

# Técnicas metaheurísticas de optimización multiobjetivo para resolver el problema del portafolio de inversión

Metaheuristic techniques of multiobjective optimization to solve the investment portfolio problem

---

Naim Reyes-Hernández\*

Antonin Ponsich\*\*

Luis Fernando Hoyos-Reyes\*\*\*

(Fecha de recepción: 24 de noviembre de 2017. Fecha de aprobación: 30 de abril de 2018)

## RESUMEN

En este trabajo, se presenta bajo una perspectiva metaheurística el problema multiobjetivo para el portafolio de inversión, que debe minimizar el riesgo, y maximizar el rendimiento esperado. Considerando el modelo de Markowitz se resuelve el problema con dos objetivos describiendo la totalidad del frente de Pareto correspondiente. En este trabajo, se emplean cuatro técnicas metaheurísticas: SPO, NSGA-II, MOEA/D y VEGA y se efectúa un análisis comparativo sobre cinco instancias de investigación en mercados financieros de Hong Kong (Hang Sen31), Alemania (DAX85), Gran Bretaña (FTSE89), Estados Unidos de América (S&P98) y Japón (Nikkei225) empleando una base de datos de marzo de 1992 a septiembre de 1997.

Clasificación JEL: C44, C58, C61, G11

**Palabras clave:** técnicas metaheurísticas, optimización multiobjetivo, portafolio de inversión, frente de Pareto.

---

\* Universidad Autónoma Metropolitana, Unidad Azcapotzalco, División de Ciencias Básicas e Ingeniería.  
naim\_my@hotmail.com

\*\* Universidad Autónoma Metropolitana, Unidad Azcapotzalco, División de Ciencias Básicas e Ingeniería.  
aspo@azc.uam.mx

\*\*\* Universidad Autónoma Metropolitana, Unidad Azcapotzalco, División de Ciencias Básicas e Ingeniería.  
hrhf@azc.uam.mx

## ABSTRACT

*In this paper, the multiobjective problem for the investment portfolio is presented under a metaheuristic perspective, which refers to minimize de risk and maximize the expected return. Considering the Markowitz model, the problem is solved with two objectives describing the whole of the corresponding Pareto front. Four metaheuristic techniques are used: SPO, NSGA-II, MOEA/D Y VEGA and a comparative analysis is carried out on 5 research instances in financial markets of Hong Kong (Hang Sen31), Germany (DAX85), Great Britain (FTSE89), United States of America (S & P98) and Japan (Nikkei225) using a database from March 1992 to September 1997.*

*JEL classification: C44, C58, C61, G11*

**Keywords:** *metaheuristic techniques, multiobjective optimization, investment portfolio, Pareto front.*

## Introducción

El problema del portafolio de inversión fue estudiado y formulado por Markowitz (1952) quien propuso por primera vez un modelo matemático para este tipo de problemas, se trata de definir un portafolio eficiente, es decir, aquel que tiene un mínimo riesgo para un rendimiento dado o, equivalentemente un portafolio con un máximo rendimiento para un nivel de riesgo dado. Sin duda es una abstracción muy simple pero a la vez útil para empezar a entender las variables de interés y, de hecho, es posible utilizar técnicas de programación cuadrática para encontrar soluciones óptimas. Pero el problema se vuelve complejo cuando se integran ciertas prohibiciones, o reglas del mercado real, sobre los objetivos: el problema de optimización multiobjetivo restringido conlleva al uso de técnicas metaheurísticas. Este aspecto constituye el punto de partida y los objetivos del presente proyecto: implementar y estudiar el desempeño de diferentes técnicas metaheurísticas de optimización multiobjetivo, particularmente Algoritmos Evolutivos Multiobjetivo para la solución del problema de Portafolio de Inversión. La hipótesis de investigación correspondiente enuncia que los Algoritmos Evolutivos Multiobjetivo representa candidatos prometedores para obtener un conjunto de carteras eficientes, asociadas a diversos compromisos entre los objetivos de rendimiento y de riesgo del portafolio seleccionado.

El presente artículo se organiza como se describe a continuación: en la siguiente sección se proporciona el marco teórico, relacionando el problema de inversión con el área de optimización multiobjetivo. En la sección 2, se incluye el desarrollo de los métodos para la selección de poblaciones no dominadas, así como la descripción de las características y el modo operativo de las metaheurísticas manejadas. La sección 3 presenta la implementación del ajuste de parámetros, se presentan las métricas que se adoptaron. Finalmente, en la sección 4 se presentan la parte experimental de la implementación de los algoritmos con los parámetros ajustados, así como representaciones gráficas del desempeño que permiten interpretar rápidamente los resultados.

## 1. El problema del portafolio de inversión

Para evaluar la confianza y eficiencia de las técnicas implementadas, existen varios bancos de instancias ya estudiados en la literatura especializada. En este proyecto, se propone trabajar con cinco mercados de capital propuestos en (Beasley 2015), los cuales fueron elegidos debido a que son ampliamente utilizados en la aplicación práctica dentro de la literatura. La información se muestra en la Tabla 1 y fue obtenida a partir de *stocks* de marzo de 1992 a septiembre de 1997.

Para cada instancia se tiene la siguiente información de entrada: rendimiento promedio semanal de cada activo, desviación estándar del rendimiento semanal y matriz de correlación entre activos. Para evaluar las funciones objetivo solo faltan las covarianzas entre activos, las cuales serán calculadas de la siguiente manera:

Tabla 1. Datos utilizados en la investigación (Beasley 2015)

ID	País	Instancia	Tamaño (N)
0	Hong Kong	Hang Seng31	31
1	Alemania	DAX85	85
2	UK	FTSE89	89
3	USA	S&P98	98
4	Japón	Nikkei225	225

$$\sigma_{ij} = \rho_{ij} s_i s_j \quad (1)$$

Donde  $s_i$  es la desviación estándar del activo y  $\rho_{ij}$  es el índice de correlación entre los activos  $i$  y  $j$ .

Además se dispone de los frentes de Pareto reales de todas las instancias, lo cual es una ventaja para poder evaluar el desempeño de las técnicas de optimización probadas más adelante.

En la presente investigación el proceso de selección de activos para un portafolio será trabajado utilizando el modelo matemático propuesto por Markowitz (1952). Éste se plantea originalmente, como un problema de optimización mono-objetivo, en el cual se pretende maximizar el rendimiento esperado, sujeto a un determinado riesgo establecido o equivalentemente minimizar el riesgo para un rendimiento establecido (Robert C. Merton, 1972).

En el marco de la presente investigación, estos valores establecidos serán denotados como metas y modificándolos, puede ser resuelto como un problema multiobjetivo o MOP (Multi Objective Problem).

El modelo de Markowitz (1952) puede formularse como un MOP, donde el vector objetivo está formado por las funciones de rendimiento esperado y riesgo (a menudo calculado como la varianza del rendimiento). De acuerdo a lo anterior, en el problema del portafolio de inversión,  $\mathbf{x} = [x_1, x_2, \dots, x_N]$  representa un conjunto de inversión específico llamado cartera y  $x_i$  es la proporción del capital que se invierte en el activo  $i$ ,  $f_1(\mathbf{x})$ , cuantifica el riesgo de la cartera y  $f_2(\mathbf{x})$  el rendimiento esperado. El modelo correspondiente se formula de la siguiente manera:

$$\text{Minimizar } \left\{ f_1(\mathbf{x}) = \left[ \sum_{i=1}^N \sum_{j=1}^N x_i x_j \sigma_{ij} \right], \quad - f_2(\mathbf{x}) = \left[ \sum_{i=1}^N x_i \mu_i \right] \right\}$$

*sujeto a*

$$\begin{aligned} \sum_{i=1}^N x_i &= 1 \\ x_i &\geq 0, \quad i = 1, \dots, N. \\ 0 &\leq x_i \leq 1, i = 1, 2, \dots, N \end{aligned}$$

Donde  $x_i$  es la proporción del capital que se invertirá en el activo  $i$ ,  $\sigma_{ij}$  es la covarianza entre el activo  $i$  y el  $j$ ,  $\mu_i$ , es el rendimiento esperado del

activo  $i$  y  $N$  es el número de activos disponibles en el mercado. La restricción de igualdad indica que el total del capital disponible debe ser invertido en la cartera.

Dado un MOP de la forma antes presentada, sea  $\Omega \subset \mathbb{R}^n$  el espacio factible, decimos que una solución  $\mathbf{y} \in \Omega$  donde  $\mathbf{y} = [y_1, y_2, \dots, y_n]$  es no dominada si  $\nexists \mathbf{x} \in \Omega, f_i(\mathbf{x}) \leq f_i(\mathbf{y}) \forall i \in \{1, 2, \dots, m\}$ , y al menos para alguna  $j$   $f_j(\mathbf{x}) < f_j(\mathbf{y})$ .

Algunos autores llaman tales soluciones óptimo de Edgeworth-Pareto, pero es más común denominarla simplemente óptimo de Pareto, o soluciones Pareto-óptimas (Hernández-Lerma y Hoyos-Reyes, 2011). El conjunto  $P$  es conocido como el conjunto de Pareto y es el que se suele presentar como la solución al MOP. Su imagen en el espacio de los objetivos es el frente de Pareto real (PF o PFtrue por sus siglas en inglés).

La presente investigación se concentrará en métodos para problemas continuos no lineales, los cuales, en el estado del arte, también son referidos como de optimización difícil, debido a que no existe algoritmo eficiente o método que pueda garantizar óptimos globales (Ponsich *et al.*, 2013) de manera exacta para tales problemas. En estos casos, se recurre frecuentemente al uso de técnicas metaheurísticas para optimización global. Una clase de estas metaheurísticas son los Algoritmos Evolutivos.

## 2. Técnicas metaheurísticas en optimización multiobjetivo

En esta sección se abordará una de las primeras propuestas para afrontar un problema de optimización multiobjetivo. Se trata de una manera sencilla de plantear el problema desde la perspectiva monobjetivo e implementar una técnica clásica para resolverlo, en este caso un algoritmo genético (GA). Esta técnica constituye un buen punto de partida y de referencia para estudiar los alcances de otros métodos, médula de la presente investigación, los cuales fueron creados específicamente para atacar los problemas de interés en este proyecto.

### 2.1 La noción de función de agregación

Una función de agregación, suma lineal de pesos, combinación convexa o suma ponderada de objetivos (SPO), está definida por Zhang y Li (2007) de la siguiente manera:

Dado un vector de pesos  $\lambda = (\lambda_1, \dots, \lambda_m)^T$ , con  $\lambda_i \geq 0$  para toda  $i = 1, \dots, m$  y  $\sum_{i=1}^m \lambda_i = 1$ , se puede reformular el problema de optimización multiobjetivo como un problema de optimización escalar:

$$\begin{aligned} &\text{minimizar } g^{ws}(x|\lambda) = \sum_{i=1}^m \lambda_i f_i(x) \\ &\text{sujeto a } x \in \Omega \end{aligned}$$

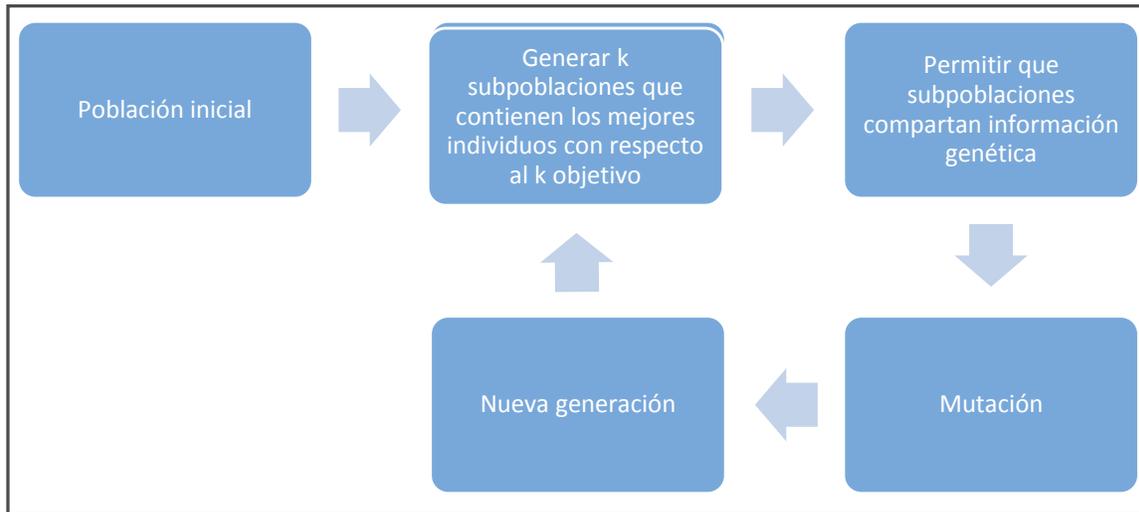
Donde  $\Omega$  es el espacio factible, y  $x$  es el vector de variables. De esta forma, se está reduciendo el problema original a un problema de un sólo objetivo. Se obtiene un conjunto de diferentes vectores óptimos de Pareto al resolver el problema escalar varias veces, utilizando múltiples vectores de pesos. Esta técnica sencilla tiene, sin embargo, desventajas importantes, entre las cuales las más relevantes son:

- i) Para cada solución no dominada deseada, es necesario resolver un problema de optimización escalar, por lo que la aproximación del frente de Pareto aproximado requiere de múltiples ejecuciones del método de solución.
- ii) Debido a la reformulación del problema, no se pueden alcanzar regiones no convexas del frente de Pareto.
- iii) La selección adecuada de los vectores de pesos a utilizar es problemática, ya que, según la naturaleza (lineal o no) del problema, una distribución uniforme de dichos vectores no garantiza que las soluciones obtenidas se distribuyan uniformemente a lo largo del frente.

## 2.2 La estrategia VEGA

Una de las primeras propuestas hechas específicamente para MOPs es la estrategia Algoritmo Genético Evaluado por Vector-Objetivo (o VEGA, por las siglas en inglés de Vector Evaluated Genetic Algorithm), la cual fue presentada por Schaffer (1985). Su funcionamiento está basado en el paradigma de los algoritmos genéticos como su nombre lo indica, pero la característica principal yace en la parte más polémica en el entorno multiobjetivo: el modo en el que se le asigna aptitud a un individuo. Como se puede observar en la Figura 1 se realiza una fase de clasificación en la que se crean  $k$  subpoblaciones con los mejores individuos con respecto a cada uno

Figura 1. Etapas de un algoritmo basado en la estrategia VEGA



de los objetivos. Por lo tanto, los individuos muy aptos con respecto a más de un objetivo podrían aparecer en más de una subpoblación. Cabe mencionar que todas las tablas y figuras disponibles en este artículo son de elaboración propia.

En la versión del algoritmo implementada en la presente investigación, es acertado destacar los siguientes dos puntos:

- i) Se tiene como meta en un MOP determinar las soluciones no dominadas, en cada generación éstas tendrán la característica de que, por ser buenas en todos los objetivos, aparezcan en múltiples subpoblaciones y por ende tener mayor posibilidad de ser seleccionadas para cruza.
- ii) La etapa de creación de subpoblaciones permite especializar el proceso de optimización para objetivos aislados a continuación cuando las soluciones son combinadas, dado que hay un proceso de selección por medio de torneo binario, dentro de esta subetapa el elitismo por dominancia estará presente, lo cual le permitirá al algoritmo ser más dinámico.

### 2.3 La estrategia Nondominated Sorting Genetic Algorithm II (NSGA-II)

El método Algoritmo Genético basado en Sorteo de Dominancia (o NSGA-II, por las siglas en inglés de Nondominated Sorting Genetic Algorithm)

fue propuesto por Deb, Pratap, Agarwal, y Meyarivan (2002) y tiene una complejidad de  $O(m\Psi^2)$ , donde  $m$  es el número de objetivos y  $\Psi$  es el tamaño de la población. Se trata de una versión más eficiente y elitista del NSGA anteriormente propuesto por Srinivas y Deb (1995).

El NSGA-II es una estrategia en la cual es crucial el concepto de dominancia de Pareto en la evolución de las poblaciones en el tiempo. De hecho, la clasificación de soluciones no dominadas implica un costo computacional relativamente muy alto. A grandes rasgos, las etapas que conforman esta estrategia se muestran en la Figura 2.

### 2.3.1 Asignación de distancia crowding

Esta métrica se usa en la estrategia NSGA-II para decidir, en el caso de que sea necesario, qué individuos del último frente capaz de completar una nueva generación deben ser seleccionados. La Crowding Distance (CD) es una aptitud asignada únicamente para los elementos del último frente que no fue posible añadir a la nueva generación, de acuerdo a la Figura 3,  $f_4$  sería tal frente.

Figura 2. Etapas de un algoritmo basado en la estrategia NSGA-II

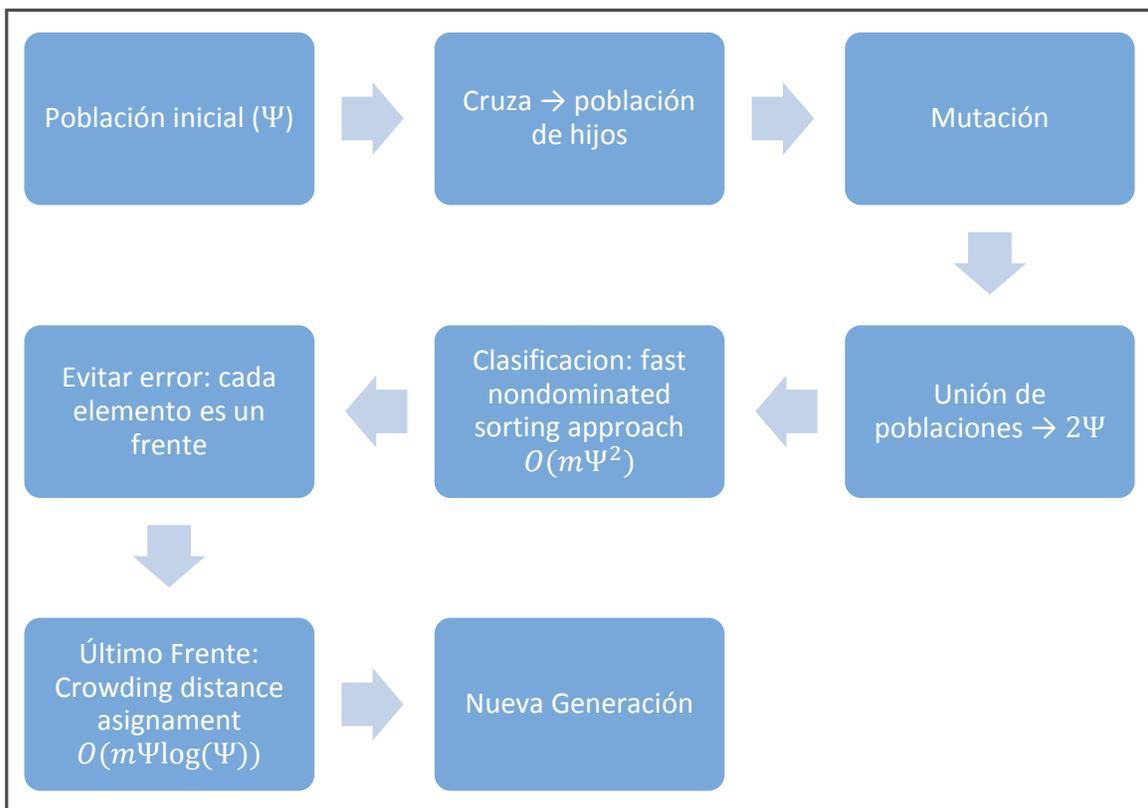
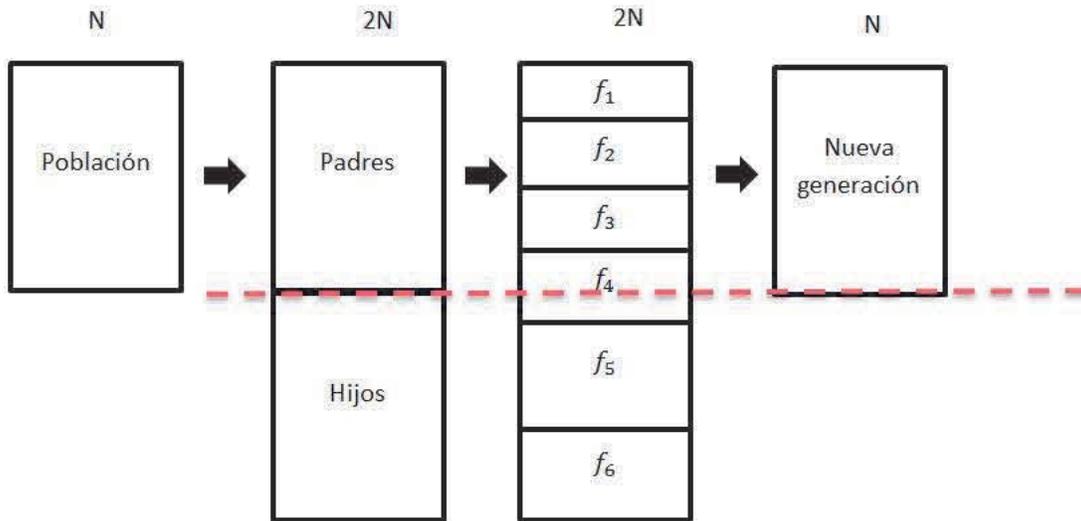


Figura 3. Generación de frentes NSGA-II



Así pues, para el frente  $f_4$  se calcula la métrica utilizando el algoritmo NSGA-II. La CD es proporcional al perímetro del cuboide cuyos vértices están formados por las soluciones vecinas del individuo considerado, en su mismo frente. A medida que el cuboide ocupe un mayor volumen, se le carga un valor más grande a la CD del  $i$ -ésimo individuo, promoviendo su selección ya que está en una región menos densamente poblada. Nótese que, previo al cálculo de CD, es necesario normalizar los objetivos de tal manera que diferencias de magnitud entre sus valores no afecten la validez del procedimiento.

#### 2.4 Multiobjective Evolutionary Algorithm based on Decomposition (MOEA/D)

El Algoritmo Evolutivo Multiobjetivo basado en Descomposición (MOEA/D, por las siglas en inglés de Multi-Objective Evolutionary Algorithm based on Decomposition) es una técnica propuesta por Zhang y Li (2007), basada en una reformulación del MOP en un conjunto de problemas escalares y que se resuelven simultáneamente, de esta forma el proceso de optimización se opera en un entorno monobjetivo. La estrategia explícitamente descompone el MOP en  $Q$  subproblemas de optimización escalar y se resuelve estos subproblemas simultáneamente por medio de la evolución de una población de soluciones. Cada subproblema es parametrizado por un vector de pesos,

que permite orientar la búsqueda hacia cierta región del frente de Pareto. En cada generación, la población se compone de la mejor solución encontrada hasta ese momento para cada subproblema. El MOEA/D proporciona una manera simple pero eficiente de introducir la aproximación por descomposición en el cálculo evolutivo multiobjetivo.

Resolver problemas de optimización escalar en lugar de directamente el problema de optimización multiobjetivo (MOP por sus siglas en inglés) en su conjunto, puede implicar que, cuestiones tales como la asignación de la aptitud de las soluciones (también conocida como *fitness*) y el mantenimiento de la diversidad —que causan dificultades para otros MOEAs no basados en descomposición— podrían llegar a ser más fáciles de manejar en el marco de la técnica de este apartado.

MOEA/D tiene una menor complejidad computacional en cada generación que otras técnicas de optimización multiobjetivo, dado que no implica ningún proceso de clasificación de Pareto: usando una pequeña población es capaz de producir pocas soluciones finales muy uniformemente distribuidas. El ajuste de los vectores de pesos, representa sin embargo, una tarea difícil, particularmente para problemas de 3 o más objetivos.

Las etapas del algoritmo se enseñan en la Figura 4. Se puede apreciar que después de la creación y evaluación de la población inicial, se establecen las relaciones de vecindario que existen entre los diferentes individuos de acuerdo a la cercanía entre los vectores de pesos que parametrizan cada subproblema escalar. Esto permite calcular el punto ideal ( $z$ ) usado posteriormente en la función de escalarización (ver subsección siguiente) e inicializar el archivo externo (EP) reportando las soluciones no-dominadas encontradas. Posteriormente, cada iteración consiste en generar soluciones nuevas gracias a los operadores genéticos, calcular su aptitud y eventualmente integrarlas a EP. Finalmente, este ciclo generacional se repite hasta cumplirse algún criterio de paro, típicamente basado en la utilización de recursos (tiempo o número de evaluaciones de los objetivos).

#### 2.4.1 Aproximación de intersección de frontera (BI)

Existen distintas propuestas para efectuar la descomposición (o escalarización) necesaria para la implementación del MOEA/D. En la presente investigación se ha experimentado particularmente con una de las tres consideradas por (Zhang y Li, 2007), conocida como aproximación de intersección de frontera (BI: Boundary Intersection Approach).

Bajo ciertas condiciones de regularidad, el PF de un MOP (de maximización) continuo es parte de la frontera más alta a la derecha de un objetivo fijado a alcanzar. Geométricamente, estos enfoques BI tienen como objetivo, como su nombre lo indica, encontrar puntos de intersección entre el límite superior y un conjunto de líneas que emanan del punto ideal (denotado como  $z^*$ ). Si estas líneas están distribuidas de manera adecuada, se puede esperar que los puntos de intersección resultantes proporcionen una buena aproximación, uniforme, de la totalidad del PF. Cabe mencionar que este enfoque es capaz de aproximar FP no cóncavos (Jin, Y., Qu, R., y Atkin, J., 2016).

Matemáticamente, se tienen los siguientes subproblemas de optimización escalar:

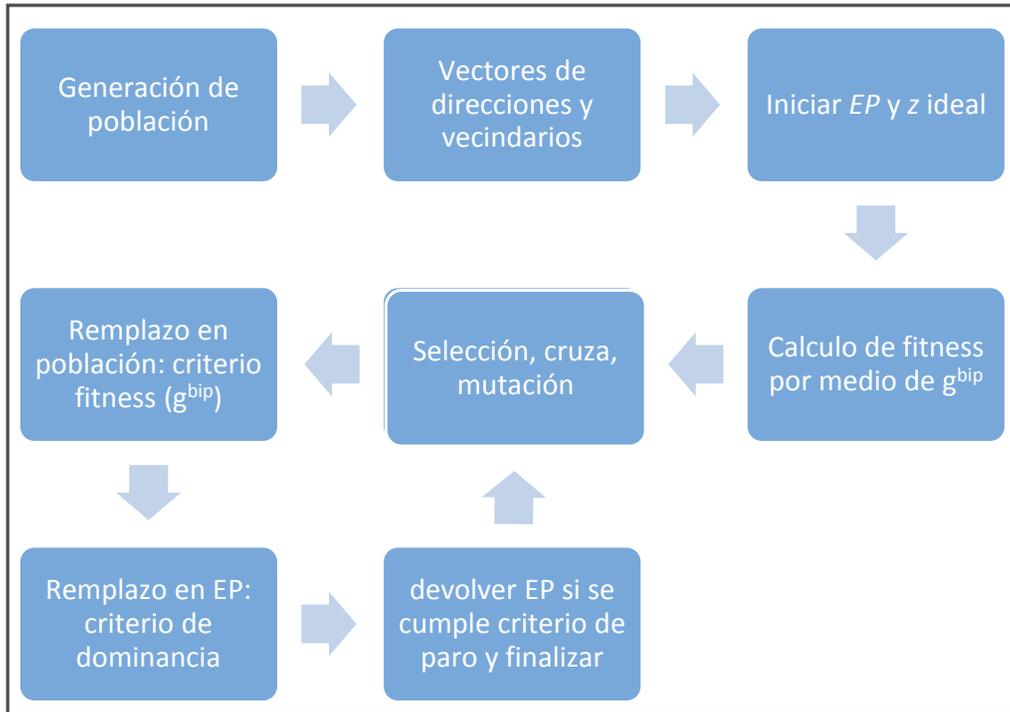
$$\begin{aligned} & \text{minimizar } g^{bip}(x|\lambda, z^*) = d_1 + \theta d_2 \\ & \text{s. a.} \\ & \mathbf{x} \in \Omega \\ & \text{donde} \\ & d_1 = \frac{\| (z^* - F(\mathbf{x}))^T \lambda \|}{\|\lambda\|} \\ & d_2 = \|F(\mathbf{x}) - (z^* - d_1 \lambda)\| \\ & \theta > 0 \text{ parámetro de penalización} \end{aligned}$$

Donde el parámetro  $\theta$  define el desvío ( $d_2$ ) con respecto a la dirección  $\lambda$  ( $\lambda$  es determinada por el vector de pesos parametrizando el subproblema considerado) en la búsqueda del problema escalar actual, contra la velocidad a la que la solución se irá acercando al PF ( $d_1$ ).

Las relaciones de vecindad entre los sub-problemas se definen en función de las distancias entre vectores de pesos. Así pues, las soluciones óptimas a dos subproblemas vecinos deben ser similares. Cada subproblema (es decir, la función BI particular a un vector  $\lambda$ ) se optimiza en MOEA/D mediante el uso de la información sólo de sus subproblemas vecinos.

Es importante mencionar que la estrategia MOEA/D requiere además de un espacio de memoria o archivo de almacenamiento llamado *EP*, en el cuál se administran soluciones no dominadas. Se ha propuesto delimitar el tamaño del archivo por medio de CD (ver Figura 4).

Figura 4. Etapas de un algoritmo basado en la estrategia MOEA/D



### 3. Métricas utilizadas y ajuste de parámetros

Particularmente, en la presente investigación se decidió considerar las siguientes métricas de desempeño, debido a que al tomarlas en cuenta se engloban las características necesarias para considerar una aproximación multiobjetivo de buena calidad.

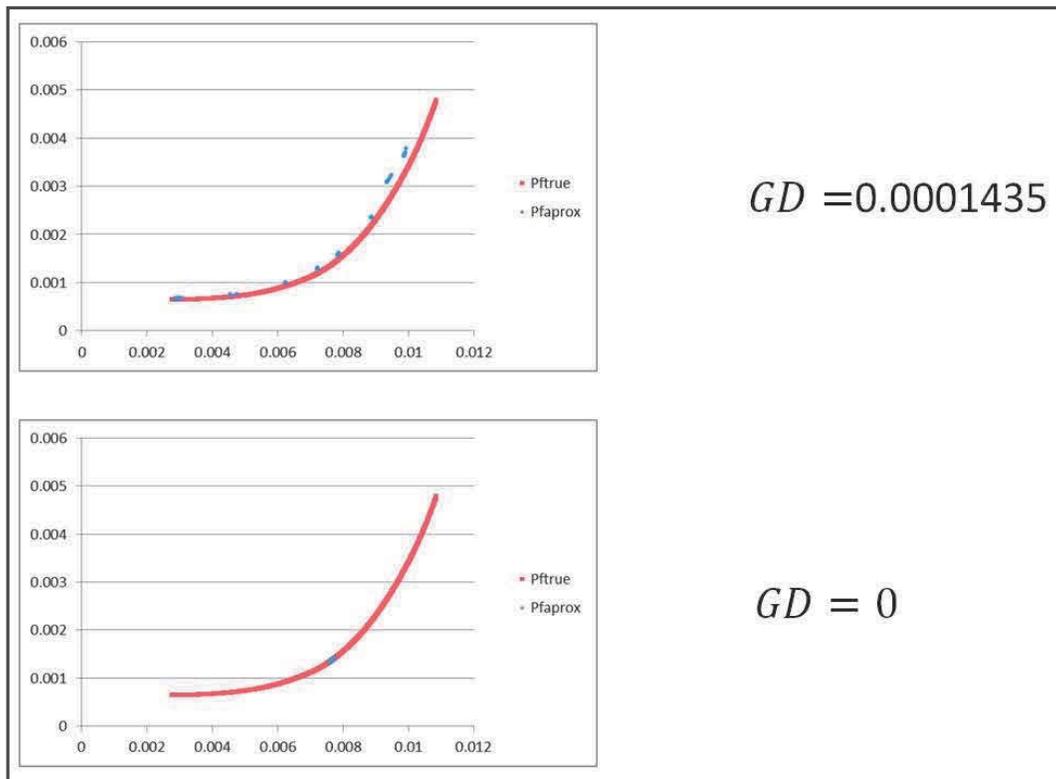
#### 3.1 Distancia generacional

La distancia generacional (*gd* por sus siglas en inglés) se utiliza para saber en promedio qué tan lejos está el frente aproximado (PF<sub>aprox</sub>) del real (PF<sub>true</sub>), por lo cual se trata de una medida de convergencia. Para el cálculo de esta métrica se necesita el frente real, la fórmula es la siguiente:

$$gd = \frac{\sum_{i=1}^{|PF_{aprox}|} d_i}{|PF_{aprox}|} \quad (1)$$

Donde  $d_i$  es la distancia euclidiana (en el espacio de los objetivos) entre la  $i$ -ésima solución del PFaprox y la solución más cercana del PFtrue. Un valor cercano a cero indica buena convergencia. La pregunta que surge es ¿por qué no es suficiente tal métrica para explicar la calidad de un frente aproximado? Al observarla, se muestran dos frentes aproximados, el primero es una buena aproximación y de hecho el valor de  $gd$  es bueno (muy cercano a cero), por otro lado, el segundo frente tiene un mejor valor para la métrica, y sin embargo no es una mejor aproximación ya que contiene pocas soluciones muy cercanas, incluso parte del frente real, las cuales no podrían proporcionar indicios de la tendencia de la curva real por completo (ver Figura 5).

Figura 5. Cálculo de la distancia generacional



### 3.2 Dispersión eficiente de conjuntos

Llamada en inglés *Efficient Set Spacing* o simplemente *Spacing*, describe numéricamente la propagación o dispersión de los vectores de PFaprox, la fórmula para su cálculo se muestra a continuación:

$$ESS = \sqrt{\frac{1}{|PF_{aprox}|} \sum_{i=1}^{|PF_{aprox}|} (d_i - \bar{d})^2} \quad (2)$$

$$\text{donde } \begin{cases} d_i = \min_{j \in PF_{aprox}, i \neq j} \left\{ \sum_{p=1}^k |f_p^i - f_p^j| \right\} \\ \bar{d} = E(d_i) \end{cases}$$

Lo ideal, al igual que  $gd$  es un valor cercano a 0.

### 3.3 Hipervolumen

Es definido como el área dominada por  $PF_{aprox}$ , con respecto al espacio de los objetivos. Para dos objetivos, esto equivale a la sumatoria de todas las áreas rectangulares delimitadas por algún punto de referencia  $R = (f_0(\mathbf{x}), f_1(\mathbf{x}))$ , su cálculo se realiza con:

$$HV = \{ \cup_i vol_i | vec_i \in PF_{aprox} \} \quad (3)$$

Se intenta lograr que este valor sea el mayor posible para concluir que la aproximación es buena. La ventaja del uso del hipervolumen es que es una métrica compatible con la definición de optimalidad de Pareto (Zitzler *et al.*, 2003), es decir que buscar todas las soluciones Pareto óptimas de un MOP es equivalente a maximizar el hipervolumen del frente aproximado correspondiente.

En muchas aplicaciones (incluyendo la presente investigación) la magnitud en la que se mide la función objetivo suele variar en intervalos de reales numéricamente muy pequeños o muy grandes, lo cual dificulta el análisis comparativo y la observación de comportamientos para el hipervolumen. Para un mejor uso de tal métrica se definió una derivada de ésta a la cual se ha nombrado hipervolumen-percent y consiste en el porcentaje o proporción del hipervolumen del frente real que ocupa el hipervolumen aproximado usando el mismo punto de referencia para ambos frentes debido a que  $F(\mathbf{x}) = (f_0(\mathbf{x}), f_1(\mathbf{x})) = (\text{retorno esperado}, \text{riesgo})$  tiene un frente de Pareto

to totalmente convexo, el punto de referencia son los valores extremos del mismo, es decir  $R = (\max\{\text{retorno esperado}\}, \min\{\text{riesgo}\})$ , min, es decir  $HV_{perc} = \frac{HV_{aprox}}{HV_{real}} \in [0,1]$ , el cual se busca sea cercano a uno.

Hasta este punto se han considerado dos métricas especializadas una en la convergencia, otra en la dispersión de forma uniforme, mientras que la tercera puede dar información sobre convergencia y dispersión. En este punto únicamente resta agregar que, para múltiples análisis, será importante conocer el número de puntos aproximados o cardinalidad del frente cercano.

### 3.4 Metodología para la configuración de parámetros

Se decidió utilizar la configuración de parámetros, conocida como diseño factorial completo, el cual consiste en seleccionar un conjunto de valores potenciales para cada parámetro y probar todas las combinaciones de éstos (véase el ejemplo de la Figura 5). A veces este método es también llamado “fuerza bruta”, aludiendo a que es la forma exhaustiva o más costosa de hacer una calibración, esto último debido al gran número de experimentos que deben realizarse:

$$s = q_1 q_2 \dots q_p \quad (4)$$

Donde  $s$  es el número de experimentos y  $q_i$  es el número de valores que deseamos probar para el  $t = \text{ésimo}$  parámetro. Por otro lado, la información que se obtiene a partir de éste tipo de análisis es más detallada, debido a que se realiza un mayor número de pruebas, lo cual proporcionar mejores indicios acerca del juego de parámetros adecuado.

Para realizar un análisis exhaustivo, es necesario hacer una delimitación del intervalo en el que varía cada parámetro. Dado que se debe seleccionar un número muy limitado de valores dentro de tales intervalos será necesario elegirlos de manera apropiada y, posteriormente profundizar la búsqueda en el intervalo para encontrar el valor más adecuado. Debido a esto, en la presente investigación se realizaron las etapas de calibración en dos fases llamadas: valores extremos y diseño factorial completo.

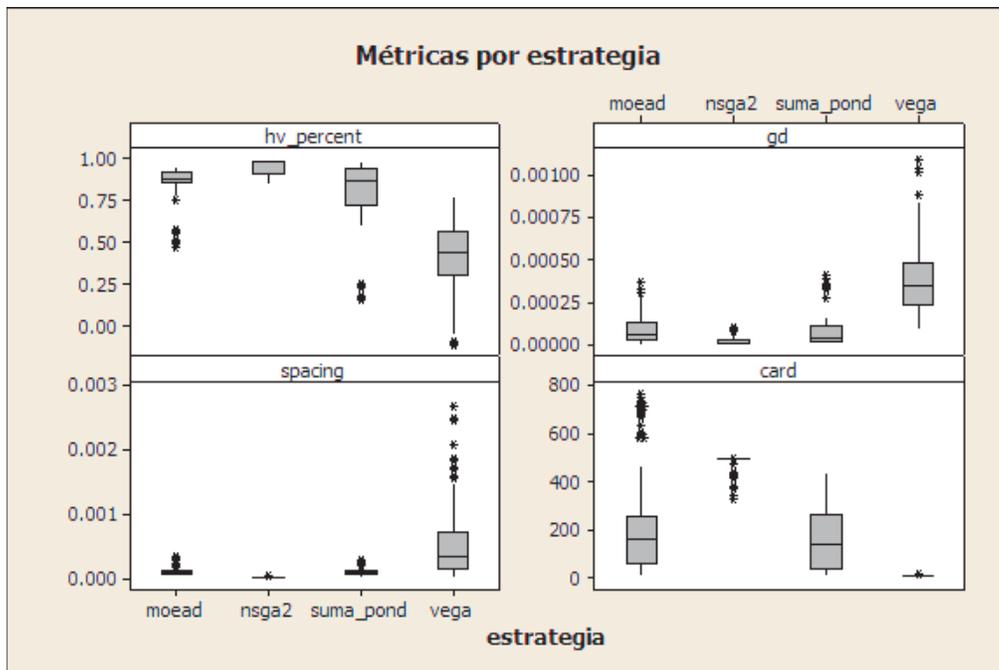
### 3.4.1 Presentación gráfica de resultados

La etapa de experimentación para seleccionar el método de manejo de restricciones se realizó bajo las siguientes condiciones:

- 30 experimentos por técnica de manejo de restricciones, por meta-heurística y por instancia (inst0 = Hang Seng31, inst1 = DAX85, inst2 = FTSE89, inst3 = S&P98, inst4 = Nikkei225).
- Análisis previo de ajuste de parámetros.
- Cruza y mutación en NSGA-II y MOEA/D se utiliza el método uniforme, mientras que a SPO y a VEGA les corresponde SBX.
- Criterio de paro:  $1000N$  llamadas a la función objetivo, donde  $N$  es el número total de activos de la instancia de acuerdo al modelo matemático (de 31 a 225).
- Archivo  $EP$  en el MOEA/D restringido a  $\Psi$  elementos, donde  $\Psi$  es el tamaño de la población.

En la etapa experimental de la selección de método de cruza y mutación se empezó a notar, a grandes rasgos, que los métodos de mejor eficiencia fueron NSGA-II y SPO. En la presente etapa experimental, antes de analizar

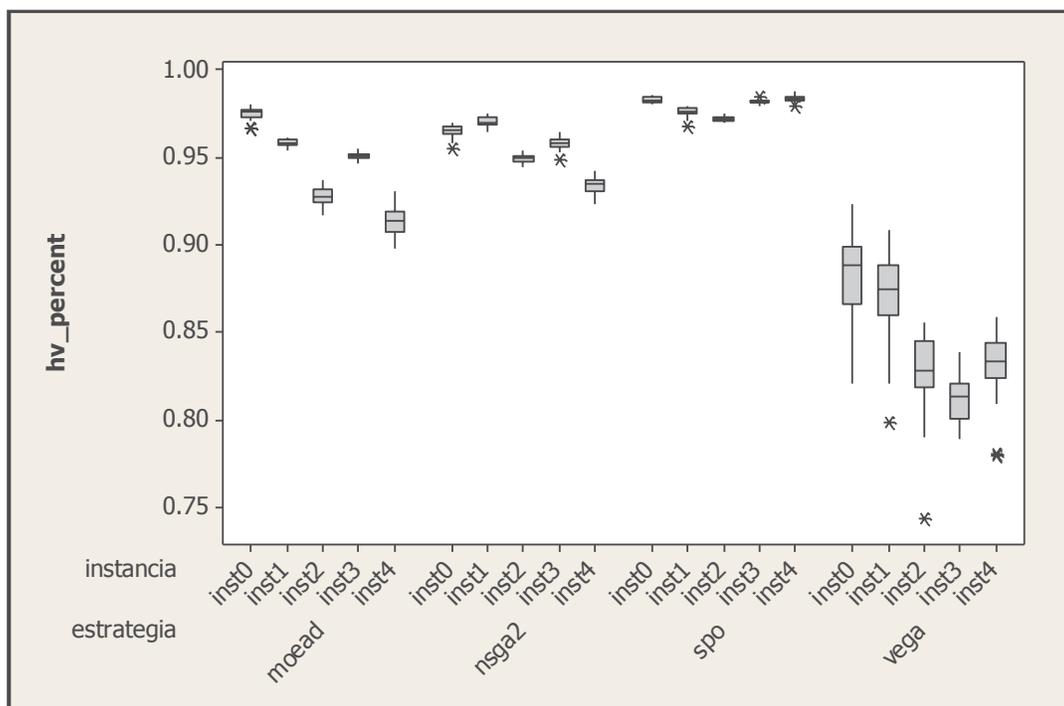
Figura 6. Métricas por estrategia



los métodos de manejo de restricciones, será aprovechada la información obtenida, para hacer algunas conclusiones generales sobre la eficiencia de los algoritmos hasta este punto de la investigación. De acuerdo a la Figura 6, la cual valora la eficiencia de cada técnica (sin importar la forma en la que se manejaron las restricciones) el orden de efectividad de mayor a menor es: NSGA2, MOEA/D, SPO y VEGA; lo anterior con respecto sólo a las métricas  $HV_{perc}$ ,  $gd$  y  $ESS$ . Las cardinalidades son muy variables con respecto a cada metaheurística y no definen la calidad del frente aproximado, además, tómesese en cuenta que se hace una calibración de parámetros que no suele sugerir el mismo valor para el tamaño de la población en todas las técnicas. Hasta ahora, VEGA se ha mostrado como una técnica muy inestable (en todas las métricas muchos valores atípicos, varianza muy alta).

Cada experimento realizado genera una aproximación del frente de Pareto de una instancia en particular, por lo cual no es posible presentarlos todos en el presente documento de forma gráfica. De modo que se enseñarán los frentes graficados correspondientes a la corrida mediana de cada estrategia en cada instancia de acuerdo a su desempeño en  $HV$ . Los frentes se

Figura 7. Implementación



muestran en el mismo gráfico dado que se aprecia se esta forma la cercanía con el frente real y entre estrategias, y también por separado, pues de este modo se observan mejor los rasgos en la forma del frente aproximado.

Las métricas *gd* y *ESS* pueden ser comparadas entre cada una de las estrategias pero sólo si se hace referencia a la misma instancia, en la Figura 8, 9 y 10 se muestran las métricas restantes.

De acuerdo a la métrica distancia generacional (Figura 8), nuevamente se observan los mejores resultados con SPO y los peores con VEGA. Por otro lado *ESS* en la Figura 9 suele variar con mayor dispersión aunque refleja conclusiones similares. Finalmente la particularidad de la métrica cardinalidad (Figura 10) es que SPO muestra resultados con muchos puntos, por otro lado, en NSGA-II el tamaño de la población es un factor sumamente crucial, por ende siempre se mantiene pequeña.

En las Figuras 11 a 19 están disponibles los frentes aproximados por cada técnica estudiada, sobre cada una de las instancias de interés en este estudio. Estas gráficas permiten apreciar visualmente las características de los frentes aproximados, tanto en términos de convergencia como de diversidad; mismos conceptos que capturan los indicadores presentados en los

Figura 8. Implementación Distancia Generacional

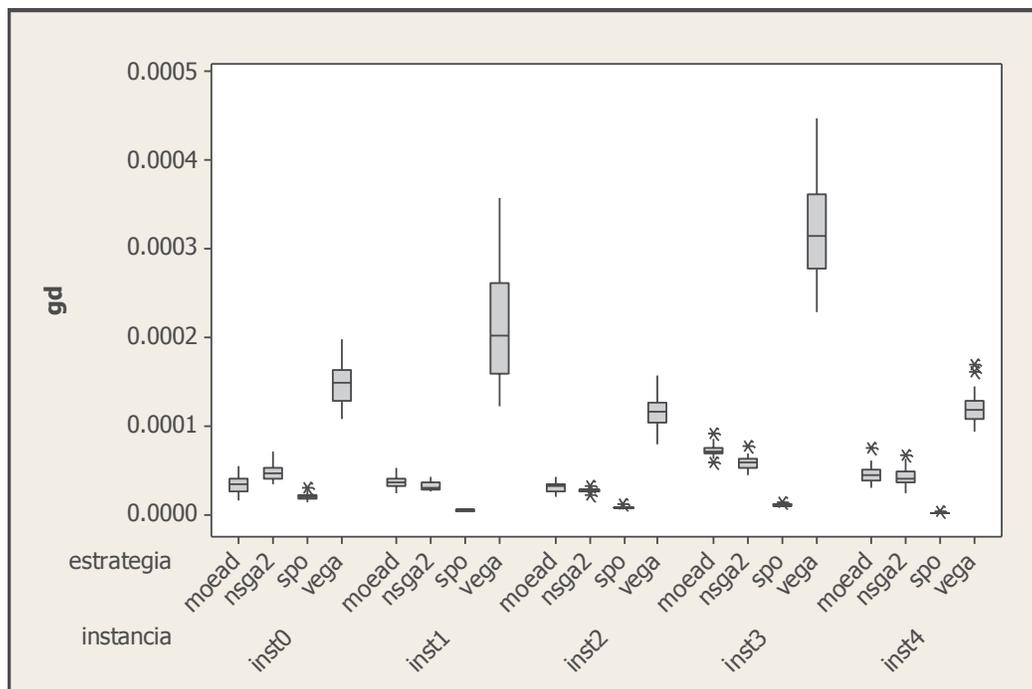


Figura 9. Implementación ESS

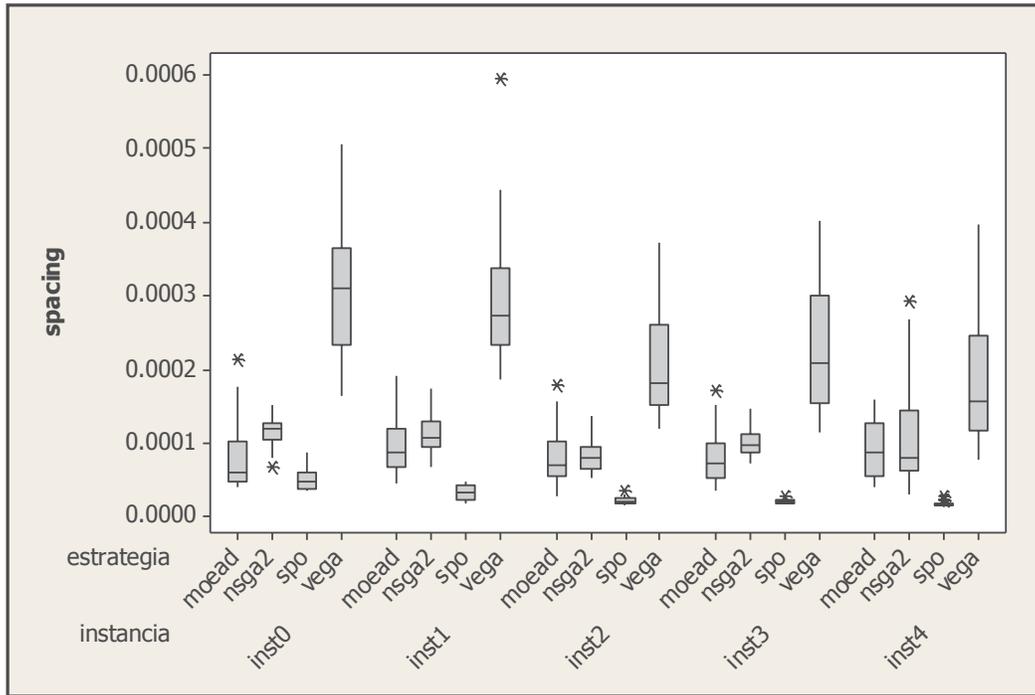
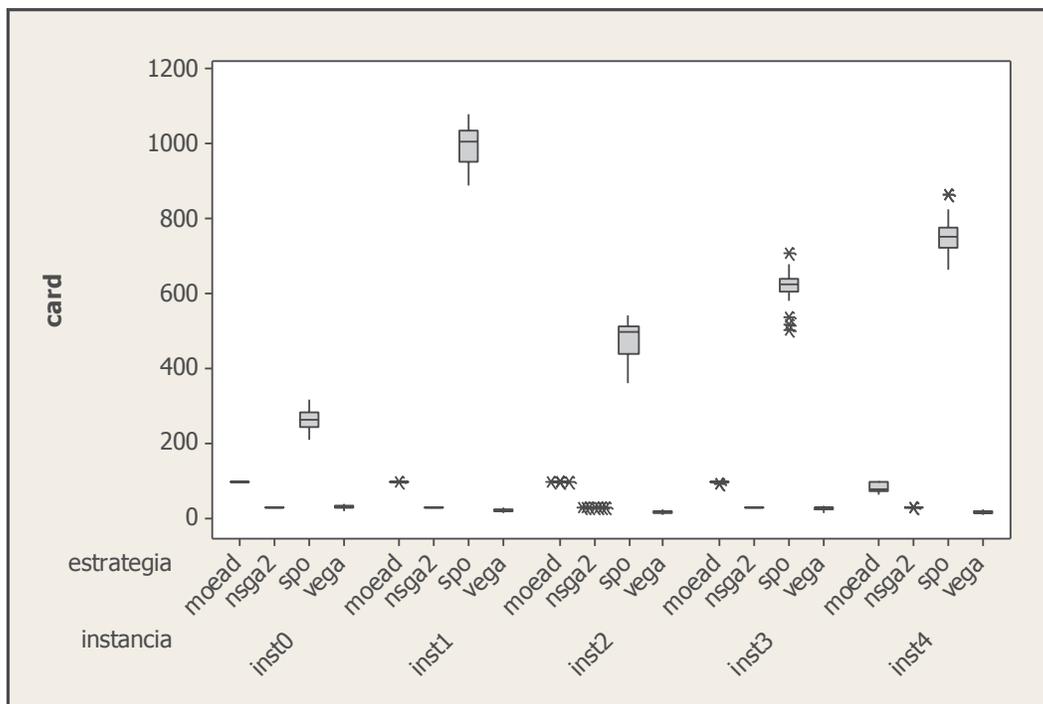


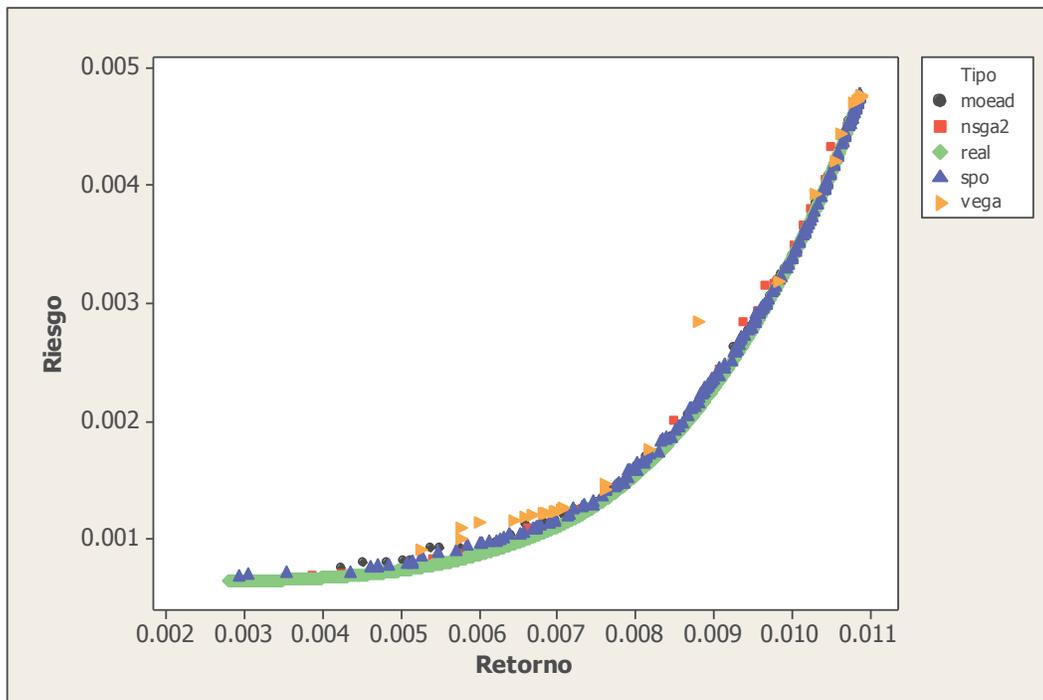
Figura 10. Implementación Cardinalidad



cuadros 1 a 6. Se puede observar en estas figuras el buen comportamiento de los algoritmos evolutivos implementados, que a excepción de VEGA logran identificar una aproximación bien distribuida del frente real. Cabe mencionar que todos los tiempos indicados en estas figuras se expresan en segundos. Finalmente, la figura 20 compara los resultados obtenidos en este estudio con los del estado del arte, en base a la métrica de Distancia Generacional.

Cabe mencionar que la métrica  $gd$  está expresada en una magnitud muy pequeña y además existe una relación inversa entre el valor de la métrica y la dificultad para mejorarla. Por lo anterior, la variación porcentual es una buena medida de la diferencia de eficiencia entre los algoritmos, dado que, a medida que el resultado ( $\nu_0$ ) es más cercano a cero, la diferencia con respec-

Figura 11. Instancia Hang Sen 31 mediana



Cuadro 1.  $HV_{perc}$  Hang Seng31 mediana

estrategia	tiempo	hv_percent
moead	9.386440039	97.6147%
nsga2	10.27579498	96.5479%
spo	8.684428215	98.2979%
vega	9.466565847	89.2091%

Figura 12. Hang Sen31 riesgo-rendimiento

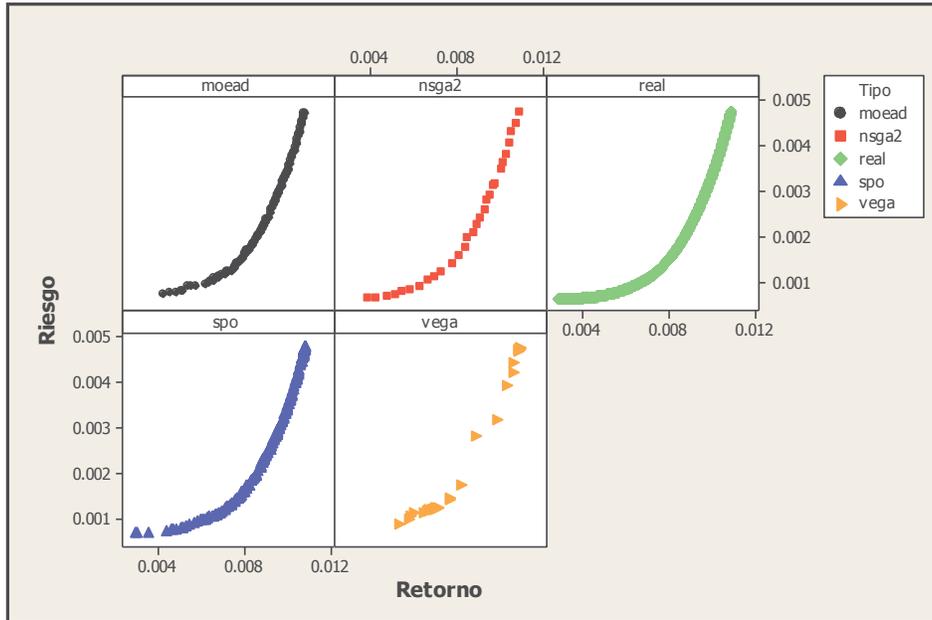
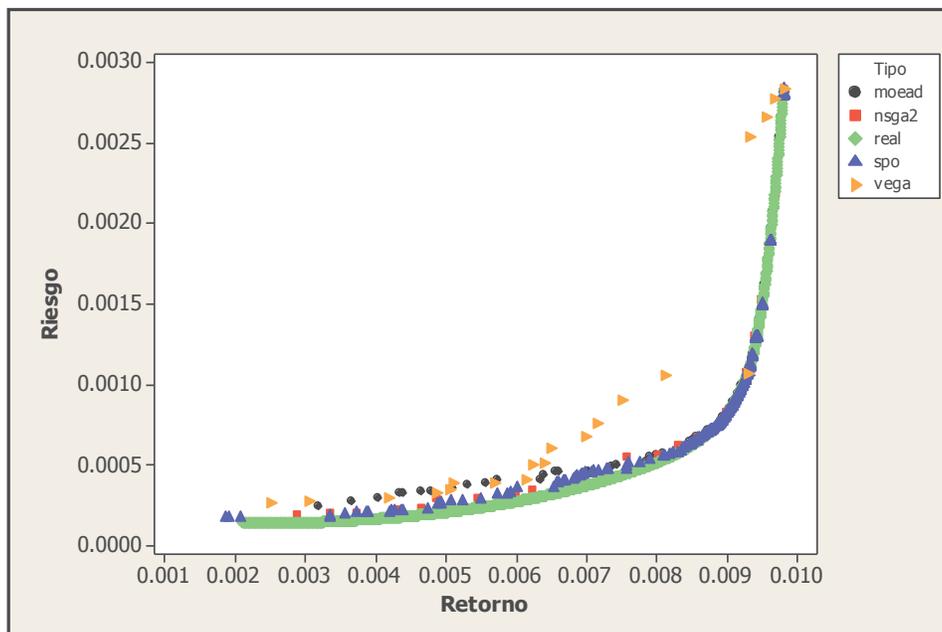


Figura 13. Instancia DAX85 mediana



Cuadro 2.  $HV_{perc}$  DAX85 mediana

estrategia	tiempo	hv_percent
moead	137.469667	95.8584%
nsga2	139.918093	97.0631%
spo	134.3552189	97.6835%
vega	137.5779932	87.5550%

Figura 14. DAX85 mediana riesgo-rendimiento

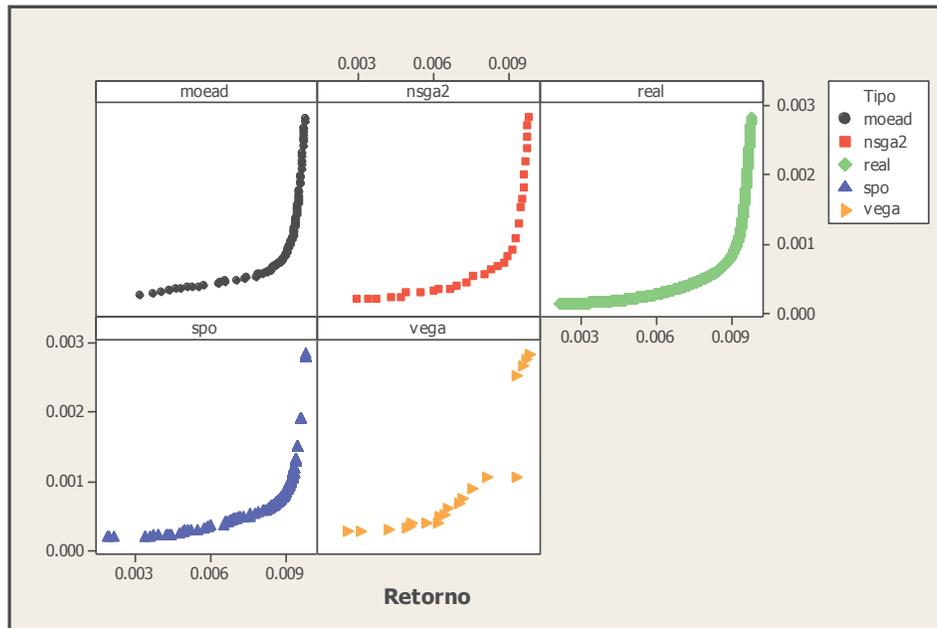
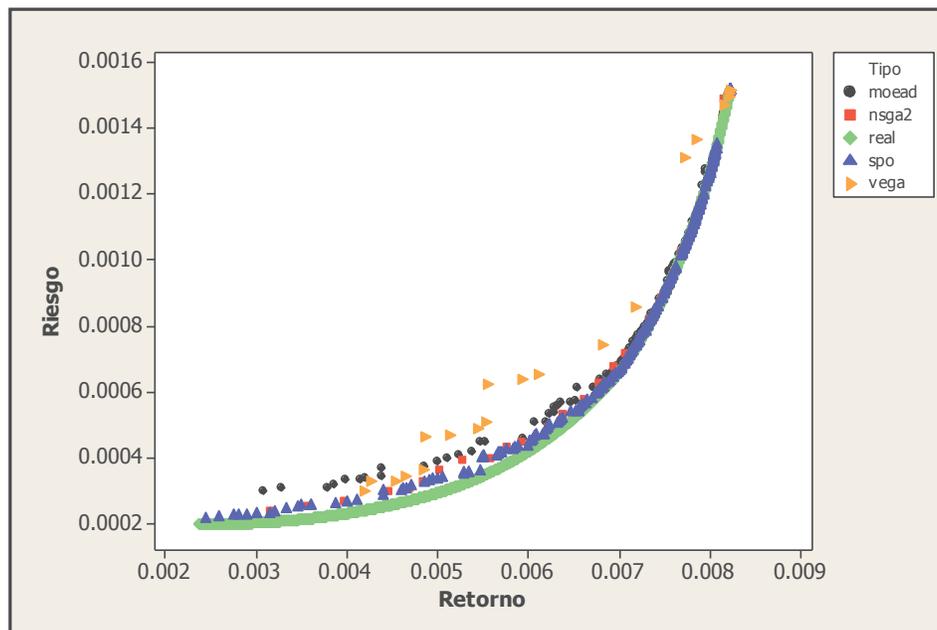


Figura 15. Instancia FTSE89 mediana



Cuadro 3.  $HV_{perc}$  FTSE89 mediana

estrategia	tiempo	hv_percent
moead	153.8806272	92.7641%
nsga2	150.2436111	94.9682%
spo	154.263298	97.2482%
vega	154.0000622	82.8718%

Figura 16. FTSE89 mediana riesgo-rendimiento

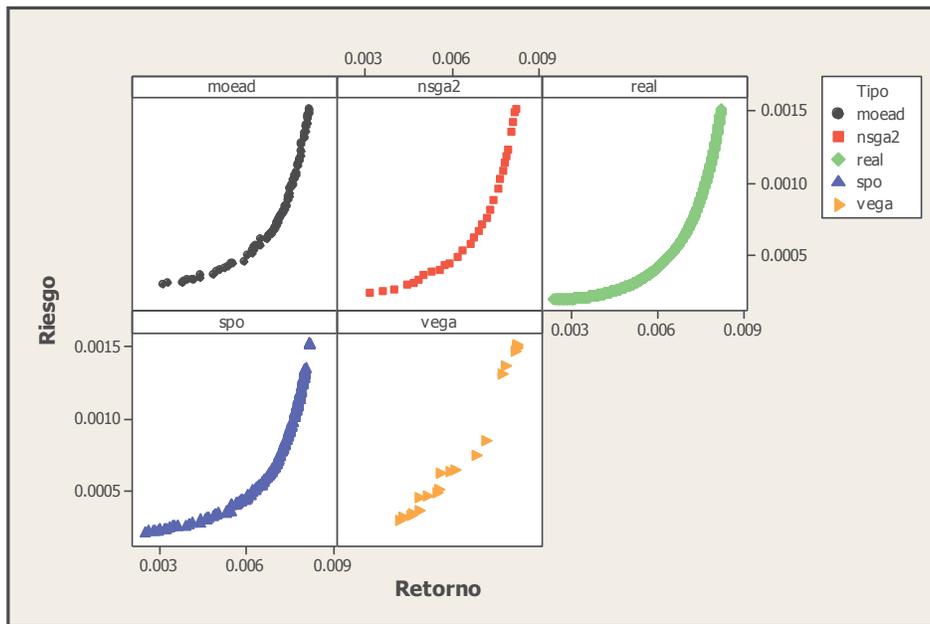
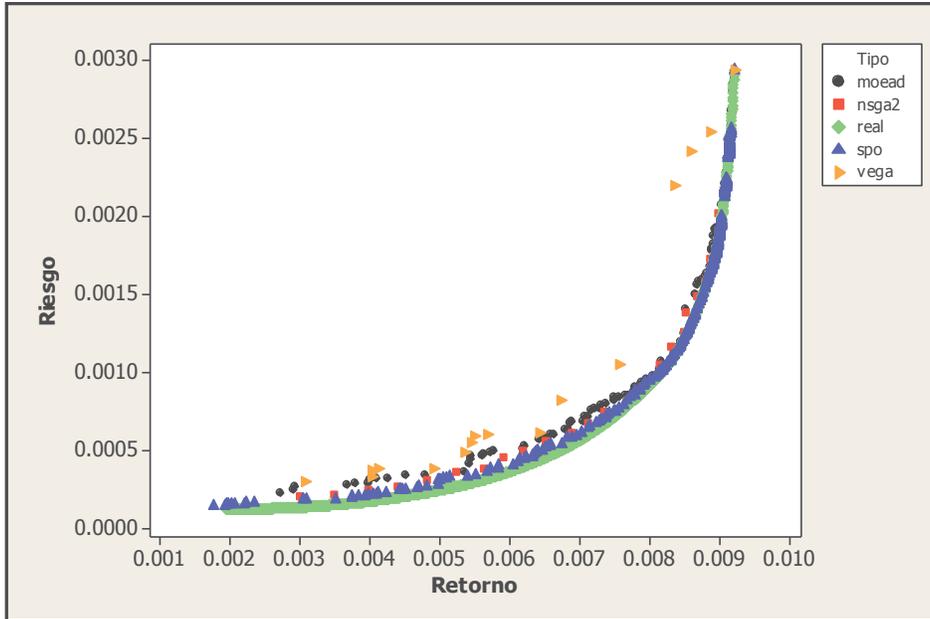


Figura 17. Instancia S&P98 mediana



Cuadro 4.  $HV_{perc}$  S&P98 mediana

estrategia	tiempo	hv_percent
moead	201.986851	95.0938%
nsga2	198.9523768	95.7831%
spo	204.486424	98.1723%
vega	202.1144722	81.3964%

Figura 18. S&P98 mediana riesgo-rendimiento

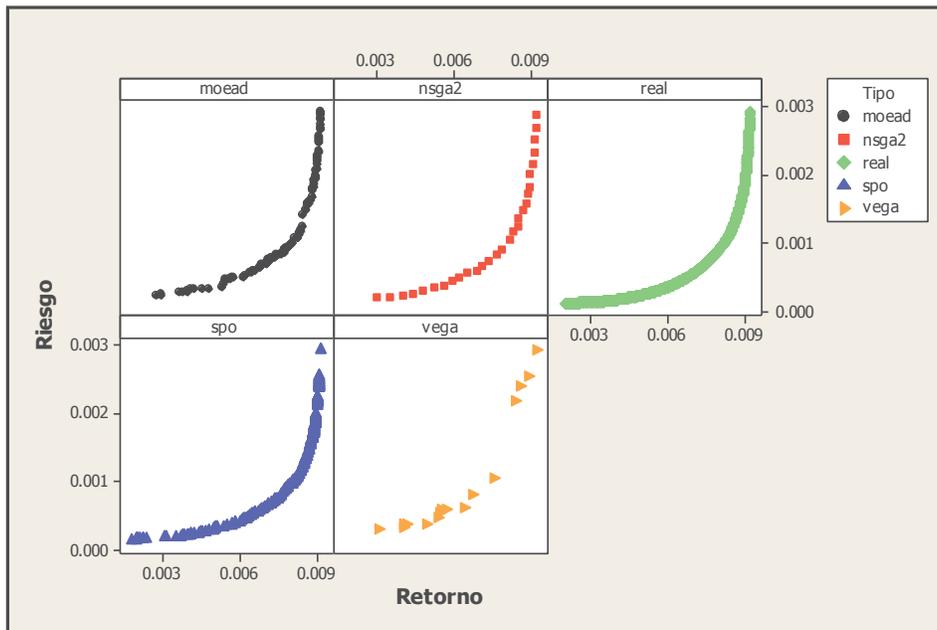
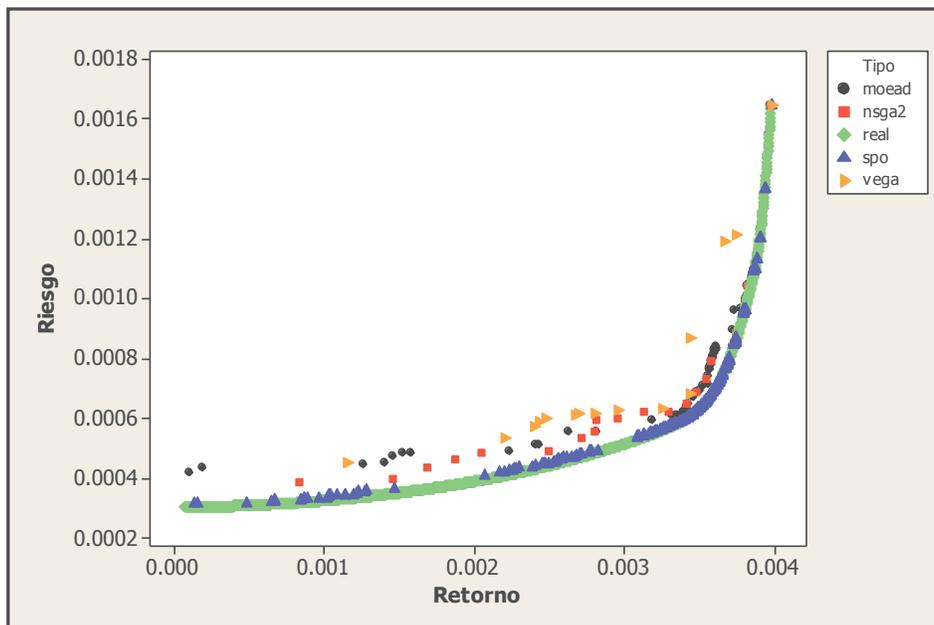


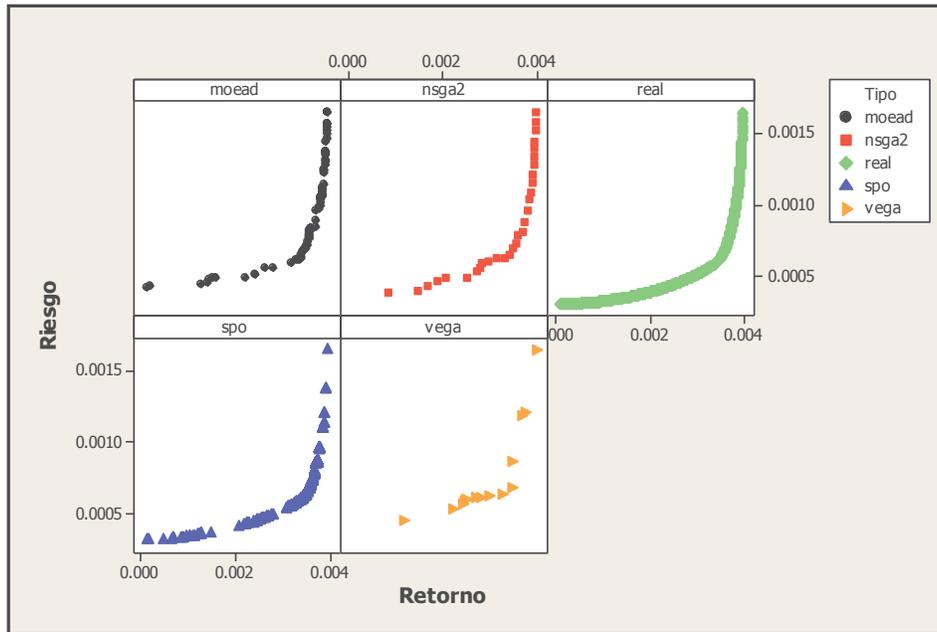
Figura 19. Nikkei225 mediana



Cuadro 5.  $HV_{perc}$  Nikkei225 mediana

estrategia	tiempo	hv_percent
moead	2398.413085	91.4709%
nsga2	2289.414533	93.4996%
spo	2605.19272	98.4124%
vega	2406.257872	83.4054%

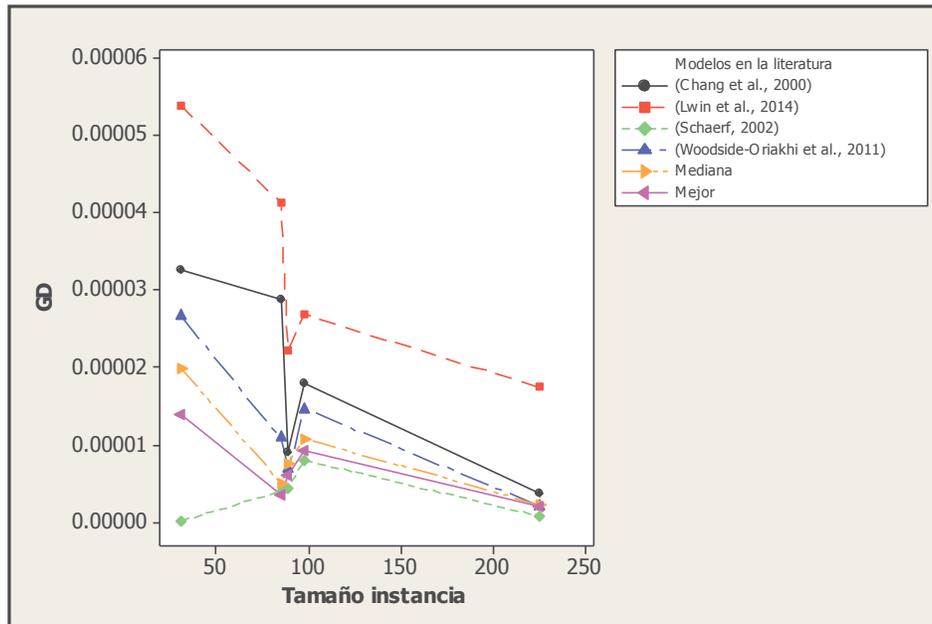
Figura 20. Nikkei225 mediana riesgo-rendimiento



Cuadro 6. Ranking con respecto a la mediana de cada métrica (mayor a menor desempeño, en rojo la menor)

Posición	$HV_{perc}$		$gd$		$ESS$		$card$	
	estrategia	instancia	estrategia	instancia	estrategia	instancia	estrategia	instancia
1	SPO	inst4	<b>SPO</b>	<b>inst4</b>	SPO	inst4	SPO	inst1
2	SPO	inst0	SPO	inst1	SPO	inst3	SPO	inst4
3	SPO	inst3	SPO	inst2	SPO	inst2	SPO	inst3
4	SPO	inst1	SPO	inst3	SPO	inst1	SPO	inst2
5	MOEA/D	inst0	SPO	inst0	SPO	inst0	SPO	inst0
6	SPO	inst2	NSGA-II	inst2	MOEA/D	inst0	MOEA/D	inst0
7	NSGA-II	inst1	NSGA-II	inst1	MOEA/D	inst2	MOEA/D	inst2
8	NSGA-II	inst0	MOEA/D	inst2	MOEA/D	inst3	MOEA/D	inst3
9	MOEA/D	inst1	MOEA/D	inst0	NSGA-II	inst2	MOEA/D	inst1
10	NSGA-II	inst3	MOEA/D	inst1	NSGA-II	inst4	MOEA/D	inst4
11	MOEA/D	inst3	NSGA-II	inst4	MOEA/D	inst4	VEGA	inst0
12	NSGA-II	inst2	MOEA/D	inst4	MOEA/D	inst1	NSGA-II	inst2
13	NSGA-II	inst4	NSGA-II	inst0	NSGA-II	inst3	NSGA-II	inst4
14	MOEA/D	inst2	NSGA-II	inst3	NSGA-II	inst1	NSGA-II	inst3
15	MOEA/D	inst4	MOEA/D	inst3	NSGA-II	inst0	NSGA-II	inst1
16	VEGA	inst0	VEGA	inst2	VEGA	inst4	NSGA-II	inst0
17	VEGA	inst1	VEGA	inst4	VEGA	inst2	VEGA	inst3
18	VEGA	inst4	VEGA	inst0	VEGA	inst3	VEGA	inst1
19	VEGA	inst2	VEGA	inst1	VEGA	inst1	VEGA	inst2
20	VEGA	inst3	VEGA	inst3	VEGA	inst0	VEGA	inst4

Figura 21. Comparación con resultados publicados bajo la métrica  $gd$ .



to a la cantidad con la que será comparada ( $v_f$ ), representará un porcentaje mucho mayor del valor obtenido.

Con respecto a los mejores resultados obtenidos, se logra superar a Schaerf (2002) solamente en la instancia DAX85, por otro lado se superan todos los resultados restantes reportados para la métrica  $gd$ .

Finalmente, para ilustrar lo que pueden ser unas de las soluciones no-dominadas identificadas por nuestros algoritmos, en el Apéndice A se presentan dos soluciones eficientes, obtenidas respectivamente para las instancias 2 y 4 (ver apéndice A), ambas ubicadas cerca del punto ideal (es decir en la “rodilla” del frente aproximado) ya que éstas soluciones son muy a menudo seleccionadas por un tomador de decisión. Y en el Apéndice B se muestran los algoritmos metaheurísticos en pseudo código de las cuatro técnicas implementadas.

## Conclusiones

En el presente artículo se estudió, adaptó e implementó un conjunto de metaheurísticas sofisticadas (Jin *et al.*, 2016) y se analizaron sus resultados de aplicarlas a un problema financiero clásico. Se hizo frente a múltiples etapas de análisis a lo largo de la obtención de datos experimentales. En el transcurso de tales etapas, la información fue abriendo pauta a indagaciones acerca de las características más relevantes como el tipo de paradigma en el que se basan las metaheurísticas, o la convexidad y forma de los frentes de Pareto reales. Se han señalado algunos efectos en las variables, que repercuten en el funcionamiento eficiente de los algoritmos, así como plantear explicaciones acerca de las posibles causas que llevaron a ello, lo cual funge como el valor más relevante de la presente investigación.

En el análisis de los métodos de selección de poblaciones no dominadas se logró destacar acerca de la relación entre comparaciones de dominancia y tiempo de ejecución, que no necesariamente guardan una relación directamente proporcional, es decir, un mayor uso de la operación de comparación, no implica necesariamente un mayor tiempo de ejecución (esto probablemente se debe a que, en el problema estudiado el número de objetivos es pequeño). En cuanto a la relación entre llamadas a la función objetivo para las metaheurísticas y el tiempo de ejecución, en la investigación fue ineludible tomar en cuenta la relación entre ambos factores. En el entorno multiobjetivo las operaciones de clasificación de Pareto constituyen un elemento de gran relevancia, y son particularmente costosas aun tratándose de dos objetivos. Fue necesario mantener un límite de población bajo para el NSGA-II, para que el tiempo de ejecución no se elevara ( $O(\Psi \log^{m-1} \Psi)$ ). En SPO y MOEA/D en un menor grado aunque no se eligieron tamaños de población grandes. Por otra parte el tamaño de población fue un parámetro que no se relacionó con el tiempo en VEGA.

El método exhaustivo para el ajuste de parámetros, a pesar de tener un costo computacional elevado, fue posible implementarlo ejecutando un número relativamente bajo de pruebas para poder llegar a una configuración que garantizara buenos resultados. Analizar valores extremos para los parámetros fue interesante debido a que, no solo se delimitó un intervalo de búsqueda, sino que se examinaron valores poco convencionales en la práctica que, en algunos casos, fueron adecuados.

Se observó, en el transcurso de la investigación, que algunas veces es adecuado determinar si la heurística está utilizando tiempo innecesario (a

partir de cierto tiempo ya no se puede mejorar, y no por la presencia de un óptimo local). Es decir, hay veces que el MOEA/D alcanza  $HV_{perc} = 95\%$  en 10 segundos y, a partir de ese momento, ya no hay más mejoras significativas. O sea que en 10 segundos MOEA/D ya dio el máximo rendimiento y ya no tiene sentido ejecutarlo por más tiempo. En un estudio posterior, podría estudiarse más a fondo este límite de eficiencia, el cual es muy distinto en cada estrategia.

No se esperaba que un Algoritmo Genético basado en una Suma Ponderada de Objetivos se mostrara como el de mayor eficiencia. Esto se debe, posiblemente a dos factores, el primero, el número de objetivos es bajo, y, por otro lado, los frentes reales son muy uniformes, sin discontinuidades y convexos.

Analizando la forma operativa de cada algoritmo, se abre pauta a concluir que hay factores dentro de éstos, que los hacen mejores al manejar las características particulares de cada instancia resuelta, la presente investigación abrió pauta al estudio a detalle de este aspecto como una perspectiva a futuro, inclusive cada estrategia que fue implementada, es sensible de diferente forma a cada uno de los parámetros calibrados. Aunque el tamaño de la instancia es crucial en la dificultad del problema, otras características del frente de Pareto también tendrán relevancia en la eficacia con la que particularmente cada estrategia resuelva el problema, y ese es un factor que no hay que perder de vista al elegir una técnica para resolver un MOP.

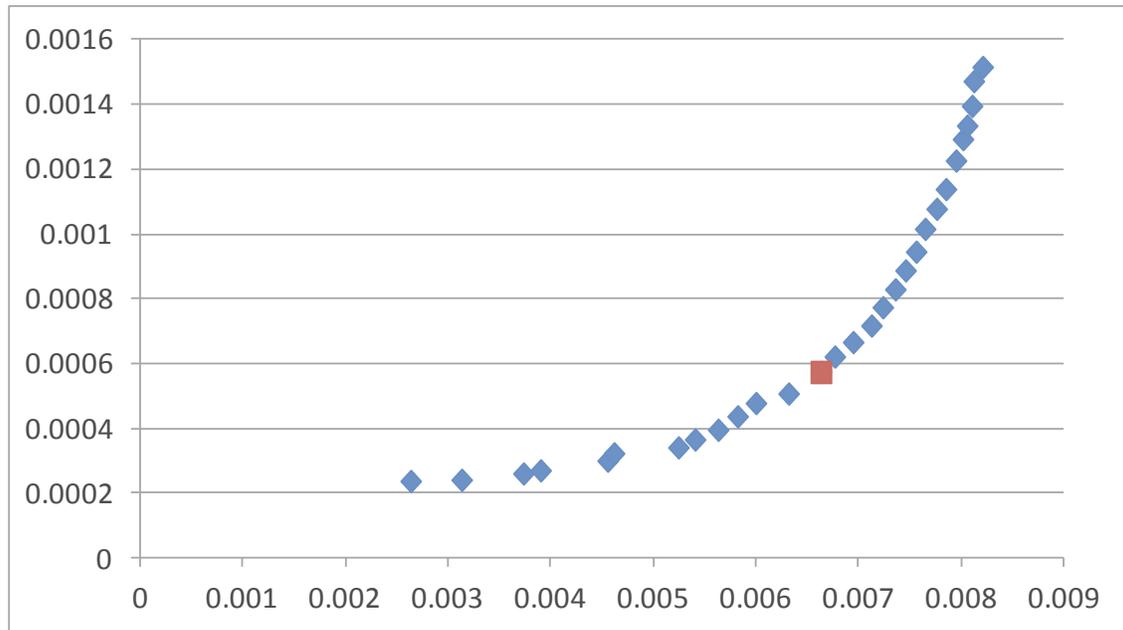
Cabe mencionar, que tampoco la diferencia de desempeño entre estrategias fue tan significativa. En general los resultados obtenidos son satisfactorios para MOEA/D, NSGA-II y SPO. La estrategia VEGA fue de las primeras propuestas para MOP, debido a esto, algoritmos más recientes logran superarla fácilmente. Sin considerar esta excepción, estas conclusiones validan la hipótesis de investigación del presente trabajo, es decir que los Algoritmos Evolutivos Multiobjetivo representan una opción viable para resolver el problema de Selección de Portafolios de Inversión. Se establece una clasificación de los algoritmos implementados en términos de la calidad de su desempeño, resultado de una etapa de calibración exhaustiva y realizada para las cuatro técnicas de interés.

## Referencias bibliográficas

- Beasley, J. E. (2015). *OR Library*. <http://people.brunel.ac.uk/~mastjjb/jeb/orlib/portinfo.html>.
- Deb, K., Pratap, A., Agarwal, S., y Meyarivan, T. (2002). "A Fast and Elitist Multiobjective Genetic Algorithm: NSGA-II". *IEEE TRANSACTIONS ON EVOLUTIONARY COMPUTATION*, pp. 182-197.
- Hernández-Lerma, O., y Hoyos-Reyes, L. F. (2011). "A multiobjective formulation of optimal control problems with additive costs". *Administración de riesgos. Volumen III. Modelos y entorno financiero*. México, D.F.: UAM-Azcapotzalco, pp. 245-271.
- Jin, Y., Qu, R., y Atkin, J. (2016). "Constrained Portfolio Optimisation: the state-of-the-art Markowitz Models". *The 2016 International Conference on Operations Research and Enterprise Systems*, Roma: University of Nottingham, UK, Faculty of Science, School of Computer Science, pp. 88-395.
- Markowitz, H. (1952). "Portfolio Selection". *The Journal of Finance*, vol. 7, núm. 1, pp. 77-91.
- Merton, R. C. (1972). "An Analytic Derivation of the Efficient Portfolio Frontier". *The Journal of Financial and Quantitative Analysis*, vol. 7, núm. 4, pp. 1851-1872.
- Ponsich, A., López Jaimes, A., y Coello, C. A. (2013). "A Survey on Multiobjective Evolutionary Algorithms for the Solution of the Portfolio Optimization Problem and Other Finance and Economics Applications". *IEEE Transactions on Evolutionary Computation*, vol. 17, núm. 3, pp. 321 - 344.
- Schaerf, A. (2002). "Local search techniques for constrained portfolio selection problems". *Computational Economics*, núm. 20, pp. 177-190.
- Schaffer, J. D. (1985). "Multiple Objective Optimization with Vector Evaluated Genetic Algorithms". *Genetic Algorithms and their Applications: Proceedings of the First International Conference on Genetic Algorithms*, New Jersey, USA: Lawrence Erlbaum, pp. 93-100.
- Srinivas, N., y Deb, K. (1995). "Multiobjective function optimization using non-dominated sorting genetic algorithms". *Evol. Comput.*, vol. 2, núm. 3, pp. 221-248.
- Zhang, Q., y Li, H. (2007). "MOEA/D: A Multiobjective Evolutionary Algorithm Based on Decomposition". *IEEE Transactions on Evolutionary Computation*, vol. 11, núm. 6, pp. 712 - 731.
- Zitzler, E., Thiele, L., Laumanns, M., Fonseca, C., y Grunert da Fonseca, V. (2003). "Performance Assessment of Multiobjective Optimizers: An Analysis and Review". *IEEE, Transactions on Evolutionary Computation*, vol. 7, núm. 117 p. 132.

## Apéndice A. Descripción de dos soluciones eficientes

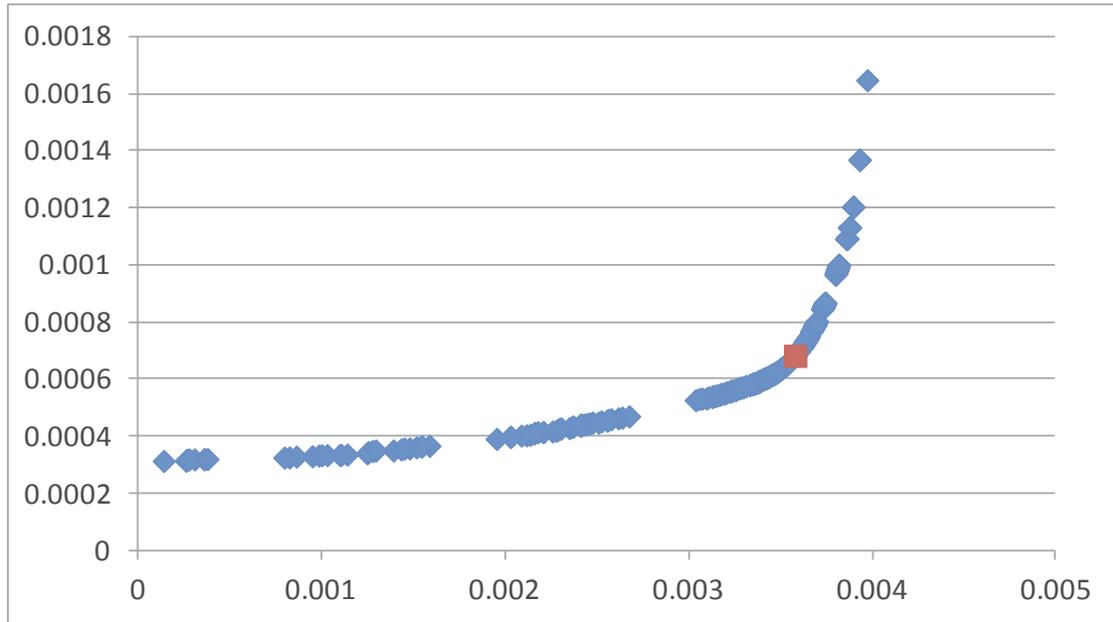
1. Solución encontrada con la estrategia “NSGA-II”, cerca de la “rodilla” del frente para la instancia 2 (FTSE89, Reino Unido). Nota: sólo se indican las variables estrictamente positivas.



Detalle de la solución:

$x_1$	0.073009922
$x_2$	0.016520877
$x_9$	0.183325563
$x_{17}$	0.423983624
$x_{25}$	0.009790336
$x_{36}$	0.191115913
$x_{61}$	0.038100786
$x_{70}$	0.016240054
$x_{81}$	0.047912925
Rendimiento esperado	0.00662809
Riesgo (varianza)	0.000572884

2. Solución encontrada con la estrategia “SPO”, cerca de la “rodilla” del frente para la instancia 4 (Nikkei225, Japón). Nota: sólo se indican las variables estrictamente positivas.



Detalle de la solución:

$x_8$	0.29327773
$x_{42}$	0.12549203
$x_{61}$	0.34957127
$x_{114}$	0.03953417
$x_{213}$	0.1921248
Rendimiento esperado	0.003578485
Riesgo (varianza)	0.000682816

## Apéndice B. Algoritmos en pseudocódigo

### 1. Algoritmo genético simple para optimización escalar considerando Suma Ponderada de Objetivos (SPO).

```
Variables de entrada

NP: tamaño de la población
Gmax: Número de generaciones
zm: Tasa de mutación

generar población aleatoria factible pob_f
para generación en {0, ..., Gmax-1}:
    para i en pob_f:
        selección de padres
        cruza
        si |pob_hijos| < |pob_f|:
            agregar hijo a pob_hijos
        mutación a una tasa especificada zm
    p = unión población hijos y padres
    p_nueva = selección(p)
    pob_f = p_nueva
devolver mejor individuo de pob_f
```

### 2. Algoritmo VEGA Canónico.

```
variables de entrada

NP: Tamaño de la población
Gmax: Número de generaciones
obj: Número de objetivos

inicio
#tamaño de subpoblación
tam_sub = NP/obj
generar población aleatoria factible pob_f
para generación en [0, ..., Gmax-1]:
    m_pool = ∅
    para k en [0, ..., obj-1]:
        ordenar pob_f de menor a mayor con respecto al valor del objetivo k
        sub_pob = ∅
        i = 0
        mientras tamaño de sub_pob < tam_sub:
            agregar pob_f[i] a sub_pob
            i = i + 1
        m_pool = m_pool U sub_pob
    ordenar aleatoriamente a m_pool
    mientras cardinalidad(pob_hijos) < NP:
```

```

    seleccionar padres de m_pool
    cruza(padre1, padre2) #se producen 2 hijos
    agregar hijo a pob_hijos
    mutacion de pob_hijos
    pob_f = pob_hijos
    devolver no dominados de pob_f

```

### 3. Algoritmo NSGA-II.

```

Gmax: Número de generaciones
NP: Tamaño de la población

Inicio
Generar pob_f
Para generacion en [0, ..., Gmax-1]:
    mientras tamaño de pob_hijos < tamaño de pob_f:
        selección de padres por torneo binario
        cruza(padre1, padre2) #se producen 2 hijos
        agregar hijos a pob_hijos
    mutar pob_hijos
    p_total = pob_f U pob_hijos
    partición(p_total) = ordenar por jerarquía de dominancia (menor a mayor rango)
    p_nueva = ∅
    para cada frente en la partición:
        unir frente con p_nueva si cardinalidad(p_nueva) ≤ NP:
            elegir individuos de último frente en función de crowding-distance
    pob_f = p_nueva
ob_f = no dominados de pob_f
evolver pob_f

```

#### crowding-distance

```

L: número de objetivos
frente en el espacio de los objetivos
I: cardinalidad(I)

rd = [0, 0, ..., 0] #crowding distances
Para m en [0, 1, ..., obj-1]:
    I = ordenar I con respecto al objetivo m de menor a mayor
    crd[0] = ∞
    crd[L-1] = ∞
    para i en [1, 2, ..., L-2]:
        si I[L-1][m] - I[0][m] ≠ 0:
            I[i][obj] = I[i][obj] + (I[i+1][m] - I[i-1][m]) / (I[L-1][m] - I[0][m])
rd = ordenar crd de menor a mayor
evolver crd

```

#### 4. Algoritmo MOEA/D.

```
NP ← tamaño de la población
Gmax ← número de generaciones (criterio de paro)
T ← número de vecinos para cada individuo (preferiblemente par)

inicio
  generar vectores de peso # e.g. lam = {[0, 1], [0.1, 0.9], [0.2, 0.8], ... , [1, 0]}
  generar población pob_f
  agregar no dominados de pob_f a EP
  z = los mejores valores para cada objetivo en EP
  cálculo de aptitud de individuos en pob_f (g_bip)
para generación en Gmax hacer:
  para cada individuo en pob_f hacer:
    seleccionar dos vecinos
    cruzar produciendo dos hijos
    mutación de los dos hijos
    evaluación función objetivo de los dos hijos
    actualiza z ideal (si algún hijo es mejor en algún objetivo)
    #reemplazo de vecinos e individuo si hijo es mejor
  Para cada hijo hacer:
    Para cada vecino de individuo hacer:
      calcular el fitness del hijo con el vector de vecino
      Si fitness de hijo es mejor que el de vecino:
        reemplazar al vecino con hijo
    calcular el fitness de hijo con el vector de individuo
    Si el fitness calculado de hijo es mejor que el de individuo:
      reemplazar a individuo con hijo
  si ningún elemento de EP domina a hi:
    se integra hijo a EP
    se eliminan los dominados por hi en EP

devolver EP
```