



## Revisión sobre algoritmos de optimización multi-objetivo genéticos y basados en enjambres de partículas

Review about genetic multi-objective optimization algorithms and based in particle swarm

Joaquín Javier Meza Álvarez<sup>1</sup> Juan Manuel Cueva Lovelle<sup>2</sup> Helbert Eduardo Espitia Cuchango<sup>3</sup>

**Para citar este artículo:** Meza, J., Cueva, J., y Espitia, H. (2015). Revisión sobre algoritmos de optimización multi-objetivo genéticos y basados en enjambres de partículas. *Revista Redes de Ingeniería*, 6(2), 54-76.

### Resumen

El enfoque evolutivo, así como el comportamiento social, han mostrado ser una muy buena alternativa en los problemas de optimización, donde se presentan varios objetivos a optimizar. De la misma forma, existen todavía diferentes vías para el desarrollo de este tipo de algoritmos. Con el fin de tener un buen panorama sobre las posibles mejoras que se pueden lograr en los algoritmos de optimización bio-inspirados multi-objetivo, es necesario establecer un buen referente de los diferentes enfoques y desarrollos que se han realizado hasta el momento. En este documento se revisan los algoritmos de optimización multi-objetivo más recientes, tanto genéticos como basados en enjambres de partículas. Se realiza una revisión crítica con el propósito de establecer las características más relevantes de cada enfoque, a fin de identificar las diferentes alternativas que se tienen para el desarrollo de un algoritmo de optimización multi-objetivo bio-inspirado.

**Palabras clave:** computación evolutiva, optimización multi-objetivo evolutiva.

### Abstract

The evolutionary approach as social behavior have proven to be a very good alternative in optimization problems where several targets have to be optimized. Likewise, there are still different ways to develop such algorithms. In order to have a good view on possible improvements that can be achieved in the optimization algorithms bio-inspired multi-objective it is necessary to establish a good reference of different approaches and developments that have taken place so far. In this paper the algorithms of multi-objective optimization newest based on both genetic and swarms of particles are reviewed. Critical review in order to establish the most relevant characteristics of each approach and thus identify the different alternatives have to develop an optimization algorithm multi-purpose bio-inspired design is performed.

**Keywords:** evolutionary computation, evolutionary multi-objective optimization.

1. Docente de planta, Universidad Distrital Francisco José de Caldas.
2. Docente de planta, Universidad de Oviedo.
3. Docente de planta, Universidad Distrital Francisco José de Caldas.

## INTRODUCCIÓN

Los algoritmos de optimización multi-objetivo bio-inspirados han mostrado ser una buena herramienta para la solución de problemas con diferentes funciones objetivo [1]. De los algoritmos más representativos se tienen los que se apoyan en evolución y también en enjambres de partículas. Específicamente, los algoritmos basados en el concepto de evolución logran un buen acercamiento al frente de Pareto, sin embargo requieren de muchas generaciones; por otro lado los algoritmos que se sustentan en enjambres de partículas presentan una alta tasa de convergencia, no obstante su principal desventaja es lograr un adecuado manejo de la diversidad [1], [2].

En el presente documento se realiza una revisión acerca del desarrollo que se ha tenido de los algoritmos de optimización multi-objetivo basados en el concepto de evolución, así como los inspirados en el comportamiento de enjambre de partículas. De otro lado, en un problema de optimización multi-objetivo se dispone de un conjunto de objetivos cuantificables y un conjunto bien definido de restricciones. Debido a que el conjunto de soluciones a menudo presenta compromisos entre los múltiples objetivos, la solución final considera tanto la técnica de optimización como los procesos de decisión, de los cuales se tienen: a priori, progresiva y a posteriori [3].

En el método a priori se combinan los diferentes objetivos en una función de costo (escalar). En el método progresivo la toma de decisiones y la optimización se realizan al mismo tiempo. La información de selección (preferencia) parcial se proporciona en la medida que se realiza la optimización, proporcionando un conjunto de soluciones a considerar. Por su parte, en el método a posteriori se presenta un conjunto de soluciones candidatas óptimas de Pareto y se elige a partir de ese conjunto [3].

## Esquema y objetivo de la revisión

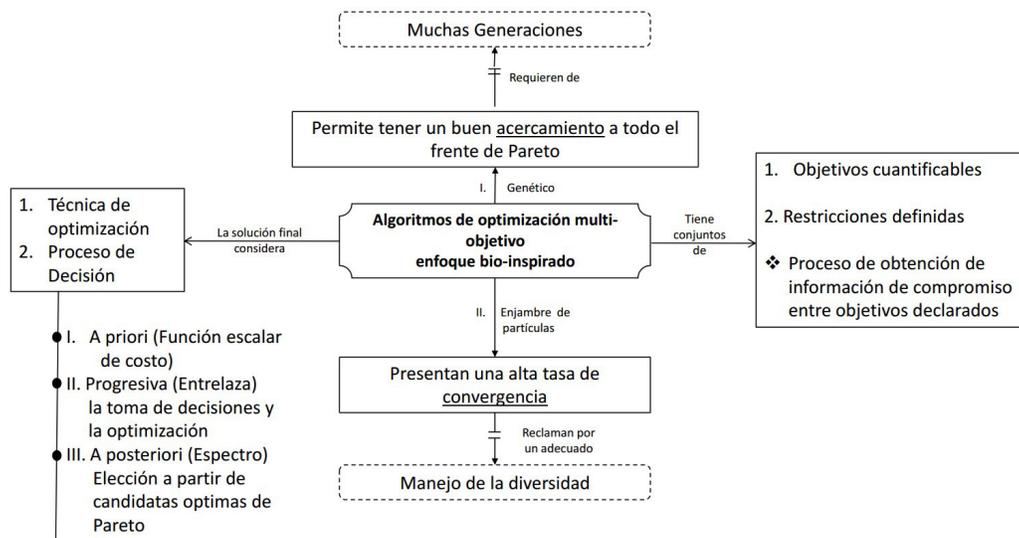
La revisión se realiza observando los diferentes enfoques orientados a resolver el problema de optimización multi-objetivo y los algoritmos desarrollados. La revisión sobre algoritmos genéticos Genetic Algorithm (AG) se realiza a fin de contextualizar los diferentes enfoques evolutivos multi-objetivo, adicionalmente se profundiza en el desarrollo realizado sobre algoritmos de optimización basados en enjambres de partículas, Particle Swarm Optimization (PSO) aplicados a problemas multi-objetivo. La mayoría de los algoritmos considerados emplean el concepto de frente de Pareto el cual consiste en un conjunto de soluciones con los mejores compromisos entre los objetivos a ser optimizados.

En la figura 1 se pueden apreciar las principales características identificadas de los algoritmos revisados, siendo características a resaltar la convergencia y la cantidad de generaciones empleadas. Por su parte, en la figura 2 se pueden apreciar los enfoques más significativos de los algoritmos genéticos y los basados en enjambres de partículas empleados en optimización multi-objetivo.

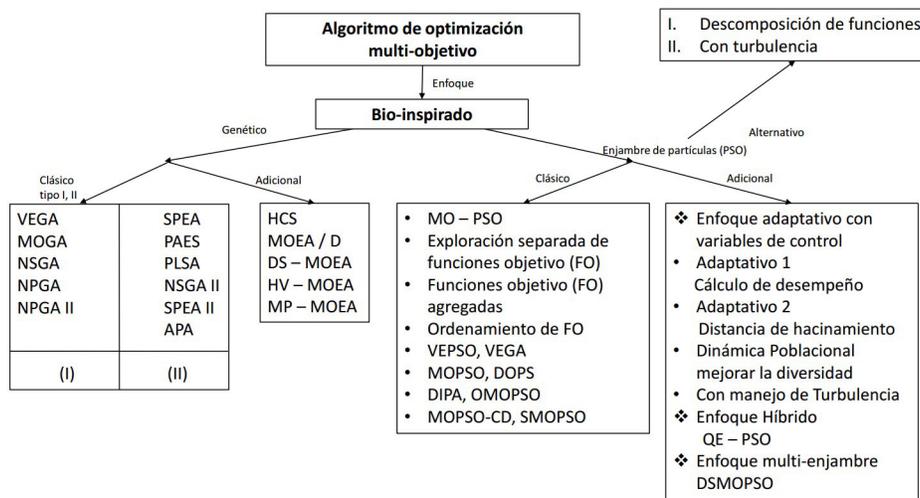
## ENFOQUES TRADICIONALES DE ALGORITMOS GENÉTICOS MULTI-OBJETIVO

Los algoritmos evolutivos multi-objetivo Multi-objective Evolutionary Algorithms (MOEAs) resultan ser una buena alternativa en ingeniería cuando se tienen varios criterios de diseño, los cuales pueden estar en conflicto cuando no se puede establecer una ponderación adecuada para tener una sola función objetivo [4], [5]. En esta sección se revisan las técnicas más populares desarrolladas en optimización multi-objetivo basadas en algoritmos genéticos.

Como una primera aproximación de los algoritmos genéticos se tiene: algoritmo genético para



**Figura 1.** Características principales de los algoritmos genéticos y basados en enjambres de partículas para optimización multi-objetivo.



**Figura 2.** Enfoques de algoritmos genéticos y basados en enjambres de partículas para optimización multi-objetivo.

el vector evaluado, Vector Evaluated Genetic Algorithm (VEGA); algoritmo genético multi-objetivo, Multi-Objective Genetic Algorithm (MOGA); algoritmo genético de ordenamiento no dominado, Nondominated Sorting Genetic Algorithm (NSGA); algoritmo genético con nichos de Pareto, Niche-Pareto Genetic Algorithm (NPGA) y algoritmo genético con nichos de Pareto II, Niche-Pareto Genetic Algorithm II (NPGA II) [1].

Un segundo grupo de algoritmos representativos son: el algoritmo evolutivo de fuerza de Pareto, Strength Pareto Evolutionary Algorithm (SPEA [6], [7]); estrategia de evolución archivada de Pareto, Pareto Archived Evolution Strategy (PAES [8]); algoritmo envolvente de Pareto basado en selección, Pareto Envelope-based Selection Algorithm (PESA [9]); algoritmo genético de ordenamiento no dominante, Non-dominated Sorting Genetic Algorithm

II (NSGA II [10]) y SPEA II [11]; algoritmo adaptativo de Pareto, Pareto Adaptive Algorithm (APA [12] [13]).

A continuación se presenta una revisión de los algoritmos evolutivos multi-objetivo, los cuales según [3] se consideran los más representativos.

### **Algoritmo genético para el vector evaluado**

En este algoritmo se divide la población original en un grupo de sub-poblaciones de tal forma que para cada función objetivo se tiene asociada una sub-población. Luego se realiza un proceso de selección para cada sub-población considerando la respectiva función objetivo asociada. En el siguiente paso las sub-poblaciones se mezclan para obtener una nueva población y de esta forma seguir con los procesos tradicionales de cruce y mutación utilizados en un algoritmo genético [4].

### **Algoritmo genético multi-objetivo**

Estrategia donde se emplean los elementos básicos de un algoritmo genético (cruce, mutación y selección) aplicados al problema multi-objetivo utilizando el concepto de dominancia de Pareto. Para la implementación del algoritmo se calcula el *ranking* de un individuo considerando el número de individuos que lo dominan. Adicionalmente, para realizar los cruces se emplea el valor del desempeño compartido [14].

En este enfoque a cada individuo de la población se le asigna un valor con el cual será ordenado para realizar el proceso de selección. El valor con el cual se ordenan los individuos (*ranking*) se asigna considerando el criterio de no dominancia. En un primer paso a todos los individuos se les asigna un valor de 1, para los individuos no dominados se mantienen el valor de 1, por su parte, los individuos dominados se penalizan sumando a valor indicial el número total de individuos que lo dominan. Posteriormente los individuos se ordenan

de menor a mayor considerando los valores calculados previamente, en este respectivo orden se asigna un valor de desempeño para cada individuo por interpolación desde el menor (mejor) hasta el mayor (peor). En el caso de tener individuos que comparten la misma posición en ranking se realiza un promedio para que tengan el mismo valor de desempeño. Finalmente al tener asignado a cada individuo su respectivo valor de desempeño se procede a realizar los procesos de cruce y mutación [14].

### **Algoritmo genético de ordenamiento no dominado**

En este algoritmo las soluciones se ordenan por su calidad, colocando los individuos no dominados en una lista asignando un valor de aptitud considerando el tamaño de la población. Con el fin de mantener la diversidad, los individuos no dominados comparten los valores de desempeño creando una lista de individuos no dominados [15].

En NSGA se utiliza un método de selección de clasificación para enfatizar soluciones no dominadas actuales y un método de nichos o segmentos especializados, el cual se utiliza para mantener la diversidad de la población.

### **Algoritmo genético con nichos de pareto**

En este algoritmo se realiza la selección por torneo basada en la dominancia, de forma aleatoria dos individuos se comparan con un subconjunto de la población, en este caso se escogen al azar dentro de la población dos individuos que compiten y un conjunto de comparación formado por otros individuos. Generalmente para la selección por torneo si un candidato está dominado por el grupo de comparación, mientras que el otro no lo es, este último se selecciona para la reproducción. En este algoritmo también se utiliza una técnica para compartir el fenotipo en el espacio de soluciones [16].

La principal mejora de NPGA II sobre NPGA consiste en utilizar el grado de dominación (el número de soluciones en la población actual que lo domina) como la puntuación determinante en la selección del torneo. En este caso el método del torneo es determinista, en contraste con el método probabilístico utilizado por NPGA [17].

### **Algoritmo evolutivo de fuerza de Pareto**

Con este enfoque se mantiene una población externa para cada generación almacenando todas las soluciones no dominantes obtenidas hasta el momento. Cada población externa es mezclada con la población actual donde su desempeño se asigna considerando el número de soluciones que una solución domina. Para asegurar la diversidad entre soluciones no dominantes se usa la técnica de agrupación determinista [6], [18].

### **Estrategia de evolución archivada de Pareto**

En este algoritmo un padre es generado por una mutación de un hijo. Si el hijo domina al padre, el hijo es aceptado como el siguiente padre y la iteración continúa. Si el padre domina al hijo, el hijo (descendencia) es descartado y se genera la nueva solución mutada. Si la descendencia y el padre no son dominantes el uno al otro se emplea un conjunto de comparación de individuos usados previamente. Para mantener la población diversa alrededor del frente de Pareto, se considera un archivo de soluciones no dominantes. Una nueva descendencia es comparada con un archivo para verificar si estos dominan cualquier miembro del archivo, si es así, entonces la descendencia completa es aceptada como un nuevo padre. Las soluciones dominantes son eliminadas desde el archivo.

### **Algoritmo genético de ordenamiento no dominante II**

En el caso del algoritmo NSGA II, primero se establece el número de soluciones que una determinada

solución domina así como el conjunto formado por las soluciones dominadas. Con la anterior información se obtiene el primer frente formado por las soluciones no dominantes. Del frente actual se realiza una reducción de las soluciones dominadas. Las soluciones no dominantes que son permanentes después de esta reducción se organizan en una lista separada. Estos procesos continúan usando el nuevo frente identificado como el frente actual. Una nueva población es formada considerando los frentes individuales desde los frentes hasta exceder el tamaño de la población. Las soluciones de los últimos frentes permitidos son clasificados de acuerdo a la relación de la cantidad de la población.

NSGA II usa un parámetro (denominado distancia de multitud) para estimar la densidad de cada individuo. La distancia de multitud de una solución es el promedio de la longitud del lado del cubo que encierra el punto sin incluir cualquier otro punto de la población. Las soluciones del último frente aceptado son clasificadas de acuerdo a la distancia respecto a la multitud.

### **Algoritmo evolutivo de fuerza de Pareto II**

El algoritmo SPEA 2 usa dos poblaciones, una actual y otra externa. Después de una evaluación todas las soluciones no dominantes de la población actual y de la población externa son pasadas a la siguiente población. Si el número de estas soluciones es menor que el tamaño de la población entonces la siguiente población es completada con individuos dominantes desde la actual y la población externa. La diferencia con SPEA consiste en la asignación del valor de desempeño y en el empleo de un operado de truncado. La función de ajuste es calculada de manera diferente para las soluciones desde la población externa y la actual. SPEA II incorpora información de la densidad de soluciones para discriminar entre los individuos que tienen un valor de ajuste idéntico. En este algoritmo el tamaño de archivo es fijo, adicionalmente cuando el número de individuos no dominados es menor que el

tamaño del archivo predefinido, el archivo se llena por individuos dominados [6], [18].

### Algoritmo adaptativo de Pareto

En este algoritmo cada solución se puede codificar en más de un alfabeto diferente [12], [13]. Adicionalmente la representación de una solución particular no es fija, es decir la representación es adaptable y se puede cambiar durante el proceso de búsqueda como efecto del operador de mutación.

Cada solución es representada con un número entero correspondiente al número de elementos del alfabeto y una cadena de símbolos del alfabeto. Si se tienen dos símbolos corresponde a la codificación binaria estándar. El alfabeto sobre el que se codifica cada individuo puede cambiar durante el proceso de búsqueda. Cada individuo se selecciona para la mutación, que es el operador de variación única. Los hijos y los padres son comparados. Si los hijos dominan al padre entonces la descendencia entera de la nueva población y del padre es removida. Si los padres dominan la descendencia después de varias mutaciones entonces se elige otro alfabeto. En este algoritmo se utiliza una única población de individuos.

### ENFOQUES ADICIONALES DE ALGORITMOS GENÉTICOS MULTI-OBJETIVO

Adicional a las técnicas revisadas anteriormente, las cuales se pueden considerar como estándar, se han realizado a lo largo del tiempo diferentes propuestas y enfoques para la solución del problema de optimización multi-objetivo mediante algoritmos evolutivos. A continuación se realiza una breve descripción de estos enfoques.

### Métodos híbridos

Sobre métodos híbridos en [19] se propone una técnica para reducir la sobrecarga cognitiva asociada

con la importancia relativa de un determinado criterio dentro de un entorno de diseño multi-objetivo, que implica un gran número de objetivos. En este trabajo se realiza la integración con varias técnicas de búsqueda de diseño basado en algoritmos genéticos y optimización (suma ponderada, Pareto ponderado, métodos coevolutivos ponderados y escenarios ponderados).

En [20] se propone una nueva estrategia de búsqueda local para algoritmos meméticos multi-objetivo. Se describe un procedimiento de búsqueda iterativa, denominado escalado de la colina con pasos laterales, Hill Climber with Sidestep (HCS). La característica de HCS consiste en la capacidad de moverse hacia y a lo largo del conjunto de Pareto dependiendo de la distancia de la iteración actual hacia este conjunto. El procedimiento de búsqueda local utiliza la geometría de los conos de dirección de este tipo de problemas de optimización y funciona con o sin información del gradiente.

Otro trabajo se puede apreciar en [21], donde se propone un algoritmo evolutivo multi-objetivo híbrido basado en un doble espacio para realizar la optimización multi-objetivo, Double Space Based Multiobjective Evolutionary Algorithms (DS-MOEA). En comparación con MOEAs tradicionales, DS-MOEA no solo considera el proceso evolutivo en el espacio de la solución, también tiene en cuenta el proceso de aprendizaje de los conocimientos en el espacio objetivo.

Según [22], para los problemas de optimización multi-objetivo no se tiene mayor desarrollo sobre métodos que emplean el gradiente y su uso algorítmico. En [22] se expone cómo utilizar los gradientes para optimización numérica de valor real. Específicamente, se proporciona una descripción paramétrica analítica del conjunto de direcciones de todas las soluciones no dominadas, donde una solución puede ser desplazada de tal forma que los valores objetivos mejoren o sigan siendo los mismos. Además se utiliza el resultado para evaluar la

utilidad de usar la información del gradiente para la optimización multi-objetivo, donde el objetivo es obtener un conjunto de soluciones de Pareto que se aproxima al conjunto óptimo de Pareto.

## Exploración de regiones

Según [23], los algoritmos multi-objetivo evolutivos (MOEA) basados en hipervolumen parecen ser la primera opción cuando se requiere manejar varios objetivos (más de tres objetivos). También se comenta que estudios experimentales han demostrado que los algoritmos de búsqueda basados en hipervolumen pueden superar algoritmos como NSGA II y SPEA II. Un problema con la mayoría de los algoritmos basados en hipervolumen consiste en el orden de complejidad que este tiene; el algoritmo más conocido para calcular la hipervolumen necesita tiempo exponencial. En [23] se propone un enfoque general para la reducción de objetivos con el fin de mejorar los algoritmos basados en hipervolumen. El estudio indica que la omisión (temporal) de los objetivos puede mejorar los MOEAs basados en hipervolumen drásticamente en términos de los valores de los indicadores hipervolumen obtenidos.

Por otra parte, en [24] se desarrolla un algoritmo evolutivo elitista para toma de decisiones multi-objetivo. Los algoritmos definen un territorio alrededor de cada individuo para prevenir el hacinamiento en cualquier región. Esto mantiene la diversidad al tiempo que facilita la rápida ejecución del algoritmo. También se ha desarrollado un mecanismo para incorporar información de preferencia con el fin de centrarse en las regiones que son de interés para la toma de decisiones.

## Descomposición de objetivos

Según [25], la descomposición es una estrategia básica en la optimización multi-objetivo tradicional. Sin embargo, todavía no ha sido ampliamente utilizado en la optimización evolutiva

multi-objetivo. En [25] se propone un algoritmo evolutivo multi-objetivo basado en descomposición, Multiobjective Evolutionary Algorithm based on Decomposition (MOEA/D). Se descompone un problema de optimización multi-objetivo en una serie de subproblemas de optimización escalares y se optimizan simultáneamente. Cada subproblema se optimiza utilizando la información de sus varios subproblemas vecinos, por lo cual MOEA/D tiene menor complejidad computacional en cada generación que algoritmos genéticos de ordenamiento no dominado.

Adicionalmente, en [26] se presenta un algoritmo evolutivo multi-objetivo cuando cada objetivo tiene una importancia diferente. Para resolver este tipo de problemas se propone un método de poda a fin de considerar la desviación de obtener un conjunto de soluciones no dominadas con objetivos optimizados específicamente. Se observa que enfatizar en un objetivo mejora el rendimiento de optimización en problemas con muchos objetivos.

Según [27] al dividir la optimización multi-objetivo del espacio de decisión en varias regiones pequeñas se puede tener un mejor desempeño. De esta forma en [27] se propone el algoritmo de optimización multi-objetivo basado en la búsqueda sub-regional. Dado que el algoritmo propuesto utiliza la búsqueda sub-regional, la complejidad computacional en cada generación es menor que NSGA II. El algoritmo propuesto hace uso de la estrategia de max-min como funciones de desempeño para lograr una solución distribuida uniformemente sobre el frente de Pareto.

En [28] se expone que cuando en un algoritmo evolutivo se necesitan varias evaluaciones de la función objetivo con el fin de llegar a una solución sub-óptima y cada una de estas evaluaciones es computacionalmente costosa, entonces, dichos problemas pueden ser intratables. Reemplazar el problema original con un modelo sustituto ha sido una estrategia habitual para la reducción de

tiempo. En [28] se estudian técnicas de meta-modelado como funciones de Base Radial, máquinas de soporte vectorial, regresión polinómica en diferentes aspectos, tales como la precisión, robustez, eficiencia y escalabilidad, con el objetivo de identificar las ventajas y desventajas de cada meta-modelado con el fin de elegir el más adecuado para ser empleado con algoritmos evolutivos multi-objetivo.

Por su parte, en [29] se propone otro algoritmo evolutivo multi-objetivo basado en descomposición, el cual se centra en cómo mantener un conjunto de sub-problemas para aproximarse a la solución óptima. El algoritmo emplea un generador basado en modelos multivariados de Gauss. Tanto la información local y global de la población se extrae por un conjunto de modelos de distribución de Gauss.

En [30] se descompone el problema de optimización multi-objetivo restringido en varios subproblemas y luego se optimiza cada subproblema de forma colaborativa. Cada subproblema tiene su propia subpoblación y un registro temporal. La subpoblación se compone de aquellos individuos que tienen los mejores valores de los objetivos y menores violaciones de restricción del subproblema, mientras que el registro temporal se compone de aquellos individuos que se encuentran antes. Se realiza el operador de cruce entre cada individuo en las subpoblaciones y un individuo que se elige al azar del registro temporal correspondiente. La estrategia de registro temporal hace que los individuos que tienen mejores valores de los objetivos y menores violaciones de las restricciones, tengan la oportunidad de participar en el cruce y mutación.

Según [31] aunque el enfoque de descomposición generalmente converge más rápido a través de la optimización de todos los sub-problemas a la vez, hay dos aspectos que no se abordan completamente. El primero consiste en que la distribución de soluciones depende a menudo de una descomposición a priori del problema y el segundo es la falta de diversidad de la población entre

los sub-problemas. En [31] se desarrolla un MOEA con archivos de doble nivel un archivo global y un sub-archivo. En cada generación existe un proceso de autorreproducción con el archivo global y reproducción transversal entre los sub-archivos y el archivo global para crear nuevos individuos. El archivo global y sub-archivos se comunican a través de la reproducción cruzada y se actualizan usando los individuos reproducidos. Con este enfoque se conserva la convergencia rápida y al mismo tiempo se logra la distribución de la solución a lo largo de frente de Pareto.

### Múltiples poblaciones

Sobre el desarrollo de algoritmos que emplean varias poblaciones en [32] se desarrolla un algoritmo basado en el concepto de la coevolución de una familia de preferencias, junto con una población de soluciones candidatas.

Por su parte en [33] se propone un algoritmo evolutivo multi-objetivo basado en la evolución paralela de múltiples poblaciones y una población del archivo de Pareto. Para cada población se emplea un algoritmo evolutivo a fin de optimizar por separado cada una de las funciones, donde los individuos son generados por la selección (torneo) de la unión de una población asociada a un objetivo y la población del archivo de Pareto. En cada iteración una población de tamaño finito del archivo de Pareto se actualiza de forma iterativa y se recorta por una operación de comparación.

### Propuestas para lograr mejoras de los algoritmos

En [34] se expone que una forma de mejorar la velocidad de los algoritmos evolutivos multi-objetivo consiste en utilizar un archivo activo de soluciones élite no dominadas. Sin embargo, la preservación de todos los individuos de élite es costosa en tiempo (debido a la comparación lineal con todas las soluciones archivadas). El mantenimiento de una

población de élite de un tamaño máximo fijo (por la agrupación u otros medios) alivia este problema, pero puede causar un alejamiento y la disminución de los frentes de Pareto estimados, lo cual puede afectar la eficiencia del proceso de búsqueda. En [34] se consideran algunas estructuras de datos para facilitar el uso de un archivo de élite sin restricciones, sin la necesidad de una comparación lineal para el conjunto de élite en cada nuevo individuo insertado.

Un trabajo adicional se puede apreciar en [35] donde se proponen dos algoritmos evolutivos utilizando un nuevo gen inteligente Intelligent Gene Collector (IGC) para resolver grandes problemas individuales y de optimización multi-objetivo. IGC es la fase principal de un operador de recombinación inteligente, IGC utiliza el enfoque de dividir y conquistar, que consiste en dividir adaptativamente dos individuos de los padres en pares de segmentos de genes.

En [36] se propone un método de cálculo de dominio local, para mejorar los algoritmos evolutivos multi-objetivo realizando una búsqueda distribuida basada en el dominio local. En este método, primero se transforman todos los vectores de la aptitud de los individuos a coordenadas polares vectores en el espacio de la función objetivo. Luego se divide la población en varias subpoblaciones utilizando ángulos de declinación. Finalmente se calcula el dominio local para los individuos que pertenecen a cada sub-población, con base en la dirección de búsqueda local y posteriormente se aplica la selección, la recombinación y mutación individual dentro de cada subpoblación.

Por su parte en [37] se estudian los efectos en el rendimiento de la dominación local y recombinación local aplicado a problemas multi-objetivo combinatorios. Se introduce un método que crea una vecindad alrededor de cada individuo y asigna el orden de dominancia local después de girar la dirección principal de búsqueda de la vecindad,

mediante el uso de coordenadas polares en el espacio objetivo. Para la recombinación se crea una vecindad diferente empleando búsqueda aleatoria.

En [38] se presenta un algoritmo evolutivo para la optimización con restricciones. El algoritmo se basa en técnicas de optimización multi-objetivo, es decir, un individuo de la población de los padres puede ser reemplazado si está dominado por un individuo no dominado elegido de la población descendiente. Además, se utiliza un archivo de soluciones factibles y mecanismo de sustitución. El operador de cruce se utiliza para enriquecer las habilidades de exploración y explotación.

Particularmente en [39] se realiza una propuesta para inducir una mejor distribución de soluciones no dominadas y distribuir más equitativamente la selección entre ellas. Con el método propuesto individuos similares se eliminan en el proceso de evolución mediante el uso de la distancia entre los individuos en el espacio objetivo.

En [40] se propone que en el proceso de evolución un individuo de la población de los padres puede ser reemplazado si está dominado por un individuo no dominado en la población descendiente. Además, se introducen tres modelos de un algoritmo basado en la población y un mecanismo de solución de archivado y la sustitución no factible.

Según [41] las soluciones óptimas de un problema de optimización multi-objetivo corresponden a un frente no dominado que se caracteriza por una solución de compromiso entre los objetivos. Una región de "rodilla" en este frente óptimo de Pareto, visualmente es una protuberancia convexa en la parte delantera, la cual es importante para la toma de decisiones en contextos prácticos, ya que a menudo constituye el óptimo en equilibrio. En [41] se presenta un esquema de selección que permite a un algoritmo evolutivo multi-objetivo obtener un conjunto no dominado con concentración controlable alrededor de regiones de rodilla

existentes del frente de Pareto. Este enfoque basado en preferencia se logra mediante la optimización de un conjunto de sumas ponderadas de los objetivos originales.

En [42] se presenta otro algoritmo basado en el frente óptimo de Pareto, que busca aliviar la desventaja de los MOEA en el rendimiento del tiempo. En este algoritmo se emplea un enfoque de cruce limitado para mantener la diversidad de soluciones.

Según [43] varios algoritmos de optimización multi-objetivo evolutivos han adoptado los operadores de selección basados en indicadores que aumenten o reemplacen el *ranking* de los indicadores de calidad. Un indicador de calidad mide la bondad de cada candidato solución. Muchos indicadores de calidad se han propuesto con la intención de capturar diferentes preferencias en la optimización. Por lo tanto, los operadores de selección basados en indicadores tienden a tener presiones de selección sesgada que evolucionan individuos hacia determinadas regiones en el espacio objetivo. En [43] se estudia un método para agregar (o aumentar) los operadores de selección basados en indicadores existentes.

Según [44] buena parte de los algoritmos evolutivos multi-objetivo existentes presentan dificultades cuando el frente de Pareto óptimo se encuentra compuesto por múltiples segmentos desconectados. También los MOEAs, comúnmente involucran más de tres parámetros de control, adicionalmente su desempeño en función los ajustes de los parámetros son generalmente desconocidos [44]. Considerando lo anterior en [44] se propone un MOEA, con dos operadores genéticos, cruce de aritmética extendida y mutación diversificada, con los cuales se busca mejorar la capacidad de exploración del algoritmo.

Finalmente en [45] se propone un método de selección adaptativa, *Adaptative Operator Selection* (AOS), el cual se utiliza para determinar las tasas de aplicación de los diferentes operadores considerando

sus acciones recientes dentro de un proceso de optimización. Con el fin de realizar un seguimiento de la dinámica del proceso de búsqueda se utiliza una ventana deslizante para registrar las tasas de mejora de la aptitud obtenida por los operadores.

## ENFOQUES TRADICIONALES DE ALGORITMOS PSO MULTI-OBJETIVO

A continuación se presentan diferentes enfoques para extender el algoritmo PSO a problemas multi-objetivo [46]. En esta sección se revisan las propuestas más conocidas de las cuales resaltan: algoritmos con exploración separada de funciones, enfoques de vector evaluado no basados en Pareto y algoritmos basados en dominancia de Pareto.

### Algoritmos con exploración separada de funciones objetivo

En esta categoría se encuentran los enfoques que combinan todas las funciones objetivo en una sola o consideran cada función objetivo por turnos para la evaluación del enjambre. La ventaja de estos enfoques es su sencilla actualización del enjambre y las mejores posiciones. En este algoritmo se emplea un registro externo para el almacenamiento de soluciones no dominadas. Su inconveniente es la falta de una información a priori con respecto a una mejor manipulación de las distintas funciones objetivo [46].

### Enfoque de función objetivo agregada

En estos enfoques se agregan mediante una combinación ponderada todas las funciones objetivo en una sola. Si los pesos permanecen fijos durante la ejecución del algoritmo se tiene el caso de la agregación ponderada convencional, *Conventional Weighted Aggregation* (CWA) la cual es incapaz de detectar soluciones en las regiones cóncavas de la frontera de Pareto, para evitar esto los pesos son ajustados dinámicamente durante la optimización. Algunos de estos enfoques son la agregación de

explosión ponderada Bang-Bang Weighted Aggregation (BWA) y la agregación dinámica ponderada Dynamic Weighted Aggregation (DWA) [46]. El uso de BWA resulta en cambios bruscos de los pesos que fuerzan el algoritmo a seguir moviéndose hacia el frente de Pareto. El mismo efecto se logra con DWA, aunque el cambio en los pesos es más suave, los enfoques DWA tienen mejor desempeño que los BWA en fronteras de Pareto convexas. En [47] se propone la primera aproximación de PSO multi-objetivo de agregación ponderada usando los enfoques de CWA, BWA y DWA, que proporcionaron fronteras de Pareto con esparcimiento satisfactorio.

En [48] se considera un enfoque similar, donde el enjambre se divide en sub-enjambres y cada uno utiliza un ajuste de peso específico. La mejor partícula de cada enjambre sirve como un líder solo para sí mismo. Adicionalmente se usa una decisión preliminar de Pareto con el fin de buscar más a fondo los puntos que son soluciones candidatas de Pareto óptimas.

Adicionalmente en [49] se propone un enfoque de pesos modificados dinámicamente, incorporando un operador de mutación para evitar el estancamiento del enjambre, así como un término de aceleración que incrementa la convergencia en etapas posteriores del algoritmo. Las dos posiciones, nueva y antigua, son evaluadas e introducidas en una lista a la cual se aplica la técnica de clasificación no dominada encargada de seleccionar las partículas no dominadas que sufren un proceso de mutación en un intento de mejorarlas [50]. El conjunto de partículas resultante constituye el enjambre en la siguiente iteración del algoritmo.

### **Enfoque con ordenamiento de las funciones objetivo**

Estos enfoques requieren establecer una clasificación de las funciones objetivo, la minimización se realiza para cada función de forma independiente, partiendo por las más importantes.

Originalmente en [51] se propone un esquema que implementa el ordenamiento de funciones. Este algoritmo fija la función objetivo más simple y minimiza el resto de las funciones objetivo, utilizando una variante de PSO con vecindarios dinámicos. Este algoritmo no usa ningún registro externo y las soluciones no dominadas son almacenadas como mejores posiciones de las partículas.

Finalmente en [52] se puede observar una extensión del enfoque anterior donde se incorpora un registro externo para almacenar las soluciones no dominadas y reducir el costo computacional.

### **Enfoques de vector evaluado no basados en Pareto**

Inicialmente en [47] se propuso el esquema vector evaluado PSO, Vector Evaluated Particle Swarm Optimization (VEPSO) basado en la idea del algoritmo genético de vector evaluado, Vector Evaluated Genetic Algorithm (VEGA), en este enfoque hay un enjambre dedicado para cada función objetivo y es evaluado solamente para esta función. Las mejores posiciones de un enjambre se utilizan para la actualización de la velocidad de otro enjambre.

Adicionalmente en [53], se desarrolla una nueva versión de VEPSO donde cada enjambre es asignado a un procesador y el número de enjambres no es necesariamente igual al número de funciones objetivo. La comunicación entre los enjambres se realiza a través de un esquema similar a la topología de vecindario tipo anillo.

Un enfoque similar a VEPSO, fue propuesto en [54] llamado PSO Multi-Species que utiliza sub-enjambres que forman especies, una para cada función objetivo. Cada sub-enjambre se evalúa solamente con su propia función objetivo y la información de las mejores partículas se comunica a sub-enjambres vecinos, con un término adicional en la ecuación de actualización para la velocidad de las partículas.

## Algoritmos basados en dominancia de Pareto

Estos enfoques usan el concepto de dominancia de Pareto para determinar las mejores posiciones (líderes) que guiarán el enjambre durante la búsqueda. Al respecto en [2] se propuso el PSO multi-objetivo, Multi-Objective Particle Swarm Optimization (MOPSO), uno de los primeros enfoques PSO basados en Pareto donde las soluciones no dominadas detectadas por las partículas se almacenan en un repositorio. El espacio de búsqueda se divide en hipercubos y a cada hipercubo se le asigna un valor de aptitud que es inversamente proporcional al número de partículas que contiene. Utiliza una ruleta clásica para seleccionar un hipercubo y un líder. La mejor posición se actualiza en cada iteración, basado en la relación de dominación entre la mejor posición existente de la partícula y su nueva posición. El registro tiene un tamaño limitado y las nuevas posiciones se insertan basándose en el criterio de retención que da prioridad a las soluciones situadas en las zonas menos pobladas del espacio objetivo.

Otro adelanto se puede observar en [55], donde se propone un esquema PSO multi-objetivo que se ocupa de las ineficiencias causadas por el truncamiento del registro limitado de soluciones no dominadas, utilizando una estructura árbol sin restricciones para el mantenimiento de registro llamada árbol dominado. Trabaja de manera similar a MOPSO, excepto por el repositorio que se mantiene a través de las estructuras antes mencionadas. Adicionalmente usa un operador de mutación (locura) sobre la velocidad de las partículas para preservar la diversidad.

En [56] se presenta un método que emplea el estimador de densidad del vecino más cercano en combinación con un esquema de ruleta para la selección de líderes. Los líderes seleccionados son utilizados para actualizar la posición del resto de partículas. Los líderes con radio de apiñamiento superior tienen una mayor probabilidad de selección

ya que promueven la propagación uniforme de soluciones en la frontera de Pareto.

Por otra parte, en [57] se puede observar un método basado en un esquema elitista de registro que utiliza dos funciones: una para realizar selección y otra para eliminar un valor de aptitud de cada partícula. La selección del valor de aptitud es una medida de la influencia de la partícula a la propagación de la frontera de Pareto y aumenta con la distancia de sus vecinos más cercanos.

Un desarrollo de algoritmos de optimización multi-objetivo de enjambre de partículas inspirados en algoritmos evolutivos se presenta en [58], donde la actualización de las partículas es completamente diferente a cualquier algoritmo de PSO. Las ecuaciones de actualización de las partículas son sustituidas por un árbol de herencia de probabilidad y las partículas en vez de moverse en el espacio de búsqueda con una velocidad adaptable, heredan los parámetros de su nueva posición mediante una asignación dinámica de la probabilidad de herencia que controla las probabilidades teniendo en cuenta la retroalimentación del estado convergente del algoritmo y la aptitud de la mejor partícula en general.

Una propuesta de varios algoritmos MOPSO se aprecia en [59], en los cuales se incorporan esquemas especiales para la selección de los miembros del registro que participan en la actualización de la velocidad de las partículas. Se propone un enfoque MOPSO en combinación con el método sigma que asigna un valor numérico a cada partícula y miembro del registro; una partícula utiliza como líder el miembro del registro con el valor sigma más cercano al suyo y utiliza un factor de turbulencia (mutación) para la actualización de la posición de la partícula.

Una propuesta denominada max-min PSO se presenta en [60], con este enfoque se utiliza la función de aptitud max-min donde solo los vectores

decisión con un valor de la función max-min menor que cero pueden ser soluciones no dominadas con respecto a la población actual. La función max-min promueve la diversidad del enjambre ya que penaliza a las partículas que se juntan en grupos, favorece las soluciones en medio de fronteras convexas y en los extremos en fronteras cóncavas. De la misma forma las soluciones no dominadas se almacenan en un registro para servir como líderes.

Un algoritmo MOPSO modificado denominado Another Multi-Objective Particle Swarm Optimization (AMOPSO) se propone en [61], donde sub-enjambres son utilizados para explorar las diferentes regiones del espacio de búsqueda. Cada sub-enjambre tiene su propio grupo de líderes que son seleccionados al azar y sirven como guías hacia la frontera de Pareto. Este enfoque no utiliza registro externo y alivia problemas relacionados con espacios de búsqueda discontinuos.

Otro algoritmo denominado Online Adaptation in Multi-Objective Particle Swarm Optimization (OMOPSO) es presentado en [62], el cual emplea una estimación del vecino más cercano y dos registros externos: en el primero se almacenan las mejores posiciones seleccionadas para la iteración actual de PSO y en el otro se guardan las soluciones no dominadas. También hace uso de turbulencia. Además incorpora un mecanismo para retirar los líderes, cuando su número supera un umbral.

En [63] se propone un algoritmo donde se utilizan varios registros externos, uno para las soluciones globales y uno para cada partícula, donde almacena las soluciones de Pareto óptimas recientemente descubiertas. Hace uso de una ruleta para la selección e introduce en los registros el envejecimiento de los líderes.

Por otra parte, en [64] se desarrolla el Multi-Objective Particle Swarm Optimization Crowding Distance (MOPSO-CD) que incorpora un mecanismo de distancia de hacinamiento para la selección de

la mejor partícula global y la eliminación de las soluciones no dominadas del registro externo. Emplea mutación para mantener la diversidad de las soluciones no dominadas. La distancia de hacinamiento se calcula por separado para cada solución no dominada. Una porción de soluciones no dominadas con las distancias de hacinamiento más altas son seleccionadas al azar para servir como líderes del enjambre.

Un trabajo adicional se puede apreciar en [65] donde se proponen las técnicas *Rounds*, *Random* y *Prob*, basadas en el concepto de dominancia de Pareto para seleccionar a los líderes del registro externo. *Rounds* utiliza como guía global de una partícula la solución no dominada que domina la menor cantidad de partículas del enjambre. *Random* utiliza como guía global de una partícula una solución no dominada probabilísticamente seleccionada, donde cada solución no dominada tiene la misma probabilidad de selección. *Prob* constituye una extensión de *Random* que favorece a los miembros del registro que dominan el menor número de puntos. También se emplea mutación en este algoritmo.

La presentación del algoritmo MOPSO Fitness Sharing (MOPSO-FS) se realiza en [66], la cual es una variante que emplea el intercambio explícito de aptitud donde a cada partícula en el repositorio de soluciones no dominadas se le asigna una aptitud. Este esquema de aptitud compartida asigna valores de aptitud más altos a soluciones con un número pequeño de otras soluciones que la rodean. Los líderes del enjambre son seleccionados a través de una ruleta que utiliza los valores de aptitud asignados.

Un esquema donde cada partícula conserva todas las soluciones no dominadas que se han encontrado se presenta en [67]. En este enfoque se proponen diferentes técnicas, que van desde la selección aleatoria pura hasta el uso de pesos y técnicas de preservación de la diversidad.

La presentación de Smart Multi-Objective Particle Swarm Optimization (SMOPSO) se realiza en [68], el cual incorpora una estrategia para la selección de líderes donde se evalúa cada partícula de acuerdo con cada función objetivo por separado asumiendo como media de las mejores partículas la mejor posición global, para la actualización de enjambre para cada función objetivo.

Finalmente en [69] se realiza una revisión sobre optimización, tanto de uno como de varios objetivos, analizando de manera experimental los efectos de la inercia, el coeficiente de aceleración y el mecanismo de selección aleatorio.

## ENFOQUES ADICIONALES DE ALGORITMOS PSO MULTI-OBJETIVO

En esta sección se revisan enfoques adicionales que se han realizado para los algoritmos multi-objetivo basados en enjambres de partículas.

### Enfoque adaptativo

Sobre un primer enfoque en [70] se propone el Adaptive Multi-objetivo Particle Swarm Optimization (AMOPSO). Este algoritmo incorpora la inercia y el coeficiente de aceleración como variables de control con las variables de optimización habituales. También se incorpora un nuevo parámetro de diversidad para asegurar la diversidad suficiente entre las soluciones de la parte del frente de Pareto no dominadas.

Otro trabajo se puede apreciar en [71], donde las partículas no dominadas se almacenan en un repositorio externo que se actualiza continuamente a través de un mecanismo de adaptación. El algoritmo incluye un operador de mutación auto-adaptativa. La estrategia propuesta se encuentra basada en el conocimiento de un frente aproximado de Pareto, lo cual permite calcular el desempeño de las soluciones existentes.

En [72] se propone un algoritmo donde la inercia y los coeficientes de aceleración se modifican de forma dinámica para explorar el espacio de búsqueda de manera más eficiente. La distancia de hacinamiento y el mecanismo de mutación se emplean para mantener la diversidad de soluciones no dominadas.

Adicionalmente en [74] se propone la integración de una estrategia dinámica de población, es decir que se puede controlar de forma dinámica el tamaño de la población. El algoritmo propuesto se denomina dinámica poblacional de varios enjambres MOPSO. La estrategia adaptativa de los archivos locales se diseñan para mejorar la diversidad de cada enjambre.

Finalmente en [75] se proponen tres operadores de mutación incluyendo Gauss caótico, Cauchy y Levy combinados con PSO. Se observa que estos operadores de turbulencia mejoran las capacidades de exploración. Adicionalmente se adopta una estrategia auto-adaptativa para la estimación de parámetros.

### Enfoque híbrido

Sobre enfoques híbridos en [76] se propone el Quantum Evolutionary Algorithm (QEA) el cual es un algoritmo de optimización basado en el concepto de la computación cuántica y PSO aplicado a problemas multi-objetivo.

Por otra parte, en [77] se propone un algoritmo PSO híbrido donde se utiliza un método de suma ponderada basada en recocido simulado para realizar la búsqueda local. El mecanismo de búsqueda local impide la convergencia prematura, por lo tanto, mejora la capacidad de convergencia con el verdadero frente de Pareto. Mientras tanto, el problema de optimización multi-objetivo se convierte en el problema de optimización restringido. Adicionalmente se utiliza una nueva estrategia de selección basada en el principio de dominancia para seleccionar el siguiente enjambre.

## Enfoque multi-enjambre

En relación a enfoques que emplean varios grupos de individuos, en [73] se propone un algoritmo con múltiples enjambres llamado Distance Sorting Multi-Objective Particle Swarm Optimization (DSMOPSO), el número de enjambres se ajusta dinámicamente. En este trabajo se propone una estrategia de enjambre dinámico para asignar un número apropiado de enjambres, según sea necesario. También se emplea un mecanismo modificado para la actualización PSO con el fin de gestionar mejor la convergencia y la comunicación entre y dentro de los enjambres. Finalmente se propone la compresión del espacio objetivo y la estrategia de expansión de forma progresiva para explotar el espacio objetivo durante las diferentes etapas del proceso de búsqueda.

Adicionalmente en [78] se propone una mejora de la estrategia del crecimiento del enjambre para múltiples enjambres MOPSO. Además se realizó un análisis de sensibilidad para estudiar el impacto de los parámetros de ajuste en su desempeño.

Por su parte en [79], se desarrolla una estrategia PSO de dos fases basada en agrupación. La población inicial se construye de acuerdo a la distribución de las partículas. Las subpoblaciones que representan los grupos de partículas especializados en nichos se identifican de forma dinámica utilizando algoritmos de agrupamiento basados en densidad. La evolución de las partículas se restringe en cada nicho, de la misma forma no se intercambia información entre los diferentes nichos.

Otro trabajo relacionado se puede observar en [80] donde se propone un algoritmo de optimización multi-objetivo con múltiples enjambres; así, se propone que el número de enjambres se ajuste adaptativamente en todo el proceso de búsqueda. La estrategia asigna un número apropiado de enjambres para mejorar la convergencia y la diversidad entre los enjambres. Se incluye un mecanismo de

actualización PSO para gestionar mejor la comunicación dentro de un enjambre y entre enjambres y una compresión del espacio. También se realiza una expansión del espacio de búsqueda para realizar una exploración progresiva de este.

Según [81] la técnica de enjambres PSO como también la optimización por colonia de hormigas Ant Colony Optimization (ACO) han atraído el interés de los investigadores debido a su simplicidad, eficacia y eficiencia en la resolución de problemas de optimización. En [81] proponen que al incrementar el número de individuos del enjambre aumenta la exactitud.

En [82] se expone que los algoritmos evolutivos multi-objetivo tienen dificultad para asignar la aptitud de los individuos porque los diferentes objetivos a menudo entran en conflicto. Para evitar esta dificultad en [82] se propone una técnica co-evolutiva que emplea varias poblaciones para múltiples objetivos. Con este enfoque se proporciona una manera simple y directa para resolver el problema multi-objetivo dejando que cada población corresponda a un solo objetivo. De esta manera, el problema de asignación de aptitud se puede abordar porque la aptitud de los individuos en cada población se puede asignar por el objetivo correspondiente. Así, se propone el algoritmo Coevolutionary Multiswarm PSO (CMPSO) el cual emplea un archivo compartido externo para diferentes poblaciones a fin de intercambiar la información requerida incorporando dos mecanismos. El primero consiste en modificar la ecuación de velocidad utilizando la información encontrada por diferentes poblaciones. El otro mecanismo consiste en el uso de una estrategia de aprendizaje elitista para la actualización de archivo compartido.

Por otra parte, según [83] los problemas de optimización con más de un objetivo, donde al menos un objetivo cambia en el tiempo, son llamados problemas de optimización multi-objetivo dinámicos. Si por lo menos dos objetivos están en conflicto no existe una

única solución y por lo tanto el objetivo de un algoritmo de optimización multi-objetivo dinámico es seguir el conjunto de soluciones óptimas en el tiempo. Uno de los principales problemas al resolver problemas de optimización, es equilibrar la exploración y explotación durante el proceso de búsqueda. En [83] se investiga el desempeño del algoritmo de optimización de enjambre de partículas de vector evaluado dinámico, utilizando un enjambre heterogéneo, donde cada partícula tiene un comportamiento diferente, es decir que se tienen partículas especializadas en exploración y explotación por separado.

### Descomposición de funciones

Sobre esta orientación en [84] se estudia la aplicación de técnicas de MOPSO utilizando métodos de descomposición, donde se propone un algoritmo que integra un enfoque evolutivo multi-objetivo basado en descomposición.

Por su parte en [85] se propone una técnica para el control del área de las soluciones dominantes denominada Control of Dominance Area of Solutions (CDAS). La propuesta se analiza de forma experimental para identificar la influencia en la convergencia y la diversidad a través del análisis de algunos indicadores de calidad y pruebas estadísticas.

### Propuestas para mejorar los algoritmos PSO multi-objetivo

Sobre mejoras del algoritmo MOPSO en [86] se propone un mecanismo de perturbación basada en el método de Taguchi. Se incluye la selección por torneo para las mejores soluciones globales con el propósito de ampliar el espacio de búsqueda y se emplea la perturbación de Taguchi para mejorar la capacidad de búsqueda y reducir la posibilidad de caer en óptimos locales.

Por su parte en [87] se propone un método de intercambio de información proporcionando a las

partículas la capacidad de avanzar hacia el verdadero frente de Pareto.

En [88] se propone un algoritmo que incorpora la técnica de caminata aleatoria para mejorar la búsqueda local de las soluciones no dominadas. También se considera una métrica de hacinamiento para lograr una buena distribución de las partículas sobre el frente de Pareto.

Un trabajo adicional se puede observar en [89] y [90], donde se propone un algoritmo compuesto por dos etapas. En el primer paso se divide la población en grupos y cada grupo lleva a cabo la búsqueda para una función objetivo en particular. En el segundo paso, se realiza la búsqueda para adquirir la diversidad de soluciones de Pareto empleando el mejor punto global basado en las soluciones adquiridas en el primer paso.

En [91] se presenta una mejora del algoritmo de optimización de enjambre de partículas multi-objetivo basado en un archivo de hipercubo dinámico. El algoritmo propuesto considera una modificación del método de archivo hipercubo propuesto originalmente en 2002 por Coello y Lechuga, y cambia los límites del espacio objetivo dinámicamente en el proceso de optimización. Cuando las partículas quedan atrapadas en los frentes locales de Pareto, el algoritmo introduce un proceso de mutación con el fin de ayudar a las partículas a saltar.

Según [92] para la optimización multi-objetivo basada en enjambres de partículas se tienen aspectos de importancia. El primer aspecto a considerar es el método de selección para la mejor posición individual y global. El segundo consiste en utilizar un archivo para conservar buenas posiciones del conjunto óptimo de Pareto. Considerando estos aspectos en [92] se presenta un método para seleccionar la mejor posición individual basada en el desempeño de una de las funciones objetivo. Para la selección de la mejor posición global se emplean las posiciones almacenadas previamente.

En [93] se analizan diferentes esquemas para la selección del líder. En este trabajo se muestra una variante que utiliza el indicador de hipervolumen para orientar la selección líder. Basándose en los resultados se concluye que el uso del hipervolumen para la selección del líder es un enfoque prometedor en los algoritmos PSO multi-objetivos.

Por otra parte en [94] se realiza una propuesta buscando mejorar tanto la explotación y la capacidad de exploración de MOPSO basado en la combinación de un transposón y los enfoques de siembra elitistas. Un transposón o elemento genético transponible es una secuencia de ADN que puede moverse de manera autosuficiente a diferentes partes del genoma de una célula, un fenómeno conocido como transposición.

En [95] se propone un método de máxima puntuación para identificar los mejores individuos con el fin de guiar el proceso de búsqueda sin la necesidad de utilizar la dominancia de Pareto. Con este enfoque se mantiene un archivo con la máxima capacidad. En cada iteración, el archivo se actualiza con la inclusión de las soluciones no dominadas de la población combinada y el archivo. Si el tamaño del archivo supera la capacidad máxima, se corta según el *ranking* máximo.

Por otra parte, en [96] se propone un algoritmo PSO basado en un conjunto de operadores de evolución. El método para la evolución de los conjuntos de soluciones PSO se administra junto con la selección de las mejores partículas globales y locales.

En [97] se propone el cálculo de la distancia de la raíz cuadrada, (Square Root Distance SRD) entre las partículas y los líderes para el proceso de selección. Este nuevo criterio puede hacer que todos los enjambres exploren el frente de Pareto de manera más uniforme. La segunda propuesta consiste en el procedimiento para actualizar los miembros del archivo. Cuando el archivo externo está lleno y un nuevo miembro se va a agregar, un miembro del

archivo con el valor SRD pequeño entre sus vecinos se elimina. Con esta disposición, las soluciones no dominadas pueden ser bien distribuidas.

Según [98] debido a su rápida convergencia la optimización de enjambre de partículas incurre en una rápida pérdida de la diversidad durante el proceso evolutivo, lo cual no permite tener una aproximación precisa y bien distribuida del verdadero frente de Pareto. Considerando lo anterior en [98] se propone evaluar el entorno evolutivo incluyendo la densidad, el rango y los indicadores de diversidad basados en las mediciones de distancia de grupos, el potencial y la entropía de distribución, respectivamente. Con este enfoque se busca ajustar dinámicamente el equilibrio en la exploración y explotación de acuerdo con la información de realimentación desde el entorno evolutivo.

## DISCUSIÓN

Con la revisión realizada se puede apreciar que existen diferentes enfoques y propuestas tanto de algoritmos evolutivos como de algoritmos basados en enjambres de partículas para optimización multi-objetivo. Al respecto es de resaltar que se mantiene la investigación en este campo, siendo una alternativa la propuesta de una Estrategia de Optimización multi-objetivo (EOM) con el control de la diversidad (CD) basada en la emulación de la propiedad emergente *movimiento exitoso* de un enjambre que persigue una meta y la consigue de manera colaborativa.

## CONCLUSIONES

Con esta revisión es posible identificar que el enfoque más empleado para el desarrollo de algoritmos de optimización multi-objetivo bio-inspirados se encuentra basado en la dominancia de las soluciones encontradas.

Considerando los resultados reportados en los artículos revisados se observa que los algoritmos

genéticos multi-objetivo presentan una muy buena capacidad de exploración, sin embargo su tiempo de convergencia es lento.

Por su parte los algoritmos de optimización multi-objetivo basados en enjambres de partículas presentan una buena tasa de convergencia, a pesar de ello su capacidad de exploración se encuentra comprometida.

Es de apreciar que un factor de importancia en el momento de implementar un algoritmo de optimización multi-objetivo, basado en el frente de Pareto, consiste en la forma como se actualiza la lista de soluciones no dominadas.

Se espera emplear los conceptos identificados en esta revisión para el desarrollo de un algoritmo de optimización multi-objetivo basado en el movimiento colectivo de un enjambre de partículas, de tal forma que se logre una buena exploración del frente óptimo de Pareto.

## REFERENCIAS

- [1] C. Coello, D. Van Veldhuizen, G. Lamont, *Evolutionary Algorithms for Solving Multi-Objective Problems*, Springer, Second Edition, 2007.
- [2] C. Coello., M. Salazar, *MOPSO: A proposal for multiple objective particle swarm optimization*, In Proceedings of the IEEE Congress of Evolutionary Computation, 2002.
- [3] D. Van Veldhuizen, G. Lamont, "Multiobjective Evolutionary Algorithms: Analyzing the State-of-the-Art", *Evolutionary Computation*, vol. 8, no. 2, pp. 125-147, 2000.
- [4] J. Schaffer, *Multiple Objective Optimization with Vector Evaluated Genetic Algorithms*, Proceedings of the First International Conference on Genetic Algorithms and Their Applications, pp. 93-100, 1985.
- [5] D. Van Veldhuizen, *Multiobjective Evolutionary Algorithms: Classifications, Analyses, and New Innovations*, Ph.D. thesis, Air Force Institute of Technology, Wright - Patterson AFB, Ohio, 1999.
- [6] E. Zitzler, L. Thiele, *An evolutionary algorithm for multiobjective optimization: The strength Pareto approach*, Technical report 43, Computer engineering and Networks Laboratory (TIK), Swiss Federal Institute of Technology (ETH) Zurich, 1999.
- [7] E. Zitzler, L. Thiele, "Multiobjective evolutionary algorithms: A comparative case study and the strength Pareto approach", *IEEE Transaction on Evolutionary Computation*, vol 3, no. 4, pp. 257-271, 1999.
- [8] J. Knowles, D. Corne, *The Pareto archived evolution strategy: A new baseline algorithm for Pareto multiobjective optimization*, IEEE Congress on Evolutionary Computation (CEC), 1999.
- [9] D. Corne, J. Knowles, M. Oates, "The Pareto envelope - based selection algorithm for multiobjective optimization", *Parallel Problem Solving from Nature - PPSN VI*, pp. 839-848, 2000.
- [10] K. Deb, S. Agrawal, A. Pratap, T. Meyarivan, "A fast elitist non - dominated sorting genetic algorithm for multi-objective optimization: NSGA II", *Parallel Problem Solving From Nature - PPSN VI*, pp. 849-858, 2000.
- [11] E. Zitzler, M. Laumanns, L. Thiele, *SPEA 2: Improving the Strength Pareto Evolutionary algorithm*, Technical report 103, Computer engineering and Networks Laboratory (TIK), Swiss Federal Institute of Technology (ETH) Zurich, 2001.
- [12] D. Dumitrescu, C. Grosan, M. Oltean, *Simple Multiobjective, Evolutionary Algorithm*, *Seminars on Computer Science*, Faculty of Mathematics and Computer Science, Babe-Bolyai University of Cluj-Napoca, pp. 3-12, 2001.
- [13] D. Dumitrescu, C. Grosan, M. Oltean, "A new evolutionary adaptive representation paradigm", *Studia Universitatis Babes-Bolyai, Seria Informatica*, vol. XLVI, no. 1, pp. 15-30, 2001.
- [14] D. Goldberg, *Genetic Algorithms in Search, Optimization and Machine Learning*, Addison-Wesley Publishing Co., Reading, Massachusetts, 1989.

- [15] N. Srinivas, K. Deb, "Multiobjective optimization using nondominated sorting in genetic algorithms", *Journal of Evolutionary Computation*, vol. 2, no. 3, pp. 221-248, 1994.
- [16] J. Horn, N. Nafpliotis, D. Goldberg, *A niched Pareto genetic algorithm for multiobjective optimization*, IEEE World Congress on Computational Intelligence, IEEE Conference on Evolutionary Computation, 1994.
- [17] M. Erickson, A. Mayer, J. Horn, "The niched Pareto genetic algorithm 2 applied to the design of groundwater remediation systems", *Evolutionary Multi-Criterion Optimization, Lecture Notes in Computer Science*, vol. 1993, pp. 681-695, 2001.
- [18] E. Zitzler, M. Laumanns, L. Thiele, "SPEA 2: Improving the Strength Pareto Evolutionary Algorithm", *CIMNE, Evolutionary Methods for Design, Optimisation, and Control*, pp. 95-100, 2002.
- [19] D. Cvetkovic, I. Parmee, "Preferences and their application in evolutionary multiobjective optimization", *IEEE Transactions on Evolutionary Computation*, vol. 6, no. 1, pp. 42-57, 2002.
- [20] A. Lara, G. Sanchez, C. Coello, O. Schutze, "HCS: A New Local Search Strategy for Memetic Multiobjective Evolutionary Algorithms", *IEEE Transactions on Evolutionary Computation*, vol. 14, no. 1, 2010.
- [21] Y. Junchi, L. Guoqiang, *Double space based multiobjective evolutionary algorithm*, International Conference on Machine Learning and Cybernetics (ICMLC), 2012.
- [22] P. Bosman, "On Gradients and Hybrid Evolutionary Algorithms for Real-Valued Multiobjective Optimization", *IEEE Transactions on Evolutionary Computation*, vol. 16, no. 1, 2012.
- [23] D. Brockhoff, E. Zitzler, *Improving hypervolume-based multiobjective evolutionary algorithms by using objective reduction methods*, IEEE Congress on Evolutionary Computation (CEC), 2007.
- [24] I. Karahan, M. Köksalan, "A Territory Defining Multiobjective Evolutionary Algorithms and Preference Incorporation", *IEEE Transactions on Evolutionary Computation*, vol. 14, no. 4, 2010.
- [25] Z. Qingfu, L. Hui, "MOEA/D: A Multiobjective Evolutionary Algorithm Based on Decomposition", *IEEE Transactions on Evolutionary Computation*, vol. 11, no. 6, 2007.
- [26] L. Chi-Ho, K. Ye-Hoon, K. Jong-Hwan, *Multiobjective evolutionary algorithm reinforcing specific objective*, IEEE Congress on Evolutionary Computation (CEC), 2008.
- [27] L. Hai-lin, L. Xueqiang, *The multiobjective evolutionary algorithm based on determined weight and sub-regional search*, IEEE Congress on Evolutionary Computation (CEC), 2009.
- [28] G. Montemayor-Garcia, G. Toscano-Pulido, *A study of surrogate models for their use in multiobjective evolutionary algorithms*, 8th International Conference on Electrical Engineering Computing Science and Automatic Control (CCE), 2011.
- [29] Z. Aimin, Z. Qingfu, Z. Guixu, *A multiobjective evolutionary algorithm based on decomposition and probability model*, IEEE Congress on Evolutionary Computation (CEC), 2012.
- [30] L. Hai-Lin, W. Dan, *A constrained multiobjective evolutionary algorithm based decomposition and temporary register*, IEEE Congress on Evolutionary Computation (CEC), 2013.
- [31] W. Chen, Y. Gong, Z. Zhan, J. Zhang, Y. Li, Y. Tan, "An Evolutionary Algorithm with Double-Level Archives for Multiobjective Optimization", *IEEE Transactions on Cybernetics*, vol. PP, no. 99, 2014.
- [32] W. Rui, R. Purshouse, P. Fleming, "Preference-Inspired Coevolutionary Algorithms for Many-Objective Optimization", *IEEE Transactions on Evolutionary Computation*, vol. 17, no. 4, pp. 474-494, 2013.
- [33] Q. Rongbin, D. Wenli, W. Zhenlei, Q. Feng, *Multiobjective evolutionary algorithm based on the Pareto Archive and individual migration*, 7th World Congress on Intelligent Control and Automation (WCICA), 2008.

- [34] J. Fieldsend, R. Everson, S. Singh, "Using unconstrained elite archives for multiobjective optimization", *IEEE Transactions on Evolutionary Computation*, vol. 7, no. 3, pp. 305-323, 2003.
- [35] H. Shinn-Ying, S. Li-Sun, C. Jian-Hung, "Intelligent evolutionary algorithms for large parameter optimization problems", *IEEE Transactions on Evolutionary Computation*, vol. 8, no. 6, pp. 522-541, 2004.
- [36] H. Sato, H. Aguirre, K. Tanaka, *Local dominance using polar coordinates to enhance multiobjective evolutionary algorithms*, Congress on Evolutionary Computation, 2004.
- [37] H. Sato, H. Aguirre, K. Tanaka, *On the locality of dominance and recombination in multiobjective evolutionary algorithms*, IEEE Congress on Evolutionary Computation, 2005.
- [38] W. Yong, C. Zixing, *A constrained optimization evolutionary algorithm based on multiobjective optimization techniques*, IEEE Congress on Evolutionary Computation, 2005.
- [39] M. Sato, H. Aguirre, K. Tanaka, *Effects of d-Similar Elimination and Controlled Elitism in the NSGA-II Multiobjective Evolutionary Algorithm*, IEEE Congress on Evolutionary Computation (CEC), 2006.
- [40] C. Zixing, W. Yong, "A Multiobjective Optimization-Based Evolutionary Algorithm for Constrained Optimization", *IEEE Transactions on Evolutionary Computation*, vol. 10, no. 6, 2006.
- [41] L. Rachmawati, D. Srinivasan, "Multiobjective Evolutionary Algorithm With Controllable Focus on the Knees of the Pareto Front", *IEEE Transactions on Evolutionary Computation*, vol. 13, no. 4, 2009.
- [42] Y. Song, J. Ji, Y. Wang, C. Liu, *A New Evolutionary Algorithm for Solving Multiobjective Optimization*, Fifth International Conference on Natural Computation (ICNC), 2009.
- [43] D. Phan, J. Suzuki, *Boosting Indicator-Based Selection Operators for Evolutionary Multiobjective Optimization Algorithms*, 23rd IEEE International Conference on Tools with Artificial Intelligence (ICTAI), 2011.
- [44] C. Chi, Y. Shiu, "A Multiobjective Evolutionary Algorithm That Diversifies Population by Its Density", *IEEE Transactions on Evolutionary Computation*, vol. 16, no. 2, 2012.
- [45] L. Ke, A. Fialho, S. Kwong, Z. Qingfu, "Adaptive Operator Selection With Bandits for a Multiobjective Evolutionary Algorithm Based on Decomposition", *IEEE Transactions on Evolutionary Computation*, vol. 18, no. 1, 2014.
- [46] K. Parsopoulos, M. Vrahatis, *Multi-Objective Particles Swarm Optimization Approaches*, IGI Global, 2008.
- [47] K. Parsopoulos, M. Vrahatis, *Particle swarm optimization method in multiobjective problems*, In Proceedings of the ACM Symposium on Applied Computing, 2002.
- [48] U. Baumgartner, C. Magele, W. Renhart, "Pareto optimality and particle swarm optimization", *IEEE Transactions on Magnetics*, vol. 40, no. 2, pp. 1172-1175, 2004.
- [49] M. Mahfouf, M. Chen, D. Linkens, "Adaptive weighted particle swarm optimisation for multi-objective optimal design of alloy steels", *Lecture notes in computer science*, vol. 3242, pp. 762-771, 2004.
- [50] X. Li, "A non-dominated sorting particle swarm optimizer for multi-objective optimization", *Lecture notes in computer science*, vol. 2723, pp. 37-48 2003.
- [51] X. Hu, R. Eberhart, *Multi-objective optimization using dynamic neighborhood particle swarm optimization*, IEEE Congress Evolutionary Computation, 2002.
- [52] X. Hu, R. Eberhart, Y. Shi, *Particle swarm with extended memory for multi-objective optimization*, IEEE Swarm Intelligence Symposium, 2003.
- [53] K. Parsopoulos, M. Vrahatis, *On the computation of all global minimizers through particle swarm optimization*, IEEE Transactions on Evolutionary Computation, 2004.

- [54] C. Chow, H. Tsui, *Autonomous agent response learning by a multi-species particle swarm optimization*, IEEE Congress on Evolutionary Computation, 2004.
- [55] J. Fieldsend, S. Singh, *A multiobjective algorithm based upon particle swarm optimisation, An efficient data structure and turbulence*, In Proceedings of the UK Workshop on Computational Intelligence, 2002.
- [56] T. Ray, K. Liew, "A swarm metaphor for multi-objective design optimization", *Engineering Optimization*, vol. 34. no. 2, pp. 141-153, 2002.
- [57] T. Bartz-Beielstein, P. Limbourg, J. Mehnen, K. Schmitt, K. Parsopoulos, M. Vrahatis, *Particle swarm optimizers for Pareto optimization with enhanced archiving techniques*, IEEE Congress on Evolutionary Computation, 2003.
- [58] D. Srinivasan, T. Seow, *Particle swarm inspired evolutionary algorithm (PSEA) for multi-objective optimization problem*, IEEE Congress on Evolutionary Computation, 2003.
- [59] S. Mostaghim, J. Teich, *Strategies for finding good local guides in multi-objective particle swarm optimization (MOPSO)*, IEEE Swarm Intelligence Symposium, 2003.
- [60] X. Li, "Better spread and convergence: Particle swarm multi-objective optimization using the maximin fitness function", *Lecture notes in computer science*, vol. 3102, pp. 117-128, 2004.
- [61] G. Toscano, C. Coello, "Using clustering techniques to improve the performance of a particle swarm optimizer", *Lecture notes in computer science*, vol. 3102, pp. 225-237, 2004.
- [62] M. Reyes-Sierra, C. Coello, *Online adaptation in multi-objective particle swarm optimization*, IEEE Swarm Intelligence Symposium, 2006.
- [63] S. Ho, S. Yang, G. Ni, E. Lo, H. Wong, "A particle swarm optimization-based method for multi-objective design optimizations", *IEEE Transactions on Magnetics*, vol. 41, no. 5, pp. 1756-1759, 2005.
- [64] C. Raquel, P. Naval, *An effective use of crowding distance in multi-objective particle swarm optimization*, In Proceedings of the GECCO, 2005.
- [65] J. Alvarez-Benitez, R. Everson, J. Fieldsend, "A MOPSO algorithm based exclusively on Pareto dominance concepts", *Lecture notes in computer science*, vol. 3410, pp. 459-473, 2005.
- [66] M. Salazar, J. Rowe, *Particle swarm optimization and fitness sharing to solve multi-objective optimization problems*, IEEE Congress on Evolutionary Computation, 2005.
- [67] S. Mostaghim, J. Teich, "About selecting the personal best in multi-objective particle swarm optimization", *Lecture notes in computer science*, vol. 4193, pp. 523-532, 2006.
- [68] X. Huo, L. Shen, H. Zhu, "A smart particle swarm optimization algorithm for multiobjective problems", *Lecture notes in computer science*, vol. 4115, pp. 72-80, 2006.
- [69] M. Reyes-Sierra M., C. Coello, *Online adaptation in multi-objective particle swarm optimization*, IEEE Swarm Intelligence Symposium, 2006.
- [70] P. Tripathi, S. Bandyopadhyay, S. Pal, *Adaptive multi-objective particle swarm optimization algorithm*, IEEE Congress on Evolutionary Computation (CEC), 2007.
- [71] S. Agrawal, Y. Dashora, M. Tiwari, S. Young-Jun, "Interactive Particle Swarm: A Pareto-Adaptive Metaheuristic to Multiobjective Optimization", *IEEE Transactions on Systems, Man and Cybernetics, Part A: Systems and Humans*, vol. 38, no. 2, pp. 258-277, 2008.
- [72] W. Hui, Q. Feng, *Improved PSO-based Multi-Objective Optimization using inertia weight and acceleration coefficients dynamic changing, crowding and mutation*, 7th World Congress on Intelligent Control and Automation (WCICA), 2008.
- [73] L. Wen-Fung, G. Yen, *Dynamic swarms in PSO-based multiobjective optimization*, IEEE Congress on Evolutionary Computation (CEC), 2007.

- [74] L. Wen-Fung, G. Yen, "PSO-Based Multiobjective Optimization With Dynamic Population Size and Adaptive Local Archives", *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics*, vol. 38, no. 5, pp. 1270-1293, 2008.
- [75] C. Jinyin, Y. Dongyong, *Constrained handling in multi-objective optimization based on Quantum-behaved particle swarm optimization*, Sixth International Conference on Natural Computation (ICNC), vol. 8, 2010.
- [76] M. Hossain, M. Hossain, M. Hashem, M. Ali, *Quantum Evolutionary Algorithm based on Particle Swarm theory in multiobjective problems*, 13th International Conference on Computer and Information Technology (ICCIIT), 2010.
- [77] W. Jingxuan, W. Yuping, *A New Model Based Hybrid Particle Swarm Algorithm for Multi-objective Optimization*, Third International Conference on Natural Computation (ICNC), vol. 3, 2007.
- [78] W. Leong, G. Yen, *Impact of tuning parameters on dynamic swarms in PSO-based multiobjective optimization*, IEEE Congress on Evolutionary Computation (CEC) (IEEE World Congress on Computational Intelligence), 2008.
- [79] G. Haichang, Z. Weizhou, *Multiobjective Optimization Using Clustering Based Two Phase PSO*, Fourth International Conference on Natural Computation (ICNC), vol. 6, 2008.
- [80] G. Yen, L. Wen, "Dynamic Multiple Swarms in Multiobjective Particle Swarm Optimization", *IEEE Transactions on Systems, Man and Cybernetics, Part A: Systems and Humans*, vol. 39, no. 4, pp. 890-911, 2009.
- [81] W. Elloumi, A. Alimi, *A more efficient MOPSO for optimization*, IEEE/ACS International Conference on Computer Systems and Applications (AICCSA), 2010.
- [82] Z. Zhi-Hui, L. Jingjing, C. Jiannong, Z. Jun, "Multiple Populations for Multiple Objectives: A Coevolutionary Technique for Solving Multiobjective Optimization Problems", *IEEE Transactions on Cybernetics*, vol. 43, no. 2, pp. 445-463, 2013.
- [83] M. Helbig, A. Engelbrecht, *Heterogeneous dynamic vector evaluated particle swarm optimization for dynamic multi-objective optimisation*, IEEE Congress on Evolutionary Computation (CEC), 2014.
- [84] P. Wei, Z. Qingfu, *A decomposition-based multi-objective Particle Swarm Optimization algorithm for continuous optimization problems*, IEEE International Conference on Granular Computing (GrC), 2008.
- [85] A. De Carvalho, A. Pozo, *Analyzing the control of dominance area of solutions in particle swarm optimization for many-objective*, 10th International Conference on Hybrid Intelligent Systems (HIS), 2010.
- [86] L. Chi-Nien, H. Chih-Li, L. Shu-Yan, Yu. Yu-Hsiang, *Taguchi-based disturbance with tournament selection to improve on MOPSO*, IEEE Congress on Evolutionary Computation (CEC), 2011.
- [87] Y. Jintao, Y. Bo, Z. Mingwu, K. Yuyan, *Multiobjective Particle Swarm Optimization with Predatory Escaping Behavior*, 3rd International Workshop on Intelligent Systems and Applications (ISA), 2011.
- [88] O. Soliman, S. Mohamed, E. Ramadan, *A Bio-Inspired Memetic Particle Swarm Optimization Algorithm for Multi-objective Optimization Problems*, Third International Conference on Innovations in Bio-Inspired Computing and Applications (IBICA), 2012.
- [89] H. Hirano, T. Yoshikawa, *A study on two-step search using global-best in PSO for Multi-Objective Optimization Problems*, 6th International Conference on Soft Computing and Intelligent Systems (SCIS) and 13th International Symposium on Advanced Intelligent Systems (ISIS), 2012.
- [90] H. Hirano, T. Yoshikawa, *A study on two-step search based on PSO to improve convergence and diversity for Many-Objective Optimization Problems*, IEEE Congress on Evolutionary Computation (CEC), 2013.
- [91] Z. Guangrui, M. Mahfouf, G. Panoutsos, W. Shen, *A multi-objective particle swarm*

- optimization algorithm with a dynamic hypercube archive, mutation and population competition*, IEEE Congress on Evolutionary Computation (CEC), 2012.
- [92] T. Uchitane, T. Hatanaka, *Experimental study for multi-objective PSO with single objective guide selection*, IEEE Congress on Evolutionary Computation (CEC), 2012.
- [93] A. Nebro, J. Durillo, C. Coello, *Analysis of leader selection strategies in a multi-objective Particle Swarm Optimizer*, IEEE Congress on Evolutionary Computation (CEC), 2013.
- [94] Y. Zhenlun, A. Wu, M. Huaqing, *A Multi-objective PSO algorithm with transposon and elitist seeding approaches*, Sixth International Conference on Advanced Computational Intelligence (ICACI), 2013.
- [95] G. Ying, P. Lingxi, L. Fufang, L. Miao, *Multi-objective cloud estimation of distribution particle swarm optimizer using maximum ranking*, 10th International Conference on Natural Computation (ICNC), 2014.
- [96] S. Xiaoyan, X. Ruidong, Z. Yong, G. Dunwei, *Sets evolution-based particle swarm optimization for many-objective problems*, IEEE International Conference on Information and Automation (ICIA), 2014.
- [97] L. Man-Fai, N. Sin-Chun, C. Chi-Chung, A. Lui, *A new strategy for finding good local guides in MOPSO*, IEEE Congress on Evolutionary Computation (CEC), 2014.
- [98] H. Wang, G. Yen, "Adaptive Multiobjective Particle Swarm Optimization Based on Parallel Cell Coordinate System", *IEEE Transactions on Evolutionary Computation*, vol. 19, no. 1, pp. 1-18, 2015.

