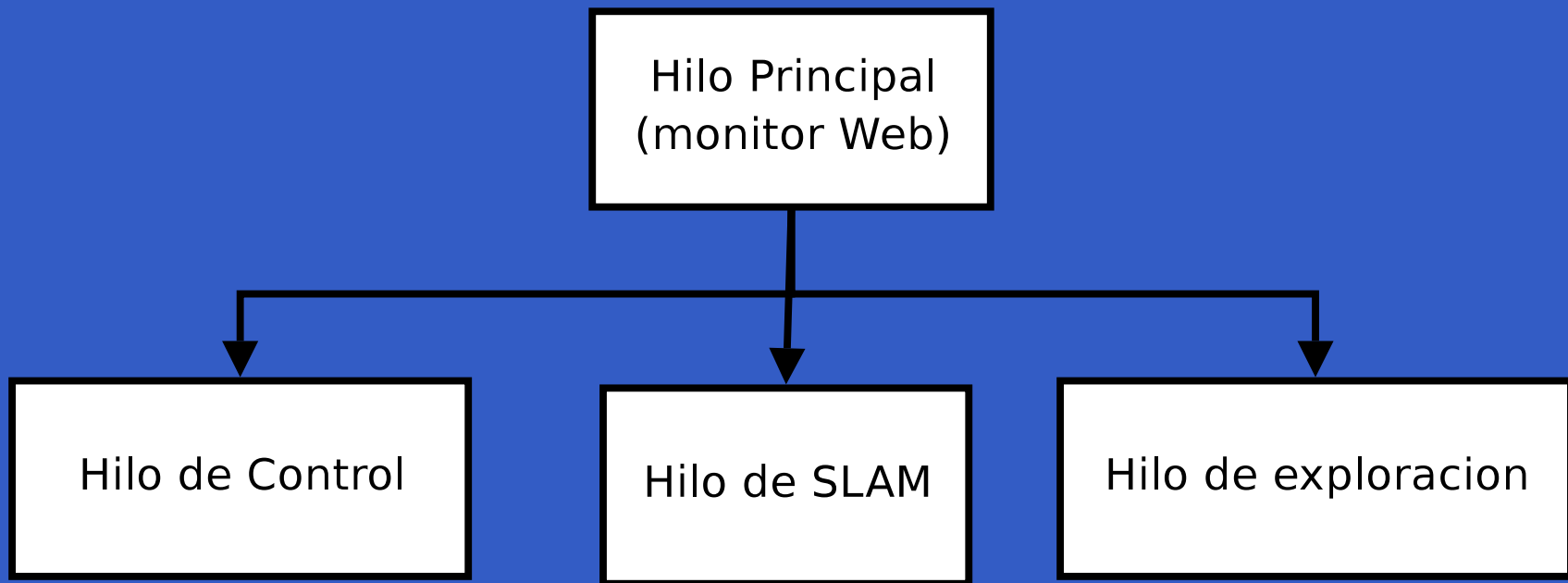
Dependencias de software

- GNU/Linux - Tux Power!
- Player/Stage - (¡La versión 2.0 está en el horno!).
- PMAP modificado - la magia del filtro de partículas.
- glib-object - objetos y retrollamadas en C.
- gthread - manejo de hilos.
- libxml - manejo del archivo de configuración.
- libsoup - servidor web incrustado.

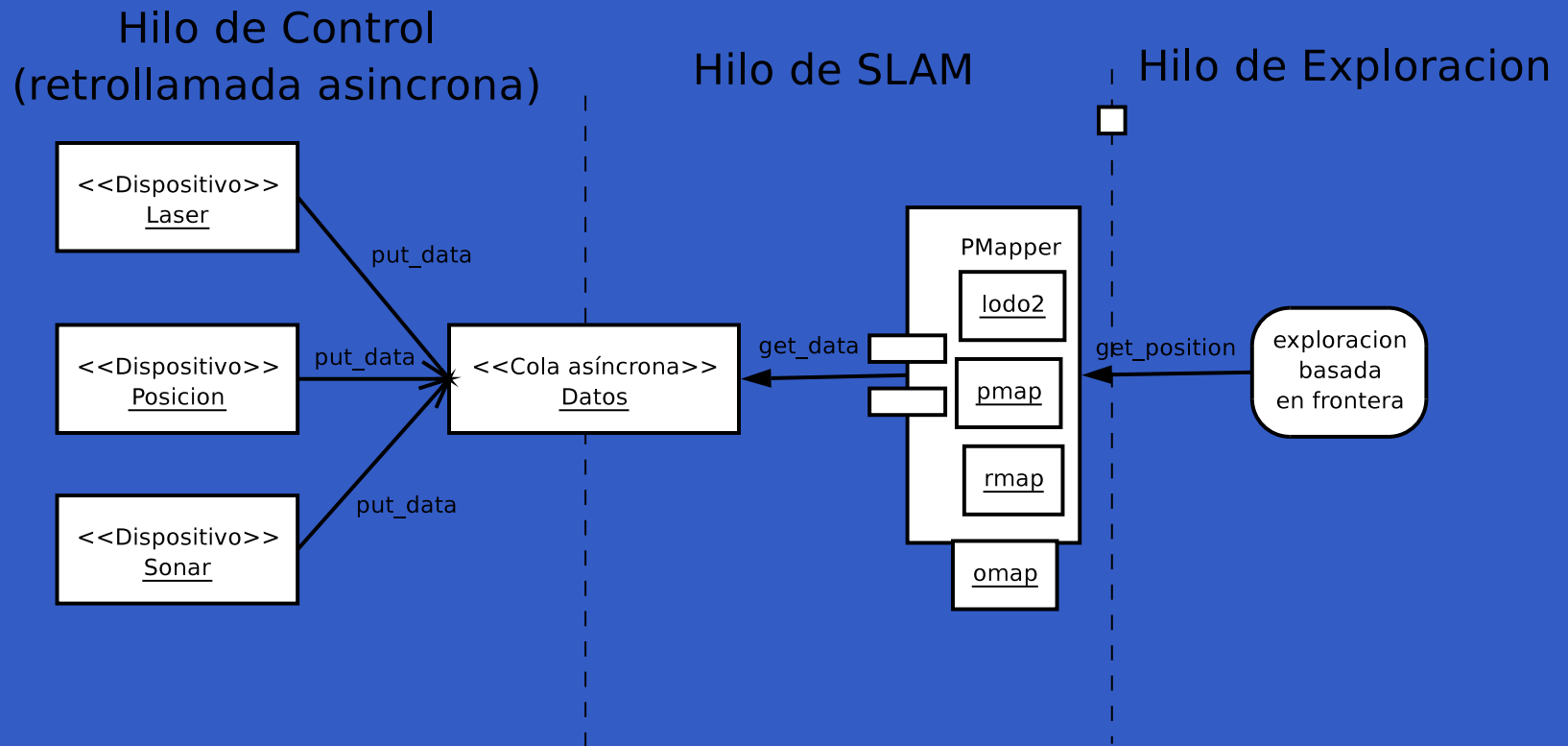
Vista de clases



Vista de hilos



Procesamiento asíncrono



Gracias y Bienvenidos

Markovio les da las gracias, y...
Welcome to the jungle!

