

**COMPARATIVA DE MÉTODOS DE  
MUESTREO EN UN FILTRO DE  
PARTÍCULAS APLICADO AL SEGUIMIENTO  
DE MULTIPLES OBJETOS.**

**MÉTODOS PROBABILÍSTICOS APLICADOS A LA  
ROBÓTICA**

<b>1. INTRODUCCIÓN.....</b>	<b>3</b>
<b>2. XPFCP (Extended Particle Filter with a Clustering Process). .....</b>	<b>3</b>
2.1 Filtro de partículas estándar: el algoritmo SIR (“Sequential Important Resampling”). .....	3
2.1.1. Inicialización. ....	3
2.1.2. Predicción. ....	4
2.1.3. Corrección. ....	5
2.1.4. Selección.....	5
2.2. Problemas del PF estándar. ....	5
2.3. XPFCP (Extended Particle Filter with a Clustering Process).....	6
2.3.1. Reinicialización. ....	6
2.3.2. Corrección. ....	6
2.3.3. Selección.....	8
2.3.4. Salida. ....	8
<b>3. MÉTODOS DE MUESTREO.....</b>	<b>8</b>
3.1. Algoritmo Multinomial.....	9
3.2. Algoritmo Residual.....	10
3.3. Algoritmo Systematic definido en [Carpenter, et al., 99a].....	10
3.4. Algoritmo Systematic definido en [Hol, et al., 06].....	11
3.5. Algoritmo Stratified.....	12
3.6. Algoritmo Minimum Variance. ....	12
3.7. Algoritmo Residual Systematic. ....	12
<b>4. RESULTADOS EXPERIMENTALES.....</b>	<b>13</b>
<b>5. CONCLUSIONES.....</b>	<b>18</b>
<b>6. BIBLIOGRAFÍA .....</b>	<b>19</b>

# 1. INTRODUCCIÓN

La razón del presente trabajo es exponer distintos algoritmos de muestreo para el filtro de partículas (PF) y comparar los resultados que se obtienen con ellos. Para las pruebas se emplea un XPFPC (*Extended Particle Filter with a Clustering Process*), ya que, como se explicará en el siguiente apartado, un PF estándar no puede ser utilizado para la estimación multimodal. Este fue aplicado al seguimiento de un número variable de objetos durante la fase de experimentación, de forma que se apreciase mejor el efecto de cada algoritmo de muestreo en el funcionamiento del PF.

## 2. XPFPC (Extended Particle Filter with a Clustering Process).

El filtro de partículas es un estimador recursivo bayesiano cuyo funcionamiento está basado en la representación discreta de la función de creencia por medio de un conjunto de partículas,  $S_t = \{\bar{s}_{i,t} = \bar{x}_t^{(i)} / i = 1 : n\}$ . Cada partícula tiene asociado un peso normalizado que representa su probabilidad de certidumbre dentro de la PDF global,  $\bar{w}_t = \{w_t^{(i)} / i = 1 : n\}$ . De este modo se conforma una creencia discreta que es la salida final del algoritmo de estimación:

$$p(\bar{x}_t, \bar{y}_t) \Rightarrow \{S_t, \bar{w}_t\} = \{\bar{x}_t^{(i)}, w_t^{(i)}\}_{i=1}^n \quad (2.1)$$

De esta función de creencia se extrae el valor final de estimación, bien mediante la envolvente de la PDF discreta, que da lugar a la función de creencia continua, o bien a través de distintos estadísticos de la creencia discreta como el valor más probable (la mediana) o el valor medio, junto con el valor de varianza que informa del valor de estimación final.

### 2.1 Filtro de partículas estándar: el algoritmo SIR (“Sequential Important Resampling”).

La versión estándar del filtro de partículas, el algoritmo SIR, consta de 3 etapas: “Predicción, Corrección y Selección”. En la Fig.1 se muestra la representación unidimensional de la evolución que sufren las partículas a lo largo de los distintos pasos de ejecución del algoritmo SIR.

El lazo de ejecución recursivo del PF estándar consiste en:

#### 2.1.1. Inicialización.

Existe una etapa de inicialización del algoritmo que únicamente se ejecuta en el instante  $t = 0$ . En ella se obtiene un conjunto de  $n$  partículas o muestras independientes e idénticamente distribuidas (i.i.d.),  $S_0 = \{\bar{s}_{i,0} = \bar{x}_0^{(i)} / i = 1 : n\}$ , a partir de la distribución inicial  $p(\bar{x}_0)$ , ponderadas todas ellas con el mismo peso  $\bar{w}_0 = \{w_0^{(i)} = \frac{1}{n} / i = 1 : n\}$ .

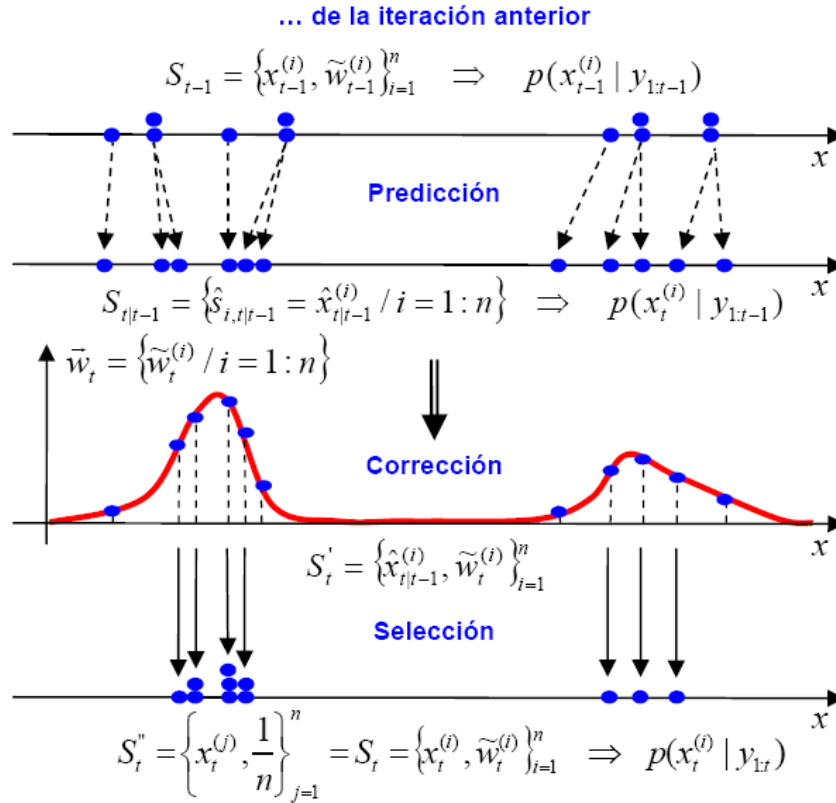


Fig.1. Representación unidimensional de la evolución del conjunto de partículas en las diferentes etapas del PF estándar [Marrón, s.p.].

En un instante  $t$  cualquiera el algoritmo SIR comienza con un conjunto de muestras ponderadas e i.i.d.  $S_{t-1} = \{\bar{x}_{t-1}^{(i)}, \tilde{w}_{t-1}^{(i)}\}_{i=1}^n$ , que representan el valor de la creencia estimada para el instante  $t-1$ ,  $p(\bar{x}_{t-1} | \bar{y}_{1:t-1})$ , por el PF.

### 2.1.2. Predicción.

Las partículas se propagan a través del modelo de estado del sistema bajo estudio según

$$\hat{\bar{x}}_{t|t-1}^{(i)} = \bar{x}_{t-1}^{(i)} \cdot p(\bar{x}_t, \bar{x}_{t-1}) = f(\bar{x}_{t-1}^{(i)}, \bar{u}_{t-1}, \bar{o}_{t-1}) \quad (2.2)$$

donde  $p(\bar{x}_t | \bar{x}_{t-1})$  y  $f(\bar{x}_t^{(i)}, \bar{u}_t, \bar{o}_t)$  caracterizan al modelo de estado del sistema de forma estocástica y determinística, respectivamente. De este modo se obtiene un nuevo set

$$S_{t|t-1} = \left\{\hat{s}_{i,t|t-1} = \hat{\bar{x}}_{t|t-1}^{(i)} / i = 1:n\right\} \quad (2.3)$$

que caracteriza a la probabilidad a priori  $p(\bar{x}_t | \bar{y}_{1:t-1})$ . El peso de las partículas no se modifica en este paso, quedando el set representado por la expresión

$$S_{t|t-1} = \left\{ \hat{\bar{x}}_{t|t-1}^{(i)}, \tilde{w}_{t-1}^{(i)} \right\}_{i=1}^n \quad (2.4)$$

### 2.1.3. Corrección.

Se actualiza el peso asociado a cada partícula a partir de la función de aproximación a la creencia  $q(\bar{x}_t | \bar{x}_{0:t-1} \bar{y}_{1:t})$  y las densidades  $p(\bar{x}_t | \bar{x}_{t-1})$  y  $p(\bar{y}_t | \bar{x}_t)$  que caracterizan, respectivamente, al comportamiento probabilístico del modelo de estados y de medidas del sistema bajo estudio

$$w(\bar{x}_{0:t}) = w(\bar{x}_{0:t-1}) \cdot \frac{p(\bar{y}_t | \bar{x}_t) \cdot p(\bar{x}_t | \bar{x}_{t-1})}{q(\bar{x}_t | \bar{x}_{0:t-1}, \bar{y}_{1:t})} \Rightarrow \tilde{w}_t = \left\{ w_t^{(i)} / i = 1:n \right\} \quad (2.5)$$

Los pesos así actualizados se normalizan después, para obtener el nuevo set

$$S'_t = \left\{ \hat{\bar{x}}_{t|t-1}^{(i)}, \tilde{w}_t^{(i)} \right\}_{i=1}^n \quad (2.6)$$

### 2.1.4. Selección.

A través del esquema de muestreo elegido y empleando el vector de pesos normalizado calculado en el paso anterior  $\tilde{w}_t = \left\{ \tilde{w}_t^{(i)} / i = 1:n \right\}$ , se obtiene el nuevo set de partículas

$S_t'' = \left\{ \bar{x}_t^{(j)}, \frac{1}{n} \right\}_{j=1}^n$ . Esta es, además, la salida discreta  $S_t = \left\{ \bar{x}_t^{(i)}, \tilde{w}_t^{(i)} \right\}_{i=1}^n$  que proporciona el PF

como estimación final de la densidad de creencia o probabilidad a posteriori  $p(\bar{x}_t | \bar{y}_{1:t})$  para el instante  $t$ . Este conjunto de muestras es usado como entrada para la iteración del algoritmo en el instante  $t+1$ .

## 2.2. Problemas del PF estándar.

El PF estándar presenta dos restricciones que lo hacen inadecuado para el seguimiento de objetos múltiples [Koller-Meier, *et al.*, 01]:

1. No es capaz de incorporar la estimación de nuevos modelos en el proceso, ya que debido a su baja probabilidad, las partículas relacionadas con estos serán rechazadas en la etapa de selección del PF.
2. El paso de corrección debe ser modificado ya que el peso de cada partícula ha de ser calculada basándose en las similitudes entre la partícula y el modelo, o el vector de salida del modelo, al que corresponde.

## 2.3. XPFCP (Extended Particle Filter with a Clustering Process).

Para conseguir el objetivo de estimación multimodal se recurre al XPF [Koller-Meier, *et al.*, 01], y se le añade un proceso de clasificación que aumenta la robustez del filtro de partículas extendido [Marrón, *et al.*, 05]. En la Fig.2 se muestra la representación unidimensional de la evolución de las partículas empleando el algoritmo XPFCP.

Las modificaciones de este con respecto al PF estándar se enumeran a continuación.

### 2.3.1. Reinicialización.

En este paso se insertan  $n_{m,t}$  nuevas partículas en la creencia obtenida en el instante temporal anterior  $p(\vec{x}_{t-1} | \vec{y}_{1:t-1})$ . Las partículas insertadas provienen del conjunto de medidas obtenidas en ese mismo instante  $p_m(\vec{y}_{t-1})$ . De ese modo se incorpora en la creencia a priori, que se obtiene tras el paso de predicción del PF, información de la función de verosimilitud de todas las variables a estimar.

Con el proceso de clasificación es posible seleccionar las medidas que son introducidas en la distribución a priori en este paso, en función de su relación con cada variable a estimar. Generalmente, se insertan  $n_{m,t} / k$  medidas por cada modelo a estimar, donde  $k$  es el número de estos.

La reinicialización del set de partículas hace que la función densidad de probabilidad se vaya amoldando dinámicamente al número de variables de la estimación. Es la única forma de obtener una creencia que represente a múltiples hipótesis sin que sean rechazadas en el paso de selección, puesto que en caso contrario sólo sería posible mantenerla en la PDF de la creencia si fuese semejante a una anteriormente ya representada por esta.

### 2.3.2. Corrección.

Para habilitar la posibilidad de la estimación multimodal, el peso de las partículas en la etapa de corrección ha de ser calculado acorde al modelo al que corresponde la partícula. Para conseguir este objetivo, la función base para obtener la verosimilitud es modificada según

$$p_i(\vec{y}_t | \vec{x}_t^{(i)}) = e^{-d_{i,t}^2 / 2\sigma^2} \Rightarrow d_{i,t} = \min_{j=1:m_t} \left\{ \sqrt{\left( h(\vec{x}_t^{(i)}) - \vec{g}_t^{(j)} \right)^2} \right\} \quad (2.7)$$

Los pesos se modifican según

$$w_t^{(i)} = w_{t-1}^{(i)} \cdot p_i(\vec{y}_t | \vec{x}_t^{(i)}) = w_{t-1}^{(i)} \cdot e^{-d_{i,t}^2 / 2\sigma^2} \quad (2.8)$$

Como se muestra en la ecuación (2.7), la nueva distribución  $p_i(\bar{y}_i|\bar{x}_i^{(i)})$  es obtenida comparando cada partícula  $\bar{x}_i^{(i)}$  con el centroide del modelo más cercano  $\bar{g}_i^{(j)}$ . Esto aumenta la robustez del estimador [Marrón, *et al.*, 05].

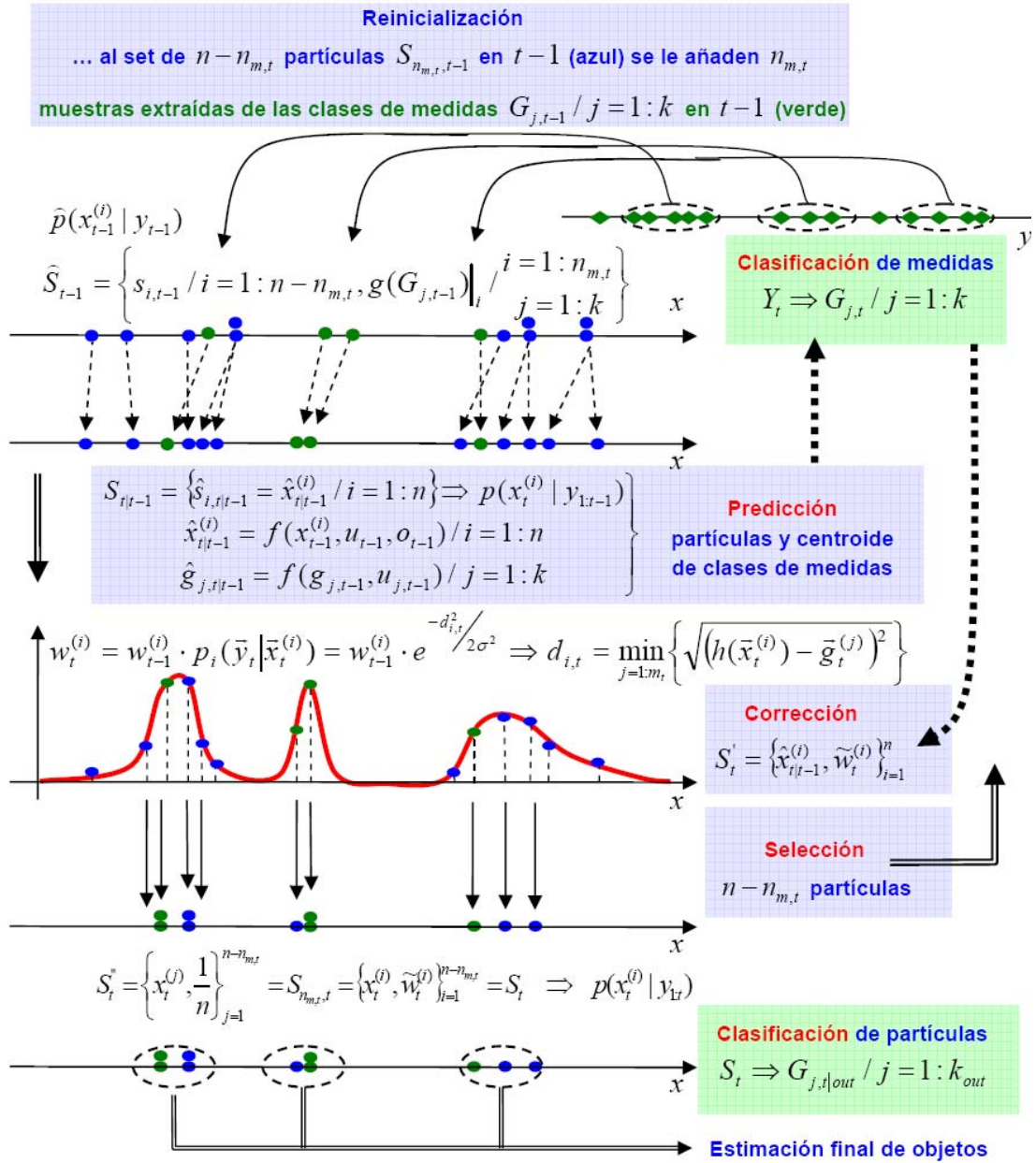


Fig.2. Representación unidimensional de la evolución del conjunto de partículas en las diferentes etapas del XPFCEP [Marrón, s.p.].

### 2.3.3. Selección.

A su funcionalidad básica se le añade la de reducir el número de partículas de salida del paso de manera que este sea constante tras el siguiente paso de reinicialización. Se eliminan  $n_{m,t}$  partículas.

### 2.3.4. Salida.

Se ejecuta un proceso de clasificación al final del algoritmo, en esta ocasión sobre el set  $S_t^n = \left\{ \bar{x}_t^{(j)}, \frac{1}{n} \right\}_{j=1}^{n-n_{m,t}}$ , empleando los centroides de las variables estimadas como el valor determinístico final para el vector de estados estimado.

## 3. MÉTODOS DE MUESTREO.

La etapa de selección, también llamada remuestreo o ‘resampling’, permite eliminar las partículas con un bajo peso asociado y reproducir a cambio aquellas que tienen un peso elevado, es decir, que constituyen hipótesis más probables de la estimación, de forma semejante al proceso de evolución en los algoritmos genéticos.

A partir del set  $S_t = \left\{ \bar{x}_t^{(i)}, \tilde{w}_t^{(i)} \right\}_{i=1}^n$  obtenido a la salida del paso de corrección del PF, se repite  $n_i$  veces cada partícula  $\bar{x}_t^{(i)}$ , de modo que el número de partículas de salida  $n_i$  que le corresponde a cada partícula  $\bar{x}_t^{(i)}$  está relacionado con su peso asociado  $\tilde{w}_t^{(i)}$ , según la expresión:

$$E\{n_i\} = n \cdot \tilde{w}_t^{(i)} \Rightarrow \sum_{i=1}^n n_i = n \quad (3.1)$$

donde  $E\{n_i\}$  es la esperanza de cada partícula. Se obtiene así un conjunto de partículas  $S_t^n = \left\{ \bar{s}_{j,t}^n = \bar{x}_t^{(j)} / j = 1 : n \right\}$ , cada una de ellas con un peso idéntico e igual a  $n^{-1}$ :

$$S_t^n = \left\{ \bar{x}_t^{(j)}, \frac{1}{n} \right\}_{j=1}^n \quad (3.2)$$

La condición establecida en la ecuación (3.1) para elegir el número de muestras o ‘hijos’  $n_i$  a repetir de cada partícula, no es obligatoria tal y como se demuestra en [Kitagawa, 96]. Debido a ello gran cantidad de proposiciones de algoritmos de muestreo han sido desarrolladas por distintos grupos de investigación hasta la actualidad. En algunos casos el objetivo de la



propuesta es minimizar la carga computacional de esta etapa en el PF, y en otros el de disminuir

la varianza de la distribución generada por el conjunto de salida  $S_t^n = \left\{ \bar{x}_t^{(j)}, \frac{1}{n} \right\}_{j=1}^n$ .

En todos los casos, el objetivo principal del método de selección en el algoritmo SIR es el de resolver el problema de degeneración presente en el algoritmo SIS (“Sequential Importance Sampling”) [Doucet, *et al.*, 01], que consiste en el hecho de que el peso total del set se concentre en una pocas hipótesis. En [van der Merwe, *et al.*, 01] se demuestra que el uso de uno u otro algoritmo de selección no reporta una diferencia importante en la eficiencia de la estimación final. El objetivo de este trabajo es encontrar el algoritmo más idóneo aplicado a un proceso de estimación concreto.

Los métodos de resampling que se han analizado en este trabajo se listan a continuación:

- Multinomial.
- Residual.
- Systematic definido en [Carpenter, *et al.*, 99a].
- Systematic definido en [Hol, *et al.*, 06].
- Stratified.
- Minimun Variance.
- Residual Systematic.

### 3.1. Algoritmo Multinomial.

Fue introducido en [Gordon, *et al.*, 93] y consiste en muestrear uniformemente  $n$  veces la creencia discreta caracterizada por el set de partículas  $S_t^i = \{\hat{s}_{i,t|t-1} = \hat{x}_{t|t-1}^{(i)} / i = 1 : n\}$ , cada una de ellas con una probabilidad  $\bar{w}_t = \{\tilde{w}_t^{(i)} / i = 1 : n\}$ .

El algoritmo consiste en generar  $N = n - n_{m,t}$  números aleatorios uniformemente ordenados

$$u_N = \tilde{u}_N^{\perp}, u_k = u_{k+1} \tilde{u}_k^{\perp}, \text{ con } \tilde{u}_k \sim U[0,1) \quad (3.3)$$

que se emplean para seleccionar un nuevo set de partículas  $S_t^j$  según la distribución multinomial siguiente

$$\bar{x}_t^{(k)} = x(F^{-1}(u_k)) = \bar{x}_t^{(j)}, \text{ con } j \text{ tal que se cumpla que } u_k \in \left[ \sum_{i=1}^{j-1} \tilde{w}_t^{(i)}, \sum_{i=1}^j \tilde{w}_t^{(i)} \right) \quad (3.4)$$

donde  $F$  denota la distribución de probabilidad acumulada de los pesos normalizados de las partículas [Hol, *et al.*, 06]. La Fig.3 muestra gráficamente el funcionamiento del método de resampling multinomial.

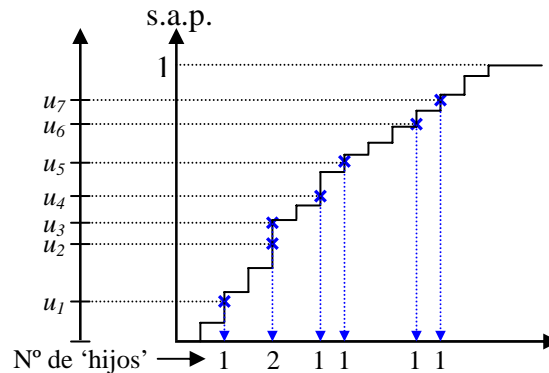


Fig.3. Ejemplo del funcionamiento del algoritmo de muestreo multinomial.

La complejidad del algoritmo es  $O(N)$  [van der Merwe, *et al.*, 01]. En los siguientes apartados se verá que la complejidad es la misma para todos los algoritmos de resampling, lo que no coincide es el tiempo de ejecución de cada algoritmo como podrá comprobarse en el apartado correspondiente a las pruebas.

### 3.2. Algoritmo Residual.

Este método alternativo de resampling fue introducido en [Liu, *et al.*, 98] y es computacionalmente menos costoso que el método de muestreo multinomial [Hol, *et al.*, 06].

Se sigue el esquema propuesto en [Hol, *et al.*, 06] y [van der Merwe, *et al.*, 01]. Este método consta de dos partes.

1. Se obtienen  $n_i = \lfloor N \cdot \tilde{w}_t^{(i)} \rfloor$  copias de la partícula  $\tilde{x}_t^{(i)}$  para el nuevo set, donde  $N = (n - n_{m,t})$  es el número total de partículas de salida deseadas.
2. Se obtienen el resto de partículas para el nuevo set remuestreando  $m = N - \sum_{i=1}^n n_i$  partículas empleando para ello el esquema de selección multinomial descrito en el apartado 3.1.

La complejidad del algoritmo es  $O(N)$  [Hol, *et al.*, 06].

### 3.3. Algoritmo Systematic definido en [Carpenter, et al., 99a].

Fue propuesto por Carpenter y Clifford [Carpenter, *et al.*, 99a] y buscaba como objetivo reducir la complejidad computacional del filtro de partículas.

El algoritmo definido en [Carpenter, *et al.*, 99a] y [Carpenter, *et al.*, 99b] es:

```

 $T = rand(0, (n - n_{m,t})^{-1}); j = 1; Q = 0; i = 0$ 
while  $T < 1$ 
if  $Q > T$  then
 $T = T + 1/(n - n_{m,t});$  salida  $\vec{x}_t^{(i)}$ 
else
seleccionar uniformemente  $k$  entre  $\{j, \dots, N\}$ 
 $i = \vec{x}_t^{(k)}$ 
 $Q = Q + \tilde{w}_t^{(k)}$ 
intercambiar  $(\vec{x}_t^{(k)}, \tilde{w}_t^{(k)})$  con  $(\vec{x}_t^{(j)}, \tilde{w}_t^{(j)})$ 
 $j = j + 1$ 
end if
end do

```

El algoritmo trata los pesos normalizados como un intervalo continuo (0,1) de sub-intervalos  $\left( \sum_{i=1}^{j-1} \tilde{w}_t^{(i)}, \sum_{i=1}^j \tilde{w}_t^{(i)} \right)$ . Este es ordenado aleatoriamente y el número de puntos  $T_k = \{T_{k-1} + k/(n - n_{m,t})\}$  perteneciente a cada sub-intervalo es contado, resultando el número  $n_i$  de ‘hijos’ por partícula.

La complejidad computacional del algoritmo es  $O(N)$  [Carpenter, *et al.*, 99a].

### 3.4. Algoritmo Systematic definido en [Hol, et al., 06].

Similar en su funcionamiento al algoritmo multinomial salvo en la generación de los números aleatorios empleados para la selección de los ‘hijos’ de las partículas. Fue propuesto en [Kitagawa, 96]. El funcionamiento del algoritmo se describe a continuación.

Se generan  $N = n - n_{m,t}$  números aleatorios:

$$u_k = \frac{(k-1) + \tilde{u}}{N}, \text{ con } \tilde{u} \sim U[0,1) \quad (3.5)$$

y para seleccionar el set de partículas  $S_t''$  se emplea el mismo método que para el algoritmo de muestreo multinomial.

La complejidad computacional del algoritmo es  $O(N)$  [Hol, *et al.*, 06].

### 3.5. Algoritmo Stratified.

Como en el método descrito anteriormente, la única diferencia de este con respecto al multinomial es la forma de generar los números aleatorios que se emplean para determinar el número de hijos por partícula. Fue propuesto en [Kitagawa, 96]. Se emplea el esquema definido en [Hol, *et al.*, 06].

Se generan  $N = n - n_{m,t}$  números aleatorios:

$$u_k = \frac{(k-1) + \tilde{u}_k}{N}, \text{ con } \tilde{u}_k \sim U[0,1] \quad (3.6)$$

el set de partículas  $S_t''$  se selecciona de la misma forma que en el algoritmo multinomial.

La complejidad computacional de este método de muestreo es  $O(N)$  [Hol, *et al.*, 06].

### 3.6. Algoritmo Minimun Variance.

Esta estrategia aúna los procedimientos de muestreo Stratified y Systematic, apartados 3.4 y 3.5, introducidos en [Kitagawa, 96] y el algoritmo ‘Tree Based Branching’ presentado en [Crisan, 01].

El algoritmo coincide con el descrito en [van der Merwe, *et al.*, 01].

Se generan  $N = n - n_{m,t}$  puntos  $u_k$  en el intervalo  $[0,1]$ , tal que:

$$u_k = \frac{k}{N}, k=1:N \quad (3.7)$$

el número de ‘hijos’  $n_i$  de cada partícula que conforman el set de partículas  $S_t''$  se obtiene del

número de puntos que pertenecen al intervalo  $\left[ \sum_{j=1}^{i-1} \tilde{w}_t^{(j)}, \sum_{j=1}^i \tilde{w}_t^{(j)} \right)$ .

La complejidad computacional del algoritmo es  $O(N)$  [van der Merwe, *et al.*, 01].

### 3.7. Algoritmo Residual Systematic.

Se sigue el esquema descrito en [Bolic, *et al.*, 03] y [Athalye, *et al.*, 05]. El número de partículas a la salida del algoritmo es  $N = n - n_{m,t}$  partículas. El algoritmo consiste en:

```

 $u_k \sim U[0, \frac{1}{N}],$ 
for  $i = 1$  to  $n$ 
 $n_i = \lfloor (\tilde{w}_t^{(i)} - u_{(k-1)}) \cdot N \rfloor + 1$ 
 $u_k = u_{k-1} + \frac{n_i}{N} - \tilde{w}_t^{(i)}$ 
end
    
```

El algoritmo genera  $n_i$  hijos de cada partícula  $\bar{x}_t^{(i)}$ .  $u_0$  es un número aleatorio entre 0 y  $\frac{1}{N}$  y actualizado según  $u_k = u_{k+1} + \frac{1}{N}$ . El número de repeticiones para la partícula  $\bar{x}_t^{(i)}$  se determina como el número  $u_k$  que pertenecen al intervalo  $\left( \sum_{i=1}^{j-1} \tilde{w}_t^{(i)}, \sum_{i=1}^j \tilde{w}_t^{(i)} \right)$ .

La complejidad computacional de este algoritmo es  $O(N)$  [Bolic, 04].

## 4. RESULTADOS EXPERIMENTALES.

Para la comparación de los métodos de muestreo se programaron los algoritmos en el entorno MATLAB y se realizaron pruebas con datos reales extraídos de estereo-visión. Como se indicó en la introducción a este trabajo, el XPFCP se empleó como sistema de seguimiento de objetos múltiples y variables.

La Fig.4 muestra la salida del XPFCP en 4 frames de una secuencia global. Cada frame está organizado verticalmente en:

- La imagen superior muestra la imagen procedente de la cámara izquierda. Los puntos verdes representan las medidas, los puntos rojos la posición de las partículas y los rectángulos rojos representan objetos o modelos validados de salida.
- La imagen inferior representa la proyección en ejes XY de los resultados 3D de la imagen superior. El significado de los diferentes elementos que aparecen es el comentado anteriormente. En este caso un círculo es la forma para el modelo cilíndrico que representa cada objeto validado de salida.

Las siguientes ecuaciones muestran el modelo empleado para estimar la posición y dinámica de los objetos a seguir

$$\bar{\vec{x}}_t = \begin{bmatrix} x_t \\ z_t \\ y_t \\ \dot{x}_t \\ \dot{z}_t \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & T & 0 \\ 0 & 1 & 0 & 0 & T \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} x_{t+1} \\ z_{t+1} \\ y_{t+1} \\ \dot{x}_{t+1} \\ \dot{z}_{t+1} \end{bmatrix} + \vec{r}_t \quad (4.1)$$

$$\vec{y}_t = \begin{bmatrix} x_t \\ z_t \\ y_t \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \end{bmatrix} \cdot \begin{bmatrix} x_t \\ z_t \\ y_t \\ \dot{x}_t \\ \dot{z}_t \end{bmatrix} + \vec{o}_t \quad (4.2)$$

Las ecuaciones (4.1) y (4.2) representan respectivamente la ecuación de estado y la ecuación de salida del modelo, donde  $[x \ y \ z]'$  es el vector de posición en coordenadas cartesianas;  $\vec{r}_t$  y  $\vec{o}_t$  son los vectores de ruido respectivamente del vector de estado y vector de salida; y  $T$  es el periodo de muestreo del proceso de estimación.

Se han realizado dos pruebas con videos diferentes, denominadas (a) y (b). Los videos son capturas del sistema de estereo-visión en los cuales aparecen diversas personas en movimiento junto con objetos estáticos, lo que supone que el XPFCP realiza el seguimiento de múltiples objetos con dinámica muy diferente. Los vídeos empleados en ambas pruebas son adjuntados.

El número total de partículas empleado en los experimentos es  $N=600$ . El número de partículas que se introducen en el paso de reinicialización es  $M=200$ .

En la primera prueba (a) el objetivo era encontrar un criterio de selección entre un algoritmo u otro basándose tanto en los tiempos de ejecución como en el número de partículas asignadas a los objetos que aparecían en el campo de visión de las cámaras. Esta última observación es una buena medida de cómo el algoritmo de selección potencia o no la aparición de nuevos objetos en la estimación del PF.

El tiempo de ejecución del PF para los diversos algoritmos de muestreo fue en torno a 0.06 s de media. La Fig.5 muestra el número de partículas asignadas a cada objeto para los diferentes métodos de muestreo propuestos. Con todos los métodos se sigue correctamente la posición de los obstáculos. Ambos resultados no permitieron establecer un criterio objetivo que permitiese establecer la idoneidad de un método u otro.

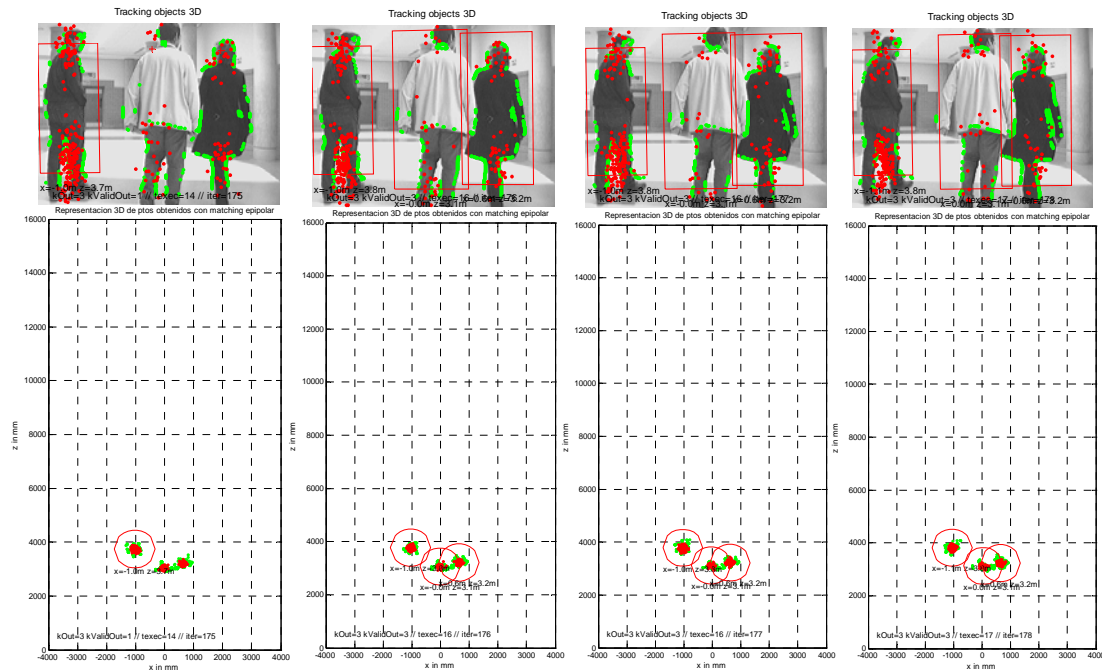


Fig.4. Imágenes secuenciales de un experimento con datos de estereovisión reales.

Para la prueba (b) se optó por emplear los métodos que se consideraron más representativos, descartando los algoritmos de muestreo Stratified, Minimum Variance y Systematic definido en [Hol, *et al.*, 06] debido a sus similitudes con el algoritmo de resampling multinomial. Los algoritmos para los que se realizó la prueba (b) fueron:

- Multinomial
- Residual
- Systematic [Carpenter, *et al.*, 99a]
- Residual Systematic

El primer paso fue fijar un criterio de lo que se consideraba un error. Los errores más representativos por su ratio de aparición se citan a continuación:

- Objeto/s perdido/s. El sistema obtiene medidas para los objetos pero no se les asignan partículas.
- 2/3 Objetos a 1. El sistema considera dos o más objetos como uno solo.
- Duplicados. Un objeto es considerado por el PF como dos o más.
- Desplazado. Las partículas asignadas a un objeto aparecen desplazadas respecto a este.

La prueba se realizó para dos vídeos distintos. Esta consistía en comprobar frame a frame la salida tanto del método de muestreo como del XPFCP para obtener el número de estos en los que no existe error y así disponer de un criterio objetivo para diferenciar los algoritmos de resampling.

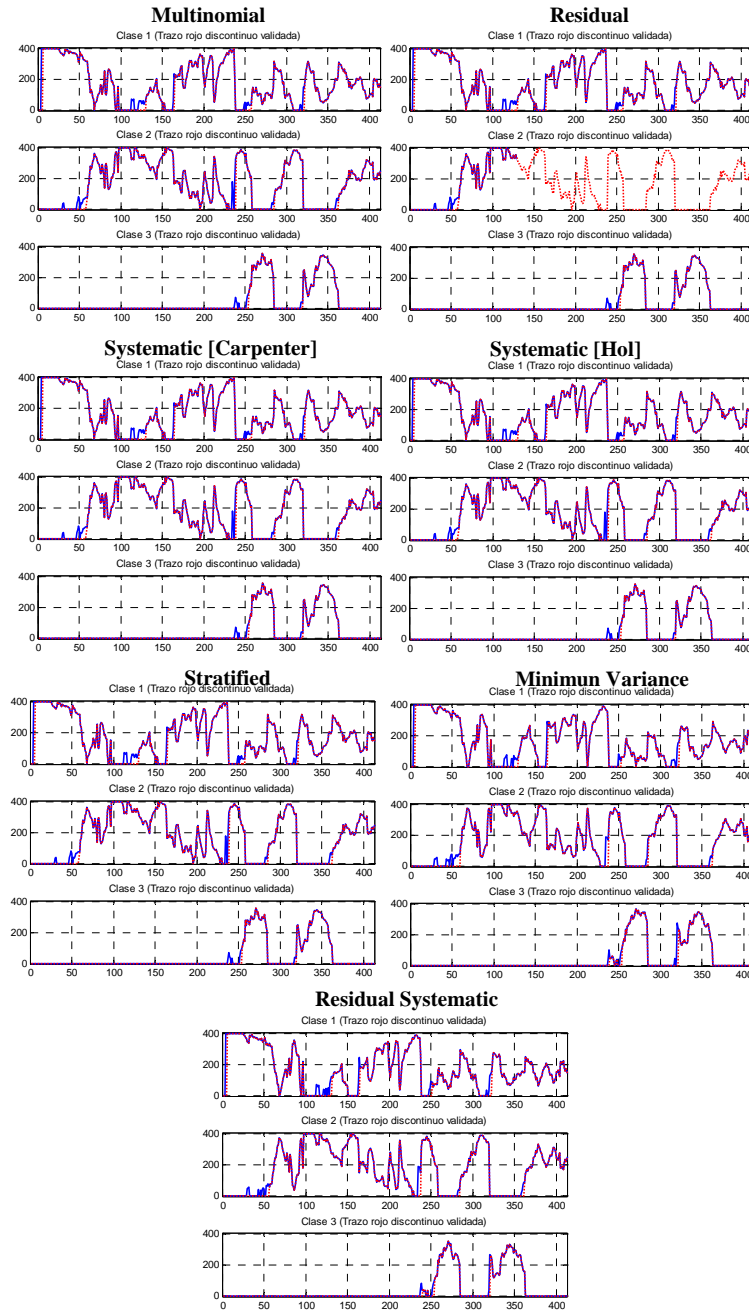
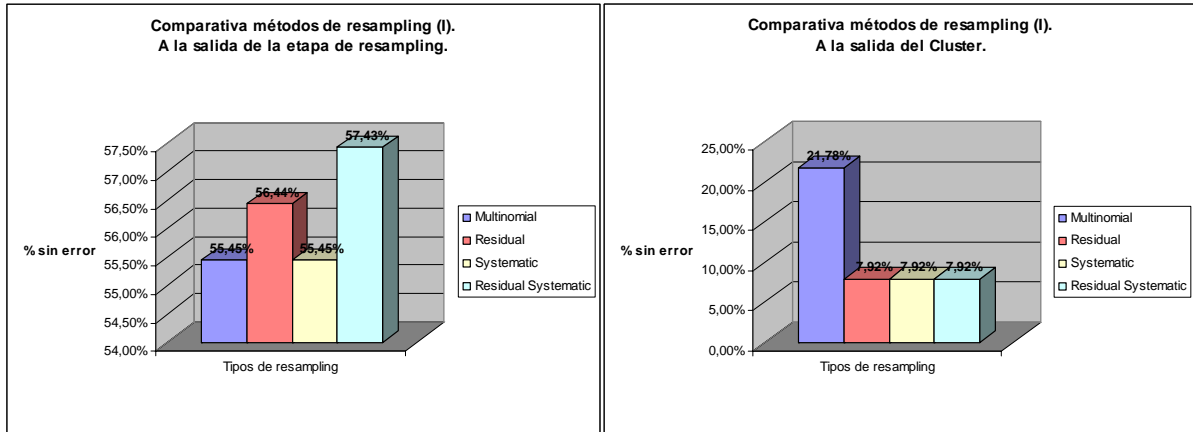


Fig.5. N° de partículas asignadas a cada modelo para diferentes métodos de muestreo.

En la Fig.6 se muestran los resultados obtenidos en el primero de los experimentos. En la Fig.6.a se muestra el porcentaje de frames sin error a la salida del método de muestreo. La diferencia en este caso entre emplear un método u otro es insignificante. En la Fig.6.b se representa el porcentaje de frames sin error a la salida del cluster, es decir, a la salida del XPFCP. La diferencia de resultados a la salida del algoritmo de muestreo y la salida del PF se debe a que en el fondo el cluster lo que hace es validar sólo aquellos modelos con suficientes partículas. En caso de que los objetos a estimar no tuvieran asignadas partículas sería un error del algoritmo de muestreo. En este caso parece justificado emplear el método Multinomial a la vista de los resultados.



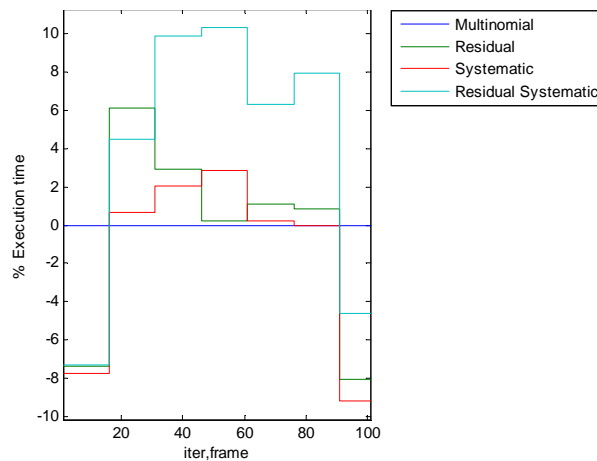
De la observación de la Fig.6.c se desprende que el método de selección Residual Systematic es el que más coste tiene desde un punto de vista de tiempos de ejecución, si bien la complejidad computacional de todos los métodos es la misma.



(a)

(b)

Comparación de los algoritmos de muestreo (respecto al algoritmo multinomial)



(c)

**Fig.6. a) Porcentaje de frames sin error a la salida de la etapa de selección. b) Porcentaje de frames sin error a la salida del XPFCP c) Comparación del tiempo de ejecución (%) respecto al método multinomial.**

En la Fig.7 se muestran los resultados obtenidos con el segundo vídeo o experimento. En Fig.6.a se muestra el porcentaje de frames sin error a la salida del método de muestreo. La diferencia entre emplear un método u otro sigue siendo insignificante. En Fig.7.b se representa el porcentaje de frames sin error a la salida del proceso de clasificación de la salida. El método Multinomial sigue siendo ligeramente mejor a la vista de los resultados. Sin embargo, como la diferencia es pequeña, si se tomasen como referencia los resultados para el tiempo de ejecución mostrados en Fig.7.c habría que considerar mejor el uso del método Systematic.

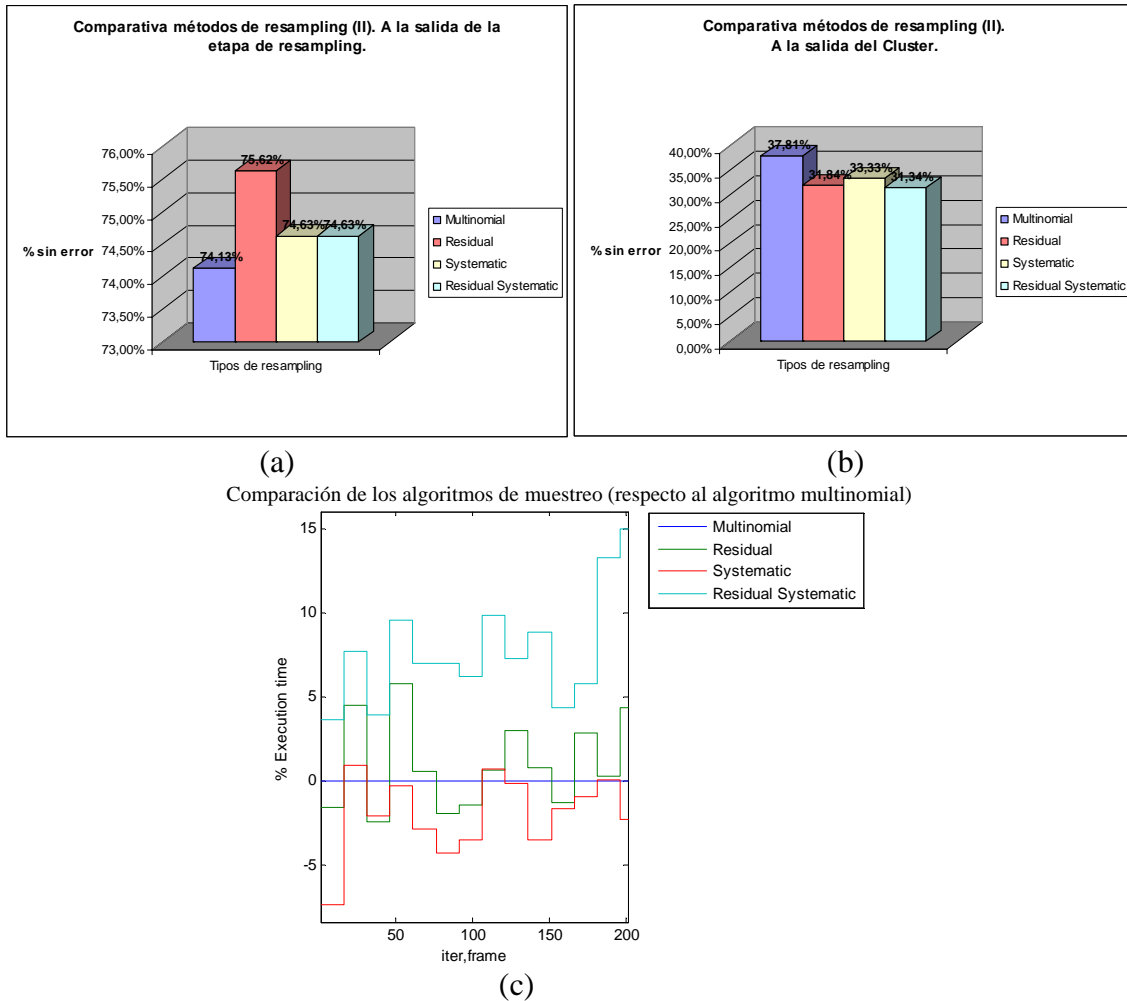


Fig.7. a) Porcentaje de frames sin error a la salida de la etapa de selección. b) Porcentaje de frames sin error a la salida del XPFPC c) Comparación del tiempo de ejecución (%) respecto al método multinomial.

## 5. CONCLUSIONES.

Desde un punto de vista de la salida de la etapa de selección, la elección de un método de muestreo u otro no es significativa como se ha podido comprobar en el apartado anterior. Ya en [van der Merwe, *et al.*, 01] y [Douc, *et al.*, 05] se apuntaba que la elección del método de muestreo no afectaba significativamente al funcionamiento del PF.

Desde el punto de vista de los resultados a la salida del XPFPC para la aplicación concreta de multitacking, el método de selección Multinomial es el que mejores resultados arroja.

En cuanto a tiempo de ejecución los resultados obtenidos favorecen el empleo del método de selección Systematic definido en [Carpenter, *et al.*, 99a].

La elección de uno u otro algoritmo de muestreo dependería de las prestaciones de la plataforma donde fuese a ser implementado en tiempo real. Si el tiempo de ejecución fuese excesivamente crítico la mejor elección es el método Systematic. En caso contrario, a tenor de

los resultados, el mejor método a implementar para esta aplicación en concreto es el algoritmo Multinomial.

## 6. BIBLIOGRAFÍA

[Athalye, *et al.*, 05] A. Athalye, M. Bolic, S. Hong, P.M. Djuric. "Generic Hardware Architectures for Sampling and Resampling in Particle Filters". *EURASIP Journal on Applied Signal Processing* 2005: 17, 2888-2902. 2005 Hindawi Publishing Corporation.

[Bolic, *et al.*, 03] M. Bolic, P.M. Djuric, S. Hong. "New Resampling Algorithms For Particle Filters". *Acoustics, Speech, and Signal Processing, 2003. Proceedings. (ICASSP '03). 2003 IEEE International Conference on Vol.2, 6-10 April 2003 Page(s):II - 589-92 vol.2*

[Bolic, 04] M.Bolic. "Architecture for Efficient Implementation of Particle Filters". A Dissertation Presented to The Graduate School in Partial Fulfillment of the Requirements for the Degree of Doctor of Philosophy in Electrical Engineering. Stony Brook University, August 2004.

[Carpenter, *et al.*, 99a] J. Carpenter, P. Clifford, P. Fearnhead. "Improved particle filter for nonlinear problems". *IEE Proc.-Radar, Sonar Navig.*, Vol. 146, No. 1, February 1999.

[Carpenter, *et al.*, 99b] J.Carpenter, P. Clifford, P. Fearnhead. "Building Robust Simulation-based Filters for Envolving Data Sets". Unpublished. Department of Statistics, University of Oxford.

[Crisan, 01] D. Crisan. "Particle filters – a theoretical perspective", A.Doucet, J.F.G. de Freitas and N.J. Gordon, Eds. *Sequential Monte Carlo Methods in Practise*, New York: Springer-Verlag.

[Douc, *et al.*, 05] R. Douc, O. Cappé, E. Moulines. "Comparison of Resampling Schemes for Particle Filtering". *Image and Signal Processing and Analysis, 2005. ISPA 2005. Proceedings of the 4th International Symposium on (2005) 64-69. September 2005.*

[Doucet, *et al.*, 01] A. Doucet, N. de Freitas, N. Gordon. "Sequential MonteCarlo methods in practice", Springer-Verlag, ISBN: 0-387-95146-6, New York, 2001.

[Gordon, *et al.*, 93] N.J. Gordon, D.J. Salmond, A.F.M. Smith. "Novel approach to nonlinear/non-gaussian bayesian state estimation", *IEE Proceedings-F in Radar and Signal Processing*, Vol. 140, n° 2, pp. 107-113, April 1993.

[Hol, *et al.*, 06] J.D. Hol, T.B. Schön, F. Gustafsson. "On Resampling Algorithms for Particle Filters". *Nonlinear Statistical Signal Processing Workshop, Cambridge, United Kingdom, September 2006.*

[Kitagawa, 96] G. Kitagawa. "MonteCarlo filter and smother for non-Gaussian nonlinear state space models", *Journal of Computational and Graphical Statistics*, Vol. 5, pp. 1-5, 1996.

[Koller-Meier, *et al.*, 01] E.B. Koller-Meier, F. Ade. "Tracking multiple objects using a condensation algorithm", *Journal of Robotics and Autonomous Systems*, Vol. 34, pp. 93-105, February 2001.

[Liu, *et al.*, 98] J.S. Liu, R. Chen. "Sequential MonteCarlo methods for dynamic systems", Journal of the American Statistical Association, Vol. 93, pp. 1032-1044, 1998.

[Marrón, *et al.*, 05] M. Marrón, M.A. Sotelo, J.C. García, D. Fernandez, D. Pizarro. "XPFCP: An extended particle filter for tracking multiple and dynamic objects in complex environments", Proceedings of the 2005 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS05), ISBN: 0-7803-9252-3, pp. 234-239, Edmonton, August 2005.

[Marrón, s.p.] M. Marrón. "Seguimiento de múltiples obstáculos en entornos interiores muy poblados basado en la combinación de métodos probabilísticos y determinísticos", Tesis Doctoral, Universidad de Alcalá, Sin Publicar.

[van der Merwe, *et al.*, 01] R. Van der Merwe, N. de Freitas, A. Doucet, E. Wan. "The unscented particle filter", Advances in Neural Information Processing Systems, Vol. 13, pp. 584-590, November 2001.