

# Hibridación entre Filtros de Partículas y Metaheurísticas para resolver problemas dinámicos

Juan J. Pantrigo, Ángel Sánchez, Antonio S. Montemayor, and Abraham Duarte

Universidad Rey Juan Carlos, Madrid, Spain

{juanjose.pantrigo, angel.sanchez, antonio.sanz, abraham.duarte}@urjc.es

**Resumen** Los problemas de optimización dinámica constituyen una generalización de los problemas de optimización en los cuales la descripción del problema y/o los datos relevantes cambian en el tiempo. Para estos problemas, es conveniente que los algoritmos de optimización utilizados dispongan de estrategias de adaptación a cambios. Los métodos metaheurísticos son procedimientos para explorar de forma efectiva y eficiente el espacio de búsqueda. Por otra parte, los filtros de partículas permiten describir la evolución temporal de un sistema. En este trabajo, se propone una nueva metodología para el desarrollo de algoritmos de optimización dinámica, denominada Filtro de Partículas Metaheurístico (MPF). MPF permite obtener algoritmos para la resolución de problemas de optimización dinámica a través de la hibridación de los métodos anteriores.

## 1. Introducción

Muchos problemas de optimización con aplicación práctica están definidos en entornos no estacionarios, es decir, el valor de las restricciones, variables relevantes del sistema, datos y/o estructuras del problema cambian con el tiempo [1]. Éstos reciben el nombre de problemas de optimización dinámica. Mientras que los métodos de optimización en problemas estáticos deben atender únicamente a encontrar una o un conjunto de soluciones de alta calidad, en los problemas dinámicos, además se debe seguir la evolución de éstas a lo largo del tiempo. En estas circunstancias, se necesitan métodos que permitan adaptar las soluciones continuamente a entornos en constante cambio [1][2].

La resolución de problemas dinámicos con metaheurísticas se aborda mediante dos estrategias posibles para obtener la solución a la instancia actual después de cambios [3]: restablecer el procedimiento de optimización (i) desde el inicio o (ii) desde las mejores soluciones encontradas a las instancias anteriores del problema. El primer enfoque, aborda el problema derivado suponiendo que no está relacionado con el anterior, lo que conlleva un desprecio de información que podría ser útil para reducir el tiempo de ejecución del algoritmo. La segunda estrategia es aplicable si el cambio en el entorno es poco importante, ya que implica una pérdida de diversidad drástica en el conjunto inicial de

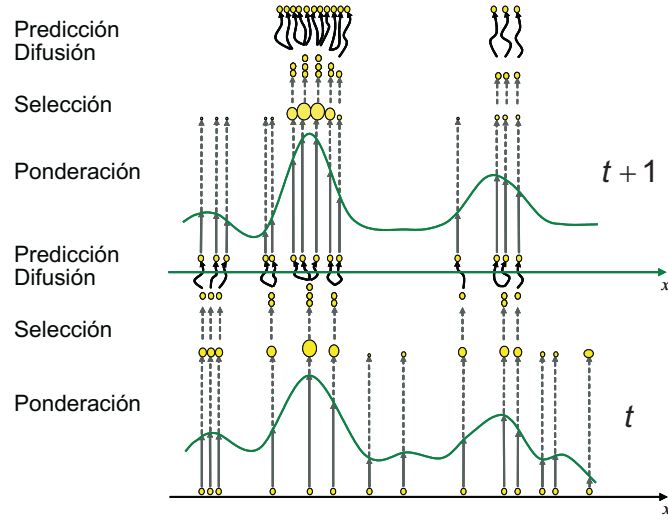
soluciones. Como ninguna de estas dos estrategias resuelve problemas dinámicos razonablemente, se han propuesto diversas modificaciones que las adaptan a condiciones no estacionarias. Entre ellas, destacan las que se basan en métodos evolutivos [3][4] y métodos constructivos [1]. Los métodos evolutivos son muy flexibles y permiten ajustar la diversidad de la población para adaptarla a los cambios en el entorno. El diseño de algoritmos de optimización dinámica asume frecuentemente que la diversidad es una propiedad deseable para conseguir adaptabilidad [3], ya que su pérdida reduce la capacidad de adaptación del algoritmo a los cambios del sistema. Sin embargo, Andrews [5] advierte que ésta puede ser una versión simplista del problema. Los métodos constructivos permiten ajustar una solución a los cambios del entorno mediante un proceso de *deconstrucción y reconstrucción* [6][1].

Los filtros de partículas resuelven problemas dinámicos introduciendo cierto conocimiento sobre la dinámica del sistema. Sin embargo, estos métodos no son adecuados para el tratamiento de problemas demasiado complejos. Por un lado, el cálculo de los pesos de las partículas, que implica la evaluación de la función de ponderación, es una tarea muy costosa para una gran parte de problemas con interés práctico y debe ser realizada una vez por cada partícula y en cada instante temporal. Además, se necesita un número impracticable de partículas para muestrear espacios de estados con un gran número de dimensiones [7][8]. Para abordar estos problemas se han propuesto diferentes procedimientos, que tienen el objetivo común de minimizar el número de partículas utilizadas sin perder calidad en las estimaciones.

La hipótesis central de este trabajo establece que, para resolver problemas de optimización dinámica, es conveniente el concurso de procedimientos de predicción, adaptación y optimización. Puesto que existen, por un lado, algoritmos de optimización (como las metaheurísticas) y, por otro, algoritmos de estimación secuencial (en particular, filtros de partículas), parece razonable abordar los problemas de optimización dinámica utilizando características de ambos métodos de manera sinérgica. En este trabajo se presenta el Filtro de Partículas Metaheurístico (*Metaheuristic Particle Filter* - MPF), una metodología de diseño de algoritmos híbridos orientados a la resolución más eficiente de problemas de optimización dinámica. Este método establece el marco conceptual de hibridación que permite obtener por instanciación diferentes algoritmos híbridos generales, que pueden ser posteriormente adaptados para resolver problemas concretos.

## 2. El problema de la estimación secuencial y los filtros de partículas

El problema de la estimación secuencial, desde un punto de vista bayesiano, consiste en el cálculo recursivo, con un cierto grado de confianza, del estado del sistema  $\mathbf{x}_t$  en el instante  $t$ , dadas las observaciones  $\mathbf{z}_{1:t} = \{\mathbf{z}_1, \dots, \mathbf{z}_t\}$ . Para ello, es necesario calcular la *pdf*  $p(\mathbf{x}_t|\mathbf{z}_{1:t})$ . La *pdf* inicial  $p(\mathbf{x}_0|\mathbf{z}_0) \equiv p(\mathbf{x}_0)$  se asume conocida. Así, la *pdf a posteriori*  $p(\mathbf{x}_t|\mathbf{z}_{1:t})$  se puede calcular en dos etapas:



**Figura 1.** Esquema general del filtro de partículas.

1. **Predicción:** supóngase que se dispone de la *pdf*  $p(\mathbf{x}_{t-1}|\mathbf{z}_{1:t-1})$  en el instante  $t - 1$ . La etapa de predicción implica el uso del modelo del sistema para obtener de manera recursiva la *pdf a priori*  $p(\mathbf{x}_t|\mathbf{z}_{1:t-1})$  en el instante siguiente  $t$ , mediante la ecuación de Chapman-Kolmogorov:

$$p(\mathbf{x}_t|\mathbf{z}_{1:t-1}) = \int p(\mathbf{x}_t|\mathbf{x}_{t-1})p(\mathbf{x}_{t-1}|\mathbf{z}_{1:t-1})d\mathbf{x}_{t-1} \quad (1)$$

2. **Evaluación:** En el instante  $t$ , se dispone de una nueva medida ( $\mathbf{z}_t$ ) que se puede utilizar durante la actualización del estado del sistema mediante el uso del teorema de Bayes:

$$p(\mathbf{x}_t|\mathbf{z}_{1:t}) = \frac{p(\mathbf{z}_t|\mathbf{x}_t)p(\mathbf{x}_t|\mathbf{z}_{1:t-1})}{p(\mathbf{z}_t|\mathbf{z}_{1:t-1})} \quad (2)$$

Las ecuaciones (1) y (2) conforman la base de la solución bayesiana óptima. Esta propagación recursiva de la densidad *a posteriori* es únicamente un resultado conceptual, puesto que, en general, no se puede determinar analíticamente [9]. Por esta razón, se encuentra en la literatura un número significativo de trabajos que presentan modelos aproximados de esas funciones de distribución.

El algoritmo de filtro de partículas (PF) es una técnica para implementar filtros recursivos bayesianos [9], cuya idea clave consiste en representar la *pdf a posteriori* por un conjunto de muestras discretas con pesos asociados y calcular los estimados basándose en esas muestras y pesos [10]. Cuando el número

de muestras es lo suficientemente grande, esta representación se convierte en equivalente a la descripción funcional de la *pdf a posteriori* [10]. La figura 1 muestra una representación gráfica del funcionamiento de PF. Inicialmente, se genera una población de partículas utilizando una *pdf* conocida. En este momento se reciben nuevas medidas y se calcula el peso para cada partícula, utilizando la ecuación (2). El resultado es un conjunto de partículas con pesos asociados que constituyen una aproximación discreta de la *pdf a posteriori*. A continuación, se lleva a cabo el paso de remuestreo. En esta etapa, se seleccionan las partículas “más importantes” (es decir, con mayor peso) para aproximar la pdf a priori en el instante siguiente aplicando sobre este conjunto la regla de predicción dada por la ecuación (1).

### 3. Problemas de optimización y metaheurísticas

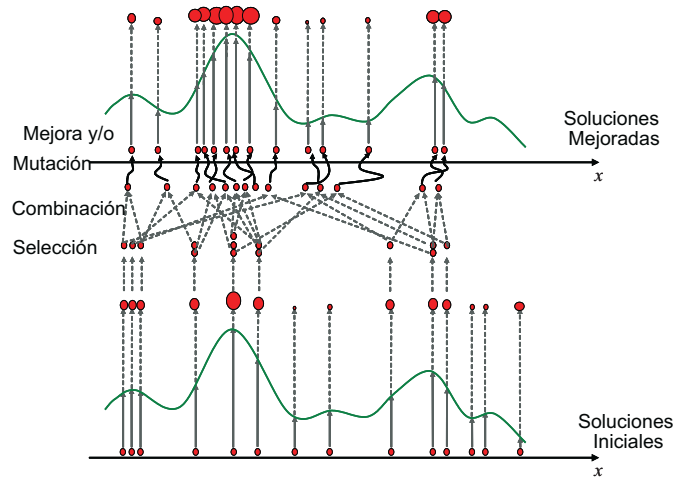
Un problema de optimización (OP) se puede definir como la mejora (minimización o maximización) de una función objetivo  $f : S \rightarrow \mathbb{R}$  sujeta a una serie de restricciones [11]. Formalmente,

$$OP = \begin{cases} \text{optimizar} & f(x) \\ \text{sujeto a} & x \in F \end{cases} \quad (3)$$

donde  $S$  es con un conjunto de soluciones, en general:  $S \subseteq \mathbb{R}^n$ ,  $f : S \rightarrow \mathbb{R}$  es una función de coste que asigna a cada solución candidata ( $x \in S$ ) un valor numérico ( $f(x) \in \mathbb{R}$ ) y  $F$  determina el conjunto de soluciones factibles  $x \in F \subseteq S$ .

Existe una colección importante de problemas interesantes en diferentes áreas de la ciencia y la ingeniería para la que no se dispone de algoritmos exactos que permitan encontrar la solución óptima en tiempos razonables. Una alternativa consiste en diseñar algoritmos aproximados que encuentren soluciones de alta calidad en un tiempo asumible. De entre todos los métodos aproximados, destacan las metaheurísticas por su eficiencia, efectividad y flexibilidad. Las metaheurísticas combinan métodos heurísticos básicos en un marco de trabajo de más alto nivel con el objetivo de mejorar la exploración sistemática del espacio de búsqueda del problema. Las metaheurísticas se han aplicado con éxito a una gran variedad de problemas de optimización. Este tipo de métodos combinan ideas que provienen de cuatro campos de investigación bien distintos: las técnicas de diseño de algoritmos (resuelven una colección de problemas), algoritmos específicos (dependientes del problema que se quiere resolver), fuentes de inspiración (del mundo real) y métodos estadísticos.

En una primera aproximación, se pueden clasificar las metaheurísticas en dos grandes bloques [12][13]: trayectoriales y poblacionales. Las metaheurísticas trayectoriales manejan en todo momento una sola solución y deben su nombre a que el proceso de búsqueda que desarrollan se caracteriza por una trayectoria en el espacio de soluciones. Es decir, partiendo de una solución inicial, generan un camino en el espacio de búsqueda mediante operaciones de movimiento. Dentro de estas metaheurísticas se pueden destacar por su interés la Búsqueda Tabú (*Tabu Search* - TS), Procedimiento Aleatorio y Adaptativo de Búsqueda



**Figura 2.** Esquema general de las metaheurísticas poblacionales.

Miope (*Greedy Randomized Adaptive Search Procedure* - GRASP) o Recocido Simulado (*Simulated Annealing* - SA). Las metaheurísticas poblacionales implementan el proceso de búsqueda manteniendo simultáneamente un conjunto de soluciones. Ejemplos de este tipo de metaheurísticas son los Algoritmos Evolutivos (*Evolutionary Algorithms* - EA), la Búsqueda Dispersa (*Scatter Search* - SS), el Reencadenamiento de Trayectorias (*Path Relinking* - PR) y los Algoritmos de Estimación de la Distribución (*Estimation Distribution Algorithms* - EDA). Las metaheurísticas poblacionales son de especial interés para nuestra propuesta de hibridación, puesto que encajan perfectamente en el marco de trabajo del filtro de partículas, cuyo carácter es eminentemente poblacional.

En la figura 2 se representa gráficamente el esquema de funcionamiento general de una metaheurística poblacional. Inicialmente, se parte de soluciones generadas aleatoriamente. En una segunda etapa, se seleccionan aquellas que poseen una mayor calidad para generar otras mediante combinación. A las soluciones generadas, se les puede aplicar una etapa de mutación y otra de mejora, con el objetivo de dotar a la población de diversidad o calidad, respectivamente. Este proceso se repite hasta que se alcanza una condición de parada establecida previamente.

#### 4. El Filtro de Partículas Metaheurístico

El Filtro de Partículas Metaheurístico (MPF) es una metodología de desarrollo de algoritmos híbridos entre filtros de partículas y metaheurísticas (o heurísticas), para la resolución de problemas de optimización dinámica. Se

debe distinguir entre el ámbito de la metodología del MPF y el de los algoritmos que se derivan de su aplicación.

La metodología MPF se aplica a algoritmos de estimación secuencial basados en filtros de partículas y en metaheurísticas. El propósito de esta metodología es el diseño de algoritmos de optimización dinámica mediante la hibridación de métodos de ambas familias. Es importante notar que los algoritmos resultantes son de propósito general en cuanto que los algoritmos de partida lo son, de modo que se pueden aplicar a un conjunto amplio de problemas dinámicos, pero son específicos en cuanto que están especializados en optimización dinámica.

#### 4.1. Los elementos de la metodología MPF

En esta sección se introduce el conjunto de definiciones necesarias para describir los elementos que componen el marco de trabajo de MPF. Gran parte de estos elementos son heredados de los algoritmos de filtros de partículas y de las metaheurísticas.

**Definition 1.** Una ***solución*** ( $\mathbf{x} \in S \equiv \mathbb{R}^D$ ) constituye la información necesaria y suficiente para representar una posible respuesta al problema de optimización.

**Definition 2.** Se entiende por ***medida*** ( $\mathbf{z} \in \mathbb{R}^M$ ) aquel conjunto de datos recogidos en un instante de tiempo, utilizando el proceso de medida (usualmente  $M > D$ ).

**Definition 3.** Dado un espacio de soluciones  $S$  y una medida  $\mathbf{z}$ , la ***función de ponderación*** ( $f : S \times \mathbb{R}^M \rightarrow \mathbb{R}$ ) es un operador que establece una relación entre el espacio de soluciones ( $\mathbf{x} \in S$ ) y las medidas ( $\mathbf{z}$ ).

**Definition 4.** Dadas una solución  $\mathbf{x}$ , una medida  $\mathbf{z}$  y una función de ponderación  $f$ , se denomina ***peso*** de la solución  $\mathbf{x}$ , y se denota  $\omega$ , al resultado de aplicar  $f$  sobre  $\mathbf{x}$  y  $\mathbf{z}$ , es decir:  $\omega = f(\mathbf{x}, \mathbf{z})$

**Definition 5.** Se define un ***individuo*** como la dupla formada por una solución  $\mathbf{x}$  y su peso  $\omega$ , y se denota como:  $(\mathbf{x}, \omega)$ .

En el marco PF, el par formado por una solución y un peso es conocido como *partícula*. Sin embargo, en el ámbito de las metaheurísticas, recibe diferentes nombres dependiendo del método. Por ejemplo, este elemento es conocido como *individuo* en GA, MA y EDA, mientras que en el contexto de SS y PR se llama *solución*.

**Definition 6.** El ***conjunto soporte*** (*SupportSet*) es el formado por  $N_s$  individuos que evolucionan en el tiempo guiados por un algoritmo de estimación secuencial.

En PF, *SupportSet* equivale al conjunto de partículas. En el contexto de los algoritmos de optimización, *SupportSet* juega el papel del conjunto inicial de soluciones.

**Definition 7.** El **conjunto mejorado** (*ImprovedSet*) es un subconjunto del conjunto soporte, formado por  $N_s$  individuos que forman el conjunto inicial para la etapa de optimización.

**Definition 8.** Un **Algoritmo de Estimación Secuencial Montecarlo (PF)** es un procedimiento que determina la evolución temporal de *SupportSet*, entendiéndola como una representación discreta de una pdf utilizando las etapas generales de predicción y actualización.

**Definition 9.** Una **Metaheurística (M)** es un algoritmo aproximado de optimización basado en poblaciones que dirige la combinación de soluciones de *ImprovedSet* para obtener otras de mayor calidad, en el contexto MPF.

**Definition 10.** El procedimiento de **Selección del conjunto mejorado (Select)** permite seleccionar adecuadamente un conjunto de individuos de *SupportSet* para formar *ImprovedSet*, siguiendo un determinado criterio.

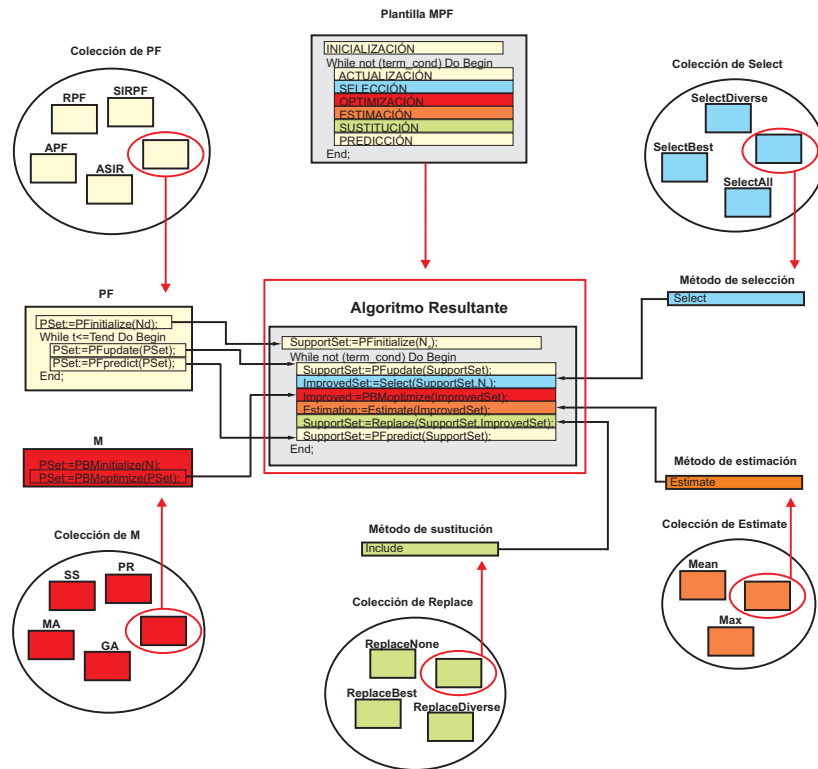
**Definition 11.** El procedimiento de **sustitución en el conjunto soporte (Replace)** permite que los individuos que forman parte del conjunto mejorado en el momento en el que finaliza el proceso de optimización, pasen a formar parte del conjunto soporte.

**Definition 12.** El procedimiento de **Estimación (Estimate)** establece una estimación de la mejor solución que describe al sistema en cada instante, basándose en una combinación adecuada de los individuos que forman parte del conjunto mejorado.

#### 4.2. Metodología de hibridación

MPF proporciona un método general de combinación de algoritmos de estimación secuencial (basados en filtros de partículas) con métodos metaheurísticos. Los algoritmos derivados de la aplicación de MPF están orientados a resolver problemas de optimización dinámica. MPF proporciona un método sistemático para la implementación de algoritmos híbridos que se puede resumir en la siguiente secuencia de pasos:

1. *Elegir un PF e identificar las etapas de inicialización, predicción y actualización.* La labor de las estrategias extraídas de PF consisten en inicializar y mantener el conjunto soporte de individuos *SupportSet* a lo largo del tiempo. Desde el punto de vista de M, el filtro de partículas juega el papel de inicializador de soluciones de la metaheurística (o generador de soluciones diversas), en cada instante temporal.
2. *Elegir una M e identificar las etapas de inicialización y optimización.* Esta etapa del algoritmo tiene como objetivo mejorar el conjunto de individuos *ImprovedSet* en cada instante temporal. Desde el punto de vista del PF, la metaheurística poblacional es un procedimiento de corrección de la estimación. Se debe recordar que PF posee sus propias estrategias para la corrección de la estimación.



**Figura 3.** Esquema general del Filtro de Partículas Metaheurístico (MPF) y cómo se obtiene un algoritmo derivado de dicho esquema.

3. *Diseñar el procedimiento de selección.* El procedimiento de selección del conjunto mejorado debe elegir  $N_i$  individuos de *SupportSet* para formar el conjunto *ImprovedSet*, siguiendo un determinado criterio.
4. *Diseñar un procedimiento de sustitución.* El procedimiento de sustitución se utiliza para añadir los individuos del conjunto mejorado *ImprovedSet* en *SupportSet*. Como el tamaño de *SupportSet* ( $N_s$ ) debe permanecer constante (ya que el número de partículas en los algoritmos PF se mantiene constante a lo largo del proceso de estimación), la incorporación de nuevos individuos implica forzosamente que otros deben ser descartados.
5. *Elegir un procedimiento de estimación del estado del sistema.* El procedimiento de estimación calcula un estimado del estado del sistema en cada instante, que constituye la salida del algoritmo.
6. *Organizar los elementos utilizando la plantilla MPF.* Una vez que se han obtenido los diferentes elementos descritos, se organizan utilizando

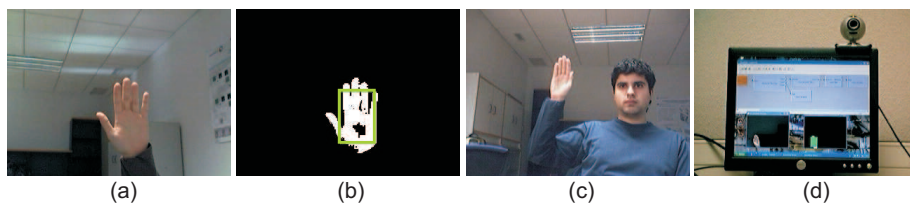
la plantilla MPF, que se muestra en la parte central de la figura 3. Las etapas encajan como piezas independientes en la plantilla para diseñar un algoritmo híbrido. El color de las cajas contenedoras indica la procedencia del procedimiento.

#### 4.3. Resolución de problemas dinámicos mediante la metodología MPF

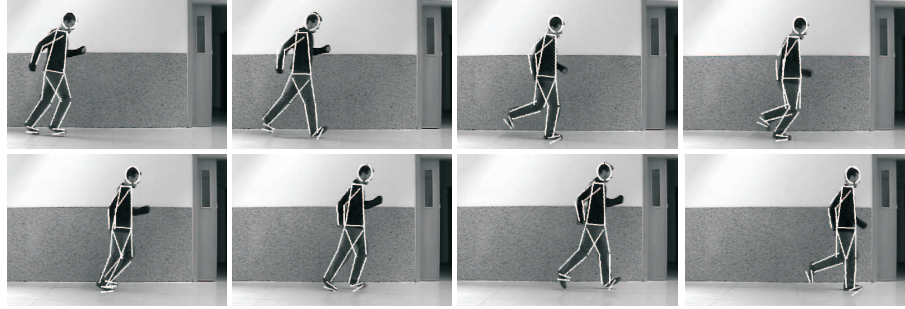
Para resolver un problemas dinámicos mediante la metodología MPF hay que considerar los siguientes pasos:

1. Determinar si el problema estacionario existe. En este caso:
  - a) Estudiar los métodos metaheurísticos que han sido aplicados con éxito a este problema
  - b) Elegir uno de ellos para ser hibridado con un filtro de partículassi no
  - a) Estudiar el problema dinámico desde el punto de vista de los filtros de partículas
  - b) Elegir una metaheurística o procedimiento heurístico adecuado en función de la dificultad del problema
2. Hibridar ambos métodos utilizando la metodología MPF
3. Determinar los parámetros adecuados mediante experimentación

Por lo tanto, para obtener el máximo rendimiento de esta metodología, es necesario, en primer lugar, estudiar el problema dinámico y también su versión estacionaria si existe. El estudio del problema debe orientarse a responder a preguntas como si es un problema difícil, si es necesario el concurso de una metaheurística o basta con una heurística, si prima la calidad de las soluciones o tiene restricciones severas de tiempo, etc.



**Figura 4.** Sistema Visual Ratón: a través de la imagen de la mano (a) y de una detección de color (b) se consigue que un usuario (c) equipado con una cámara web pueda manejar un ratón del ordenador (d).



**Figura 5.** Seguimiento de un corredor utilizando el algoritmo SSPF.

## 5. Resultados

En esta sección, se presentan brevemente algunos de los problemas dinámicos que se han abordado desde el marco de MPF. Se han agrupado en dos grandes bloques, según la temática abordada.

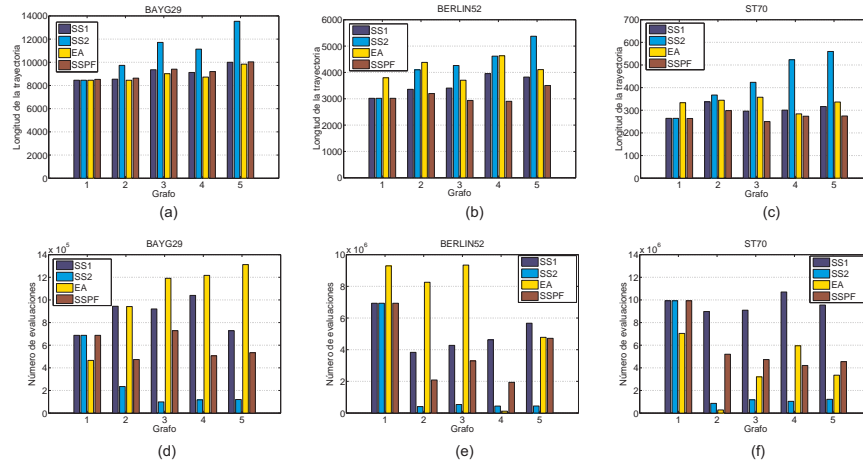
### 5.1. Seguimiento visual

En este problema se han utilizado tres algoritmos MPF: Filtro de Partículas con Búsqueda Local (LSPF), Filtro de Partículas con Búsqueda Dispersa (SSPF) y Filtro de Partículas con Reencadenamiento de Trayectorias (PRPF). LSPF es un algoritmo que combina un filtro de partículas con una búsqueda local sobre la mejor solución encontrada en cada instante de tiempo. Este algoritmo es muy útil para el seguimiento preciso de un objeto. Se ha aplicado al seguimiento de la mano de un usuario para dirigir la trayectoria del puntero del ratón del ordenador [14] (ver figura 4).

SSPF y PRPF son hibridaciones de filtro de partículas con las metaheurísticas búsqueda dispersa (SS) y reencadenamiento de trayectorias (PR). Ambos algoritmos se han aplicado con éxito a problemas de seguimiento visual más complejos como el seguimiento de múltiples objetos [15] y seguimiento de objetos articulados [16] (ver figura 5). En ambos casos, por su interés, se han aplicado los algoritmos resultantes al seguimiento del cuerpo humano.

### 5.2. Problemas combinatorios dinámicos

El problema dinámico del viajante de comercio (*Dynamic Travelling Salesman Problem - DTSP*) es una generalización del Problema del Viajante de Comercio (TSP), que consiste en encontrar la ruta más corta de un viajero que comienza su viaje en una ciudad determinada, visita un conjunto prescrito de ciudades una y sólo una vez y vuelve a la ciudad de partida. El DTSP es



**Figura 6.** Longitud de las trayectorias en (a) *BAYG29*, (b) *BERLIN52* y (c) *ST70* y tiempo de ejecución requerido en (d) *BAYG29*, (e) *BERLIN52* y (f) *ST70* utilizando SS1, SS2, MA y SSPF.

una versión dependiente del tiempo de este problema donde, en cada instancia, puede modificarse el grafo que lo define.

Se ha aplicado el Filtro de Partículas con Búsqueda Dispersa (SSPF) al DTSP [17]. Como resultado, se ha obtenido que el algoritmo SSPF presenta un mayor rendimiento que los algoritmos metaheurísticos con los que se ha comparado (Algoritmos meméticos y búsqueda dispersa). En la figura 6 se muestran los resultados obtenidos en calidad y tiempo de los diferentes algoritmos estudiados.

## 6. Conclusiones

En este trabajo, se ha perseguido identificar y extraer las mejores características de los filtros de partículas (PF) y de metaheurísticas, y utilizarlas en el diseño de algoritmos de optimización que puedan adaptarse a cambios. Como resultado, se ha propuesto un método de hibridación entre estos métodos, denominado *Filtro de Partículas Metaheurístico* (MPF). De su aplicación, se obtienen algoritmos que heredan las características más sobresalientes de cada método considerado: la capacidad de predicción y adaptación de PF y las estrategias de optimización de M. Se han aplicado los algoritmos MPF a diferentes problemas dinámicos complejos y se han comparado con otros métodos utilizados en la literatura para esos problemas, obteniendo los mejores resultados con los algoritmos propuestos. En definitiva, se ha podido comprobar, al menos sobre los experimentos analizados que, “*La combinación de estrategias de adaptación, predicción y optimización incrementa la eficiencia de la búsqueda de soluciones de alta calidad para problemas de optimización dinámica*”. Este

método híbrido establece un prometedor enfoque en el ámbito de la optimización dinámica.

## Referencias

1. Randall, M.: Constructive Meta-heuristics for Dynamic Optimization Problems. Technical Report. School of Information Technology. Bond University (2002)
2. Saleem, S. , Reynolds, R.: Cultural Algorithms in Dynamic Environments. Proceedings of the Congress on Evolutionary Computation (2000) 1513–1520
3. Mori, N., Kita, H.: Genetic algorithms for adaptation to dynamic environments - a survey. 26th Annual Conf of the IEEE Industrial Electronics Society, Vol 4 (2000) 2947–2952
4. Branke, J.: Evolutionary Approaches to Dynamic Optimization Problems - Updated Survey. GECCO Workshop on Evolutionary Algorithms for Dynamic Optimization Problems (2001) 27–30
5. Andrews, M., Tuson, A.: Diversity does not Necessarily Imply Adaptability. Proceedings of the Workshop on Evolutionary Algorithms for Dynamic Optimization Problems at the Genetic and Evolutionary Computation Conference (GECCO) (2003) 24–28
6. Dorigo, M., Ganbardella, L.M.: Ant Colony System: A Cooperative Learning Approach to the Travelling Salesman Problem. IEEE Transaction on Evolutionary Computation, 1:1 (1997) 53–66
7. Lee, M.W., Cohen, I., Jung, S.K.: Particle Filter with analytical inference for human body tracking. IEEE Workshop on Motion and Video Computing (2002)
8. Deutscher, J., Reid, I., Davidson, A.: Automatic Partitioning of High Dimensional Search Spaces associated with Articulated Body Motion Capture. Proc. of the IEEE Conf. on CVPR, vol. 2 (2001) 8–14
9. Doucet, A., Godsill, S., Andrieu, C.: On sequential Monte Carlo sampling methods for Bayesian filtering. Statistics and Computing, 10:3 (2000) 197 - 208
10. Arulampalam, S., Maskell, S., Gordon, N., Clapp, T.: A Tutorial on Particle Filters for On-line Nonlinear/Non-Gaussian Bayesian Tracking. IEEE Trans. on Signal Processing 50:2 (2002) 174–188
11. Yaigura, M., Ibaraki, T.: On metaheuristic algorithms for combinatorial optimization problems. Systems and Computers in Japan 3 (2001) 33–55
12. Blum, C., Roli, A.: Metaheuristics in combinatorial optimization: Overview and conceptual comparison. ACM Computing Surveys 35:3 (2003) 268–308
13. Gendreau, M.: An Introduction to Tabu Search. In F. Glover, G. A. Kochenberger editors, Handbook of Metaheuristic. Kluwer Academic Publishers (2003) 37-54
14. Pantrigo, J.J., Montemayor, A.S., Sánchez, A.: Local Search Particle Filter applied to Human-Computer Interaction. 4th Int. Symposium on Image and Signal Proc. and Analysis (2005) 279–284
15. Pantrigo, J.J., Montemayor, A.S., Cabido, R.: Scatter Search Particle Filter to for 2D Real-Time Hands and Face Tracking. LNCS 3617 (2005) 953–960
16. Pantrigo, J.J., Sánchez, A., Gianikellis, K., Montemayor, A.S.: Combining Particle Filter and Population-based Metaheuristics for Visual Articulated Motion Tracking. Electronic Letters on Computer Vision and Image Analysis 5:3 (2005) 68–83
17. Pantrigo, J.J., Sánchez, A., Cabido, R., Duarte, A.: Scatter Search Particle Filter to Solve the Dynamic Travelling Salesman Problem. LNCS 3448 (2005) 177–189