



UNIVERSIDAD TÉCNICA ESTATAL DE QUEVEDO
FACULTAD DE CIENCIAS DE LA INGENIERÍA
CARRERA INGENIERÍA EN SISTEMAS

Proyecto de
Investigación previo a la
obtención del título de
Ingeniero en Sistemas.

Título del Proyecto de Investigación:

“OPTIMIZACIÓN MULTI OBJETIVO UTILIZANDO MALLAS VARIABLES”

Autor:

Naira Génesis Haro Bustamante

Director de Proyecto de Investigación:

Phd. Amilkar Puris Cáceres

Quevedo – Los Ríos - Ecuador.

2015

II. DECLARACIÓN DE AUTORÍA Y CESIÓN DE DERECHOS.

Yo, Naira Génesis Haro Bustamante, declaro que el trabajo aquí descrito es de mi autoría; que no ha sido previamente presentado para ningún grado o calificación profesional; y, que he consultado las referencias bibliográficas que se incluyen en este documento.

La Universidad Técnica Estatal de Quevedo, puede hacer uso de los derechos correspondientes a este trabajo, según lo establecido por la Ley de Propiedad Intelectual, por su Reglamento y por la normatividad institucional vigente.

f. _____

Naira Génesis Haro Bustamante

III. CERTIFICACIÓN DE CULMINACIÓN DEL PROYECTO DE INVESTIGACIÓN.

El suscrito, Amilkar Yudier Puris Cáceres, Docente de la Universidad Técnica Estatal de Quevedo, certifica que el estudiante Naira Génesis Haro Bustamante , realizó el Proyecto de Investigación de grado titulado “Optimización Multiobjetivo utilizando Mallas Variables”, previo a la obtención del título de Ingeniería en Sistemas, bajo mi dirección, habiendo cumplido con las disposiciones reglamentarias establecidas para el efecto.

.....

PhD. Amilkar Yudier Puris Cáceres
DIRECTOR DE PROYECTO DE INVESTIGACIÓN

**IV. CERTIFICADO DEL REPORTE DE LA HERRAMIENTA DE
PREVENCIÓN DE COINCIDENCIA Y/O PLAGIO ACADÉMICO.**

V. CERTIFICADO DE APROBACIÓN POR TRIBUNAL DE SUSTENTACIÓN.



**UNIVERSIDAD TÉCNICA ESTATAL DE QUEVEDO
FACULTAD DE CIENCIAS DE LA INGENIERÍA
CARRERA DE INGENIERIA EN SISTEMAS
PROYECTO DE INVESTIGACION**

Título:

“Optimización multiobjetivo utilizando Mallas Variables”

Presentado a la Comisión Académica como requisito previo a la obtención del título de Ingeniero en Sistemas.

Aprobado por:

PRESIDENTE DEL TRIBUNAL DE TESIS
Ing. Oviedo Byron

MIEMBRO DEL TRIBUNAL DE TESIS
Ing. Jaramillo Iván

MIEMBRO DEL TRIBUNAL DE TESIS
Ing. Novoa Pavel

QUEVEDO – LOS RIOS – ECUADOR

2015

VI. AGRADECIMIENTO.

No podría finalizar este proyecto de investigación sin acordarme de todas aquellas personas que han hecho posible su realización y que me han apoyado, ayudado y compartido conmigo tantas experiencias durante el largo período universitario.

Es justo ahora que al ver tan cerca la ansiada meta, se hace memoria de todo los momentos buenos y malos que se dejan atrás, sabiendo que de ahora en adelante ya nada va a ser igual. Y pese a las ganas de terminar, queda siempre una sensación de nostalgia y buen recuerdo, especialmente de las personas que han hecho que mi etapa de formación sea una de las más importantes de mi vida.

Así, en primer lugar quiero dar las gracias a mis padres por todos estos años de paciencia y fortaleza en los que siempre he contado con su apoyo incondicional. Y como no mencionar a mis hermanos quienes siempre confiaron en mí y sabían que lograría alcanzar la meta.

Otra persona que no puedo dejar de mencionar es mi director de proyecto de investigación Phd. Amilkar Puris Cáceres por haber compartido todos conocimientos, y sobre enseñarme que con esfuerzo y dedicación todo se logra.

Un lugar especial en este apartado es para Orlando, quien con su incondicional apoyo y esa gran paciencia para aguantar mis agobios y cambios de humor cuando las cosas no salen como espero. Siempre tuvo una palabra de aliento para seguir adelante y no dejarme vencer por las adversidades.

A todos ellos, de corazón muchas gracias

VII. DEDICATORIA.

A Dios por haberme permitido llegar hasta este punto y darme salud para lograr mis objetivos, además de su infinita fortaleza y amor.

Dedico este proyecto de investigación a todas las personas que forman y formaron parte de mi vida, en especial a mis padres por enseñarme a ser la persona que soy y a quienes admiro por la fortaleza de carácter que tienen a pesar de todos los problemas que han tenido que enfrentar en su vida juntos y por ayudarme e impulsarme a cumplir mis metas fijadas hasta este momento.

A mis hermanos por el apoyo incondicional y siempre estar ahí cuando más los he necesitado. Y por último, pero no menos importante a mi enamorado puesto que sin su palabra de aliento y su perseverancia no habría logrado alcanzar una meta más en mi vida.

VIII. RESUMEN Y PALABRAS CLAVES.

En este proyecto de investigación se presenta una propuesta de optimización multiobjetivo basada en el algoritmo de optimización de Mallas Variables (VMO). Esta propuesta se desarrollará mediante el estudio del funcionamiento que tienen los algoritmos evolutivos, los parámetros del mismo y sus operadores. Cabe recalcar que el algoritmo VMO realiza una fase denominada clearing mediante la cual utiliza el operador de selección, razón por la cual se considera posible la realización de esta propuesta.

Una vez desarrollado el nuevo algoritmo multiobjetivo se realizarán pruebas con problemas multiobjetivos los mismos que se encuentran descritos más adelante en este texto. Los resultados obtenidos serán comparados con los de otros algoritmos de optimización multiobjetivo.

El algoritmo VMO multiobjetivo fue incorporado en la herramienta Jmetal, la cual fue seleccionada por contar con varios algoritmos de optimización multiobjetivo y monobjetivo, problemas de optimización y por ser una herramienta de código abierto. Además fueron utilizadas algunas de las clases principales de la herramienta Jmetal y se implementaron otras para obtener un correcto funcionamiento del algoritmo.

IX. ABSTRACT AND KEYWORDS

This research project presents a proposal of multi-objective optimization based on the algorithm of optimization of Mallas Variables (VMO). This proposal will be developed through the study of running that are evolutionary algorithms, of the same one parameters and its operators. It is necessary to stress that the VMO algorithm carry out a phase called clearing by which uses the selection operator, reason why is considered possible the realization of this proposal.

Once developed the new multi-objective algorithm test will be realized with multi-objective problems the same ones that are described further on in text. The results obtained will be compared with other algorithms in multi-objective optimization.

The VMO multi-objective algorithm was incorporated into Jmetal tool, which was selected by having some optimization algorithms multi-objective and monobjetivo, optimization problems and for being an open source tool. In addition were used some of the main classes of Jmetal tool and others were implemented to obtain a correct running of the algorithm.

X. TABLA DE CONTENIDO

II. DECLARACIÓN DE AUTORÍA Y CESIÓN DE DERECHOS.....	ii
III. CERTIFICACIÓN DE CULMINACIÓN DEL PROYECTO DE INVESTIGACIÓN.....	iii
IV. CERTIFICADO DEL REPORTE DE LA HERRAMIENTA DE PREVENCIÓN DE COINCIDENCIA Y/O PLAGIO ACADÉMICO.	iv
V. CERTIFICADO DE APROBACIÓN POR TRIBUNAL DE SUSTENTACIÓN.....	v
VI. AGRADECIMIENTO.	vi
VII. DEDICATORIA.....	vii
VIII. RESUMEN Y PALABRAS CLAVES.....	viii
IX. ABSTRACT AND KEYWORDS	ix
X. TABLA DE CONTENIDO	x
XI. CÓDIGO DUBLÍN.....	xii
INTRODUCCIÓN	13
CAPÍTULO I.....	14
CONTEXTUALIZACIÓN DE LA INVESTIGACIÓN.....	14
1.1 Problema de investigación	15
1.1.1 Planteamiento del problema.....	15
1.1.2. Formulación del problema.	15
1.1.3. Sistematización del problema.....	15
1.2 Objetivos.	16
1.2.1 Objetivo general.....	16
1.2.2 Objetivos específicos.....	16
1.3 Justificación.....	16
FUNDAMENTACIÓN TEÓRICA DE LA INVESTIGACIÓN	18
2.1. Marco conceptual.....	19
2.1.1 Optimización	19
2.1.2 Optimización multiobjetivo.....	19
2.1.3 Dominancia de Pareto	20
2.1.4 Optimalidad De Pareto	21
2.1.5 Conjunto de óptimos de Pareto	21
2.1.9 Metaheurística de optimización de mallas variables (VMO).....	26
2.1.10 Problemas de prueba	28
Fig. 2.7 Descripción de los Problemas ZDT	30
2.1.11 Software para optimización multiobjetivo	30

2.2. Marco referencial	33
CAPÍTULO III	35
METODOLOGÍA DE LA INVESTIGACIÓN.....	35
3.1. Localización	36
3.2 Tipo de investigación	36
3.2.1 Exploratoria.....	36
3.3. Métodos de investigación.....	36
3.3.1 Método analítico.....	36
3.4. Fuentes de recopilación de información.....	36
3.5. Diseño de la investigación.....	37
3.5.1 Diseño experimental.....	37
3.6. Instrumentos de investigación.....	37
3.7. Tratamiento de los datos.	37
3.8. Recursos humanos y materiales.	38
CAPÍTULO IV	40
RESULTADOS Y DISCUSIÓN.....	40
4. Resultados y discusión	41
4.1 Medidas de rendimiento	41
4.1.2 Spread.....	43
4.2 Modelamiento del Algoritmo VMO Multiobjetivo.....	44
4.3 Integración del VMO en la Herramienta Jmetal.....	46
4.3 Resultados de la implementación del algoritmo VMO Multiobjetivo	48
4.3 Análisis de los resultados	51
CAPITULO V	55
CONCLUSIONES Y RECOMENDACIONES.....	55
CAPITULO VI.....	58
BIBLIOGRAFÍA.....	58
Referencias Bibliográficas	59
CAPITULO VII	62
ANEXOS.....	62
5. Anexos.....	63
5.1 Tablas de funciones objetivos del Algoritmo VMO-MO.....	63

XI. CÓDIGO DUBLÍN

Título:	“OPTIMIZACIÓN MULTI OBJETIVO UTILIZANDO MALLA VARIABLES”
Autor:	Haro Bustamante Naira Génesis
Palabras Claves:	Optimización, Multiobjetivo, Mallas Variables, Problemas de Optimización
Fecha de Publicación:	
Editorial:	
Resumen:	En este proyecto de investigación se presenta una propuesta de optimización multiobjetivo basada en el algoritmo de optimización de Mallas Variables (VMO). Esta propuesta se desarrollará mediante el estudio del funcionamiento que tienen los algoritmos evolutivos, los parámetros del mismo y sus operadores. Cabe recalcar que el algoritmo VMO realiza una fase denominada clearing mediante la cual utiliza el operador de selección, razón por la cual se considera posible la realización de esta propuesta.
Descripción:	67 hojas : dimensiones 21cm x 29.7cm
URI:	

INTRODUCCIÓN

La optimización tiene muchas aplicaciones en diversas ramas del conocimiento y su objetivo es encontrar las mejores soluciones a problemas del mundo real mediante el uso de modelos matemáticos. Dentro de la optimización tenemos dos tipos de formulaciones: la primera cuando se requiere optimizar un solo objetivo denominada monobjetivo y la segunda cuando se optimizan dos o más objetivos de forma simultánea llamada multiobjetivo.

Para resolver cualquiera de estos dos tipos de problemas, existe una gran cantidad de técnicas de programación matemática y métodos numéricos. Sin embargo una de las alternativas más utilizadas para resolver este tipo de problemas son las técnicas meta-heurísticas, las cuales pueden obtener soluciones a un costo computacional razonable. Estos métodos meta-heurísticos serán utilizados dentro del desarrollo del proyecto de investigación, en la resolución de problemas de optimización multiobjetivo.

Además cabe recalcar que la optimización multiobjetivo basada en técnicas evolutivas es una meta heurística que surgió con el fin de resolver problemas que involucran una serie de objetivos, caracterizada por ser capaz de obtener un conjunto de soluciones, con los mejores compromisos entre los objetivos optimizados (Frente Óptimo de Pareto)[1]. Dentro de la optimización multiobjetivo encontramos varios algoritmos propuestos por diferentes investigadores entre los cuales podemos destacar a: SPEA[2], SPEA2[3], NSGA-II[4], entre otros.

En trabajos anteriores se presentó la Optimización basada en Malla Variable (VMO) como una meta-heurística eficiente para problemas de optimización. Este algoritmo se basa en la representación de soluciones potenciales como nodos de una malla, cuya configuración varía en su movimiento por el espacio de búsqueda[5]. Este modelo utiliza dos operadores el de expansión el cual se encarga de generar nuevas soluciones utilizando distintas formas de generación de soluciones y el de contracción mediante este la malla es reducida utilizando un operador de limpieza adaptativo (clearing) que elimina todo nodo o solución que se encuentre cerca de otro de mejor calidad (fitness).

CAPÍTULO I

CONTEXTUALIZACIÓN DE LA INVESTIGACIÓN

1.1 Problema de investigación

1.1.1 Planteamiento del problema.

La mayor parte de los problemas de optimización del mundo real son naturalmente multiobjetivos. Esto quiere decir que, suelen tener dos o más funciones objetivo que deben satisfacerse simultáneamente y que posiblemente están en conflicto entre sí[6]. Sin embargo, a fin de simplificar su solución, muchos de estos problemas tienden a modelarse como mono-objetivo usando solo una de las funciones originales y manejando las adicionales como restricciones.

Al hablar de objetivos en conflicto, decimos que la optimización multiobjetivo hace que cada objetivo lleve al problema a una solución óptima diferente. Debido a esto, en lugar de encontrar una única solución óptima del problema, se encuentran un conjunto de soluciones de buena calidad según los diferentes objetivos considerados, las cuales son llamadas soluciones óptimas de Pareto[7].

Además cabe recalcar que en trabajos anteriores se presentó la Optimización basada en Malla Variable (VMO) [8] como una meta-heurística eficiente para problemas de optimización monoobjetivo, pero en la actualidad la optimización multiobjetivo está siendo tomada en cuenta para el desarrollo de proyectos, puesto que este tipo de Metaheurísticas obtiene soluciones más cercanas a la realidad.

1.1.2. Formulación del problema.

¿Cómo modelar el algoritmo de optimización monoobjetivo basado en mallas variables (VMO) en el ámbito de la optimización multiobjetivo?

1.1.3. Sistematización del problema.

- ¿De qué manera el algoritmo VMO no reconoce soluciones dominantes?
- ¿Cómo el algoritmo pierde soluciones en espacios donde aparecen muchas soluciones óptimas?
- ¿En qué medida la propuesta desarrollada al utilizar los problemas de optimización existentes encuentra resultados competitivos dentro del área?

1.2 Objetivos.

1.2.1 Objetivo general.

Formular una propuesta multiobjetivo de optimización basada en mallas variables que obtenga resultados competitivos en el área de la optimización multiobjetivo.

1.2.2 Objetivos específicos.

- ❖ Definir el operador para encontrar soluciones dominantes dentro del conjunto de soluciones obtenidas.

- ❖ Disminuir la pérdida de soluciones en el frente de Pareto.

- ❖ Evaluar el rendimiento de la propuesta utilizando problemas de estudio en el área de aplicación.

1.3 Justificación.

La optimización es un proceso que busca la mejor solución para un determinado problema mediante el uso de modelos matemáticos. Dentro de este campo tenemos específicamente los problemas de optimización multiobjetivo que tienen dos o más objetivos a resolver y son los problemas más cercanos a realidad. Razón principal por la que se seleccionó a este tipo de problemas como base de conocimiento en el desarrollo de esta investigación.

Dentro de los algoritmos utilizados para resolver los problemas de optimización una de las alternativas más utilizadas son los algoritmos de optimización, los cuales se basan en algoritmos genéticos y codifican un problema como una generación luego simulan mediante la aplicación de operadores matemáticos genéticos (selección, cruce y mutación). Al final logran determinar el mejor individuo o solución en las múltiples iteraciones de un número finito de generaciones. Entre los algoritmos de optimización encontramos algoritmos basados en Pareto enfocándonos principalmente en los de segunda generación puesto que estos utilizan el concepto de *elitismo* es decir escoger a los mejores. Entre los más destacados tenemos: PAES [9], SPEA II [3], NSGA II [12].

Por otra parte se han venido desarrollando modificaciones en metaheurísticas de optimización que en un principio fueron monobjetivo y después pasaron a ser multiobjetivo. Esto se debe a que las metodologías de optimización monobjetivo se concentran en obtener un conjunto de elementos para lograr mejorar el resultado dado por una función objetivo, pero sabemos que en los problemas reales se pueden involucrar otra serie de objetivos que también pueden ser de interés como el que se optimizó en un principio y esto ocasiona que la solución obtenida no sea la más óptima.

Por todo lo expuesto anteriormente se plantea el desarrollo de una propuesta multiobjetivo basada en la Metaheurísticas de Optimización de Mallas Variables (VMO) con el objetivo de obtener resultados competitivos dentro del área de Optimización Multiobjetivo.

CAPÍTULO II

FUNDAMENTACIÓN TEÓRICA DE LA INVESTIGACIÓN

2.1. Marco conceptual.

2.1.1 Optimización

Optimización es buscar las mejores soluciones para un problema del mundo real. El término es frecuentemente utilizado dentro del campo de la informática. Debido a que la optimización de software busca adaptar los programas informáticos para que realicen sus tareas de forma más eficiente posible. Además se debe mencionar que dentro de la optimización existen dos tipos de problemas: los problemas en los cuales se busca optimizar un único objetivo y la segunda que se pretende optimizar de manera simultánea dos o más objetivos. Siendo este último el que va a ser utilizado dentro del desarrollo de esta investigación.

Dentro de la optimización tenemos los problemas de optimización, según Landa[11] estos se describen mediante un conjunto de elementos, de los cuales, los más importantes son:

- **Variables de decisión.** Son valores que se modifican para resolver el problema.
- **Funciones objetivo.** Es necesaria al menos una. Esta se expresa en términos de variables de decisión, y el resultado de su evaluación es el que se desea optimizar.
- **Restricciones.** Desigualdades o ecuaciones que se tiene que cumplir para que la solución obtenida se considere factible. En ciertos problemas no se presentan restricciones en este caso todas las soluciones son válidas y el proceso de búsqueda se enfoca en optimizar la función objetivo.

2.1.2 Optimización multiobjetivo

Podemos considerar a la optimización multiobjetivo como la “verdadera optimización” puesto que los problemas reales tienen más de un objetivo que resolver. Es decir que ya no se busca una única solución sino se pretende encontrar un conjunto de soluciones posibles. Cabe mencionar que el principal objetivo de la optimización multiobjetivo es obtener el frente Pareto el cual fue propuesto por Pareto en 1896 [15]. Esto se lo obtiene gracias a que los algoritmos multiobjetivo exploran grandes espacios de búsqueda debido a que ya no se busca una única solución sino un conjunto de óptimos de Pareto. A continuación detallaremos ciertos conceptos acerca de lo que es el Frente de Pareto.

2.1.2.1 Definición formal

Formalmente, la optimización multiobjetivo espera encontrar un vector de decisión

$$\vec{x} = [x_1, x_2, \dots, x_n]^T \quad (1)$$

Generalmente con $\vec{x} \in \mathbb{R}^n$, que deberá satisfacer ω restricciones de desigualdad

$$g_i(\vec{x}) \geq 0 \quad i = 1, 2, \dots, \omega \quad (2)$$

Y optimizar el vector de funciones objetivos

$$\vec{f}(\vec{x}) = [f_1(\vec{x}), f_2(\vec{x}), \dots, f_b(\vec{x})]^T \quad (3)$$

Que generalmente cumple con $\vec{f}(\vec{x}) \in \mathbb{R}^b$.

2.1.3 Dominancia de Pareto

Sean dos soluciones $u, v \in \Omega$. Se dice que u domina v (denotado como $u \succ v$) si es mejor o igual v en cada uno de los objetivos y estrictamente en al menos uno de los objetivos.

Como ejemplo, en un problema de minimización $u \succ v$ si y solo si:

$$f_i(u) \leq f_i(v) \forall i \in \{1, 2, \dots, b\} \wedge \exists j \in \{1, 2, \dots, b\} | f_j(u) < f_j(v) \quad (4)$$

La Fig. 2.1 muestra una región formada la cual es denominada como región de dominancia del punto u . Todas las soluciones ubicadas en la región de dominancia de u son soluciones dominados por u .

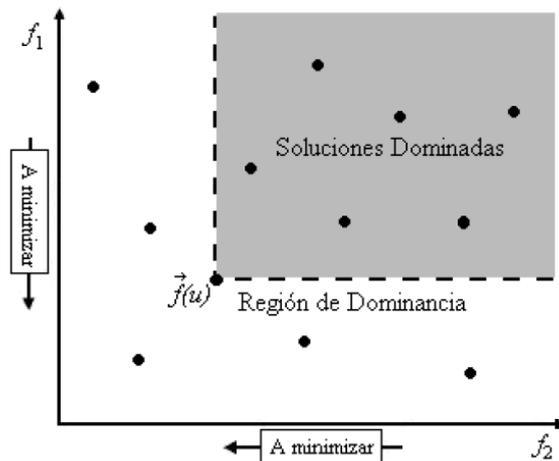


Fig. 2.1 Región de dominancia del punto $f(u) \in \Omega_0$

2.1.4 Optimalidad De Pareto

Al hablar de soluciones óptimas en los problemas de optimización multiobjetivo nos estamos refiriendo al concepto de Optimalidad de Pareto. A continuación se ilustran estos conceptos en la tabla 1, donde cada situación posible entre dos soluciones dadas $u, v \in \Omega$ genera una consecuencia sobre la comparación de las mismas.

Tabla 1. Teoría de Optimalidad de Pareto

SITUACIÓN	CONSECUENCIA
u domina v	u es mejor solución que v
v domina u	v es mejor solución que u
Ninguna de las anteriores	u y v son soluciones no comparables

Fuente: Desarrollo de una guía metodológica sobre computación evolutiva y algoritmos genéticos, para la optimización evolutiva multiobjetivo

Elaborado: Claudia Subía Pico

2.1.5 Conjunto de óptimos de Pareto

Cuando hablamos de soluciones optimas en problemas multiobjetivo, estas se deben entender bajo el concepto de optimalidad de Pareto. Decimos que una solución $\vec{x}^* \in F$ es Pareto- optima respecto a un problema de optimización multiobjetivo si y solo si no existe otra solución \vec{x} tal que \vec{x} domina a \vec{x}^* .

Según Coello[16] en 2002 manifiesta que al conjunto de soluciones se le denomina conjunto de óptimos Pareto (P). La imagen de este conjunto se denomina el frente de Pareto (PF).

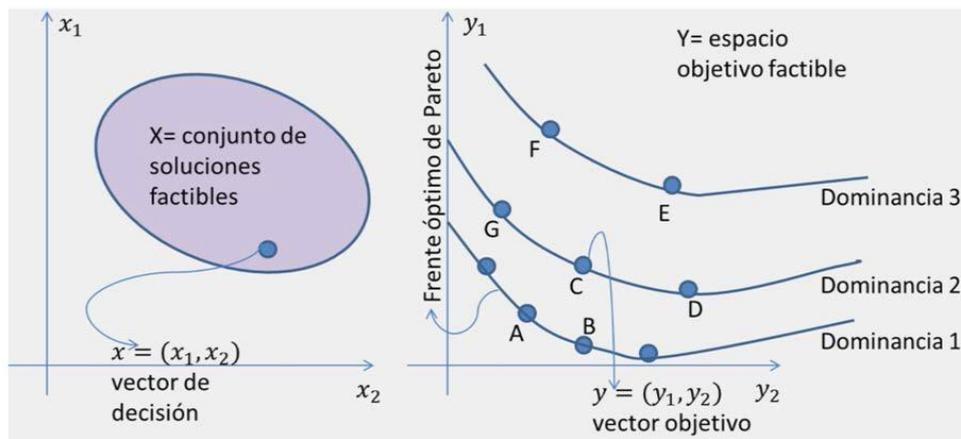


Fig. 2.2 Conjunto de soluciones optimas, frente de Pareto y su dominancia.

2.1.6 Frente de Pareto

A la imagen en y del conjunto óptimo de Pareto se lo conoce como Frente óptimo de Pareto o como Curva o Superficie de Pareto. La forma del frente de Pareto indica el tipo de compensación (trade-off) entre las diferentes funciones objetivo [16].

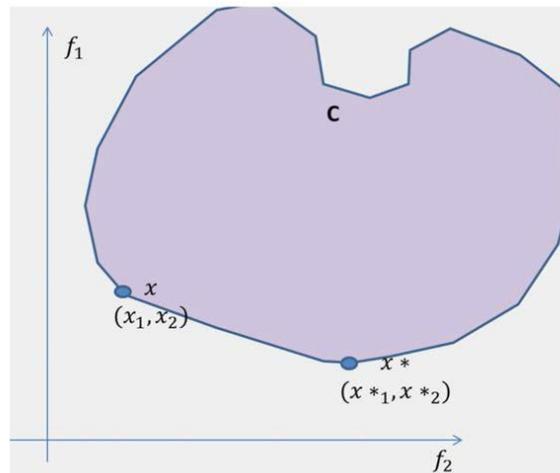


Fig. 2.3 Frente de Pareto

2.1.7 Metaheurísticas

Las meta heurísticas son estas estrategias generales para construir algoritmos, que quedan por encima de las heurísticas, y van algo más allá. Además podemos decir que las meta heurísticas pueden integrarse como un sistema experto para facilitar su uso genérico a la vez que mejorar su rendimiento.

2.1.8 Algoritmos representativos

Para resolver los problemas de optimización multiobjetivo se utilizan los algoritmos genéticos los cuales constituyen una técnica de búsqueda y optimización con un comportamiento altamente paralelo, este comportamiento está inspirado en el principio darwiniano que trata de la selección natural y reproducción genética. En este principio de selección los individuos más aptos, tienen mayor supervivencia y por lo tanto mayor probabilidad de reproducción. Los hijos de estos individuos tienen una mayor posibilidad de transmitir sus códigos genéticos a las generaciones siguientes [11].

Características principales de los algoritmos genéticos:

- Los algoritmos genéticos constan con un importante campo de aplicación en los problemas de optimización, los mismos que tienen diversos parámetros o conjuntos de variables que una vez combinados dan una solución.
- Se desarrollan codificando un conjunto de soluciones, no sobre las soluciones mismas.
- Trabajan en el conjunto de soluciones, mas no sobre una solución.
- Emplean procesos de transición probabilística para la búsqueda de la solución, no reglas determinísticas.

2.1.7.1 Esquema de un Algoritmo Genético

```
1:  $t \leftarrow 0$ 
2: inicializar  $P(t)$ 
3: evaluar los individuos en  $P(t)$ 
4: mientras no se satisfaga la condición de terminación haz
5:    $t \leftarrow t + 1$ 
6:   seleccionar padres  $C(t)$  de  $P(t - 1)$ 
7:   cruzar y mutar los individuos en  $C(t)$  para formar  $C'(t)$ 
8:   evaluar los individuos en  $C'(t)$ 
9:   seleccionar nueva población  $P(t)$  de  $C'(t)$  y  $P(t - 1)$ 
10: fin mientras
```

Fig. 2.4 Pseudocódigo de un Algoritmo Genético

2.1.8 Tipos de algoritmos multiobjetivo

Dentro de los problemas de optimización multiobjetivo los algoritmos se pueden clasificar dos tipos principales basados en el mecanismo usado para asignar el fitness:

Tabla 2. Clasificación de algoritmos multiobjetivos

ALGORITMOS BASADOS EN PARETO	ALGORITMOS NO BASADOS EN PARETO
Estos algoritmos se basan en la dominancia de Pareto para asignar el fitness. Como por ejemplo el SPEA, SPEA II, NSGA II, PAES, PESA entre otros.	Estos algoritmos usan diferentes mecanismos para la asignación del fitness. Como por ejemplo el algoritmo VEGA

Fuente: Desarrollo de una guía metodológica sobre computación evolutiva y algoritmos genéticos, para la optimización evolutiva multiobjetivo

Elaborado por: Claudia Subía Pico.

A continuación vamos hablar de los algoritmos más utilizados para resolver problemas de optimización multiobjetivo:

2.1.8.1 Algoritmo genético de ordenamiento elitista no dominado de segunda generación.

Este algoritmo evolutivo fue propuesto por Deb[4] para la optimización multiobjetivo y es uno de los más utilizados debido a su eficiencia y eficacia. El NSGA-II utiliza la jerarquización de Pareto la cual ha sido implementada de manera eficiente además se incluye un mecanismo que mantiene la diversidad de las soluciones.

Otra característica que sobresale en el algoritmo NSGA-II es el elitismo, es decir, que plantea un mecanismo para solo escoger a las mejores soluciones generadas durante el proceso de búsqueda. Esto se logra mediante la combinación en cada generación de la población de padres con la de los hijos logrando así que compitan entre sí para llegar a la siguiente generación.

2.1.8.2 Pseudocódigo del NSGAII

```
1: Algoritmo NSGA-II ( $G(N, E), s, T$ )
2: Inicio
3:   Para (cada  $t \in T$ ) haga
4:     Mientras que (exista una ruta del nodo  $s$  a nodo  $t$ ) haga
5:       Conjunto_rutas  $\leftarrow$  Ruta desde  $s$  hasta  $t$ ;
6:     fin mientras que
7:   Fin para
8:   Generar aleatoriamente  $P_0$ 
9:   Hacer ordenamiento rápido no dominado de  $P_0$ 
10:  Aplicar los operadores de selección, cruce y mutación para generar una población hija  $Q_0$ 
11:  Hacer  $t \leftarrow 1$ 
12:  Hacer  $R_t \leftarrow 1$ 
13:  mientras que  $t < g_{max}$  y  $con\_convergencia < g_{max}$ 
14:    hacer  $R_t \leftarrow P_t \cup Q_t$ 
15:    calcular el número de saltos y el retardo de los miembros  $R_t$ 
16:     $F\beta$  OrdenamientoRápidoNoDominado ( $R$ )
17:    Mientras que ( $|P_{t+1}| < N$ ) hacer
18:      Calcular DistanciaCrowding ( $F_i$ )
19:       $P_{t+1} \leftarrow P_{t+1} \cup F_i$ 
20:    Fin mientras que
21:    Ordenar  $P_{t+1}$  en forma descendente, utilizando el operador  $>_y$ 
22:    Escoger los  $N$  primeros elementos de  $P_{t+1}$ 
23:    Generar  $Q_{t+1}$ , aplicando los operadores de selección, cruce y mutación sobre  $P_{t+1}$ 
24:    hacer  $t \leftarrow t + 1$ 
25:  Fin mientras que
26: Fin
```

Fig. 2.5 Pseudocódigo del algoritmo NSGA II

2.1.8.3 Algoritmo genético PAES

Este algoritmo fue propuesto por Knowles y Corne[17] con sus siglas en inglés PAES (Pareto Archived Evolution Strategy). El algoritmo PAES funciona de la siguiente manera uno de los padres genera mediante la mutación una descendencia. Esta descendencia se compara con el padre, si la descendencia domina al padre entonces es aceptada como la siguiente matriz y la iteración continua. Si el padre domina la descendencia, esta es desechada y se genera una nueva solución mutada. Si el padre y la descendencia no se dominan entre sí, se utiliza un conjunto de comparación de individuos previamente no dominados.

2.1.8.3.1 Pseudocódigo del PAES

```
1: Generar solución aleatoria inicial c y añadirlo a archivo
2: Mutar c para producir m y evaluar m
3:   Si c domina m
4:     descartar m
5:     caso contrario
6:       si m domina c
7:         entonces reemplazar c con m y añadir m al archivo
8:         caso contrario
9:           si m es dominado por cualquier miembro del archivo
10:            descartar m
11:            caso contrario aplicar el test (c, m, archivo)
                para determinar cuál solución nueva se convierte en la
                actual y agregar m al archivo.
12:           fin de si
13:         fin de si
14:       fin de si
15: Fin hasta alcanzar un criterio de terminación
```

Fig. 2.6 Pseudocódigo del algoritmo PAES

2.1.9 Metaheurística de optimización de mallas variables (VMO)

El VMO fue propuesto por Puris [13] esta es una metaheurística poblacional en donde la población se distribuye mediante mallas de nodos que representan a las soluciones dentro del espacio de búsqueda. Este algoritmo realiza dos procesos para la exploración: el proceso de contracción y de expansión. El algoritmo VMO realiza los siguientes pasos: en primer lugar en cada ciclo, generan nuevos nodos los cuales van en dirección a los extremos locales y al extremo global; luego a partir de los nodos fronteras de la malla, todos estos pasos son realizados en el proceso de expansión. Y en el proceso de contracción de la malla, los mejores nodos provenientes de cada iteración son seleccionados como malla inicial para la iteración siguiente.

2.1.9.1 Descripción general de la Metaheurística VMO

La metaheurística VMO para realizar el proceso de expansión y generar nuevos nodos utilizando la población actual, realiza las siguientes fases:

- **Fase 1.** Generación de la malla inicial.- Aquí se generan las n posibles soluciones de forma aleatoria, las cuales representan a los nodos de la malla inicial.
- **Fase 2.** Generación de nodos hacia los mejores vecinos.- En esta fase a cada solución de la malla se buscan los nodos más cercanos. Luego se selecciona al vecino de mejor calidad(fitness), y si este vecino es mejor que la solución de la malla se genera un nuevo nodo entre ambos, caso contrario no se genera una nueva solución
- **Fase 3.** Generación de nodos hacia la mejor solución.- Aquí se crea un nodo nuevo para cada solución de la malla, pero en dirección del nodo con mejor valor de fitness de la malla actual.
- **Fase 4.** Generación de nodos utilizando los nodos fronteras.- Los nodos fronteras son las soluciones que se encuentran más lejos y más cerca del centro de la población. Y las soluciones en esta fase se generan desplazando cada nodo frontera en función del valor de desplazamiento.

Otra fase importante dentro del algoritmo VMO es la Fase Clearing puesto que gracias a esta fase se seleccionan las mejores soluciones existentes entre la malla actual y los nuevos nodos generados en el proceso de expansión. Tomando en cuenta que se debe evitar seleccionar las soluciones demasiado cercanas entre sí. Otra característica del clearing es que define una distancia d , como la distancia mínima que hay entre las soluciones, y ignora todo nodo que tenga una distancia menor a un nodo con mejor o igual calidad.

2.1.9.1 Pseudocódigo del VMO

```

1: Inicio
2: Generar  $M$  de manera aleatoria si es la primera iteración del algoritmo
3: Inicializar  $M^{temp} \leftarrow M$ 
4: Identificar la mejor solución  $n_g \in M$ 
5:   Para cada  $n_i \in M$ 
6:     obtener el vecino de mejor calidad
 $n_i^*$  (vecindad de tamaño  $k$ )
7:     Si  $n_i^*$  es óptimo local entonces
8:        $n_i^{fase 1} \leftarrow combinar(n_i, n_i^*)$ 
9:        $M^{temp} \leftarrow M^{temp} \cup n_i^{fase 1}$ 
10:   Fin para
11:   Para cada  $n_i \in M$ 
12:      $n_i^{fase 2} \leftarrow combinar(n_i, n_g)$ 
13:      $M^{temp} \leftarrow M^{temp} \cup n_i^{fase 2}$ 
14:   Fin para
15: Seleccionar  $n_s \in M$  (mayor distancia promedio)
16:  $n_s^{fase 3} \leftarrow Transformar(n_s)$ 
17:  $M^{temp} \leftarrow M^{temp} \cup n_s^{fase 3}$ 
18: Seleccionar  $n_r \in M$  (menor distancia promedio)
19:  $n_r^{fase 3} \leftarrow Transformar(n_r)$ 
20:  $M^{temp} \leftarrow M^{temp} \cup n_r^{fase 3}$ 
21: Ordenar  $M^{temp}$  de manera creciente en cuanto a calidad
22: Si selección elitista entonces
23:    $M \leftarrow primeros\ p\ elementos\ de\ M^{temp}$ 
24: Si selección representativa entonces
25:    $M^{temp} \leftarrow aplicar\ algoritmo\ a\ M^{temp}$ 
26: Si  $|M^{temp}| \geq p$  entonces
27:    $M \leftarrow primeros\ p\ elementos\ de\ M^{temp}$ 
28: Sino  $M \leftarrow M^{temp} \cup N, N$  nodos generados aleatoriamente

```

Fig. 2.6 Pseudocódigo del algoritmo VMO

2.1.10 Problemas de prueba

Dentro de la optimización multiobjetivo existe un amplio número de problemas que han sido propuestos para evaluar el desempeño de los algoritmos de optimización. A continuación se detallan el conjunto de problemas de prueba que serán utilizados dentro del desarrollo de esta investigación.

2.1.10.1 Problemas de Zitzler-Deb-Thiele

Estos problemas tiene varias características, pero la razón principal de su uso es porque tiene Frente de Pareto analíticamente calculables, permitiendocalcular de forma precisa la eficacia del conjunto de soluciones obtenidas. La Fig. 2.7 muestra la formulación genérica de los problemas ZDT.

$$\left\{ \begin{array}{l} \text{Minimizar } f(x_1, x_2, \dots, x_k) \\ \text{Minimizar } f_2(x) = g(x_{k+1}, x_{k+2}, \dots, x_n) \cdot h(f_1, g) \end{array} \right.$$

Fig. 2.7 Formulación genérica de los problemas ZDT

A continuación se detallan el conjunto de problemas utilizado para el desarrollo de esta investigación, indicando el nombre del problema, la cantidad de variables, dominio y detallando las diferentes funciones (f1), (f2) y (g) propios de la formulación general de cada problema.

Tabla 3.

Formulación de los Problemas ZDT

Nombre	N	Dominio	Funciones f_1, f_2	Funcion $g(x)$
ZDT 1	30	$x_i \in [0,1]$	$f_1 = x_1$ $f_2 = g(x) \left(1 - \left(\sqrt{x_1/g(x)} \right) \right)$	$1 + \frac{9}{N-1} \sum_{i=2}^N x_i$
ZDT 2	30	$x_i \in [0,1]$	$f_1 = x_1$ $f_2 = g(x) \cdot (1 - (x_1/g)^2)$	$1 + \frac{9}{N-1} \sum_{i=2}^N x_i$
ZDT 3	30	$x_i \in [0,1]$	$f_1 = x_1$ $f_2 = g(x) \cdot (1 - \sqrt{x_1/g} - (x_1/g) \sin(10\pi x_1))$	$1 + \frac{9}{N-1} \sum_{i=2}^N x_i$
ZDT 4	10	$x_i \in [0,1]$ $x_i \in [-5,5]$ $i = 2, \dots, n$	$f_1 = x_1$ $f_2 = g(x) \left(1 - \left(\sqrt{x_1/g(x)} \right) \right)$	$1 + 10(N-1) + \sum_{i=2}^N (x_i^2 - 10 \cos(4\pi x_i))$
ZDT 6	10	$x_i \in [0,1]$	$f_1 = 1 - e^{-4x_1} \sin^6(4\pi x_1)$ $f_2 = g(x) \cdot \left(1 - (f_1(x)/g(x))^2 \right)$	$1 + 9 \left(\left(\sum_{i=2}^N x_i \right) / (N-1) \right)^{0.25}$

Fuente: Una Versión Paralela del Algoritmo Evolutivo para Optimización Multiobjetivo NSGA-II y su Aplicación al Diseño de Redes de Comunicaciones Confiables

Elaborado: Zitzler-Deb-Thiele

2.1.10.2 Descripción de los problemas de optimización multiobjetivo

En la Fig. 2.7 se muestra la descripción de los problemas ZDT implementados.

Problemas de Optimización Multiobjetivo	
Nombre del Problema	Descripción
ZDT1	Este problema tiene 30 variables con un rango de [0,1]. Tiene frente de Pareto convexo y es el más sencillo del conjunto, además de tener una distribución uniforme de soluciones a lo largo del frente.
ZDT2	Este problema tiene 30 variables con un rango de [0,1]. Tiene frente de Pareto no convexo y la distribución de soluciones a lo largo del frente de Pareto es uniforme.
ZDT3	Este problema tiene 30 variables con un rango de [0,1]. Tiene frente de Pareto discontinuo y la distribución de soluciones es igual a los problemas anteriores.
ZDT4	Este problema tiene 10 variables con un rango de [0,1]. Su frente de Pareto es convexo. La complejidad de este problema está dada por la existencia de 100 frentes de Pareto locales en el espacio de soluciones.
ZDT6	Este problema tiene 10 variables con un rango de [0,1]. Su frente de Pareto es no convexo. La complejidad de este problema está dada por la combinación de la forma no convexa del frente de Pareto y la distribución no uniforme de soluciones a lo largo de él.

Fig. 2.7 Descripción de los Problemas ZDT

2.1.11 Software para optimización multiobjetivo

En la actualidad existen varios software que cuentan con algoritmos multiobjetivo y mono objetivo, problemas de optimización y otras características que hacen que estos programas hayan sido utilizados en varias investigaciones realizadas por otros autores. Uno de los principales software utilizados es Jmetal.

JMetal es un framework basado en Java que es utilizado para la optimización multiobjetivo usando técnicas meta heurístico. Tiene un esquema jerárquico de clases perfectamente documentado. Los algoritmos comparten los mismos componentes básicos de los operadores genéticos, facilitando de este modo el desarrollo de nuevas técnicas de multiobjetivo. También consta con varios algoritmos tradicionales, diversos problemas que estudian el desempeño de algoritmos, así mismo tiene un conjunto de indicadores de calidad los cuales miden grado de confiabilidad de los conjuntos de soluciones obtenidos por los algoritmos.

2.1.12 Estructura de Jmetal

En la Fig. 2.8 se muestra la descripción de las principales clases utilizadas por la herramienta Jmetal y en la Fig. 2.9 se puede observar un diagrama de bloques de la estructura de Jmetal.

Clases Principales de Herramienta Jmetal	
Nombre de la Clase	Descripción
SolutionSet	Es el tipo que se encarga de almacenar el conjunto de resultados óptimos que han sido evaluados y comparados y han pasado el criterio de aceptación del problema. Es por tanto, el conjunto de soluciones que formará el frente de Pareto. Si existe una solución que es igual de buena que otra en un objetivo pero en el otro objetivo es peor, se descartará automáticamente.
Algorithm	Esta clase se encarga generar las soluciones potenciales, y de realizar todas las evaluaciones necesarias para llegar a obtener un conjunto de soluciones finales, manejando todos los operadores definidos en el problema.
Problem	Es la clase donde se definen los parámetros que determinarán si una solución es o no aceptable. Tiene dos funciones importantes que son la Evaluate y la EvaluateConstraints, a las que accederá el algoritmo para evaluar una a una las soluciones generadas.
Operator	Un operador es una función que cambia algo en una o varias soluciones candidatas, con el fin de mejorarlas y que obtengan una mayor puntuación a la hora de su evaluación. Además, usando operadores, se consiguen soluciones más variadas y por tanto, más posibilidades de encontrar las soluciones óptimas. Existen al menos 3 tipos de operadores genéticos en jMetal: Mutation, crossover and selection.
Experiment	Es una clase diseñada para realizar experimentos, ya sea con varia combinaciones de problemas, algoritmos, parámetros, etc.

Fig. 2.8 Descripción de las principales clases de Jmetal

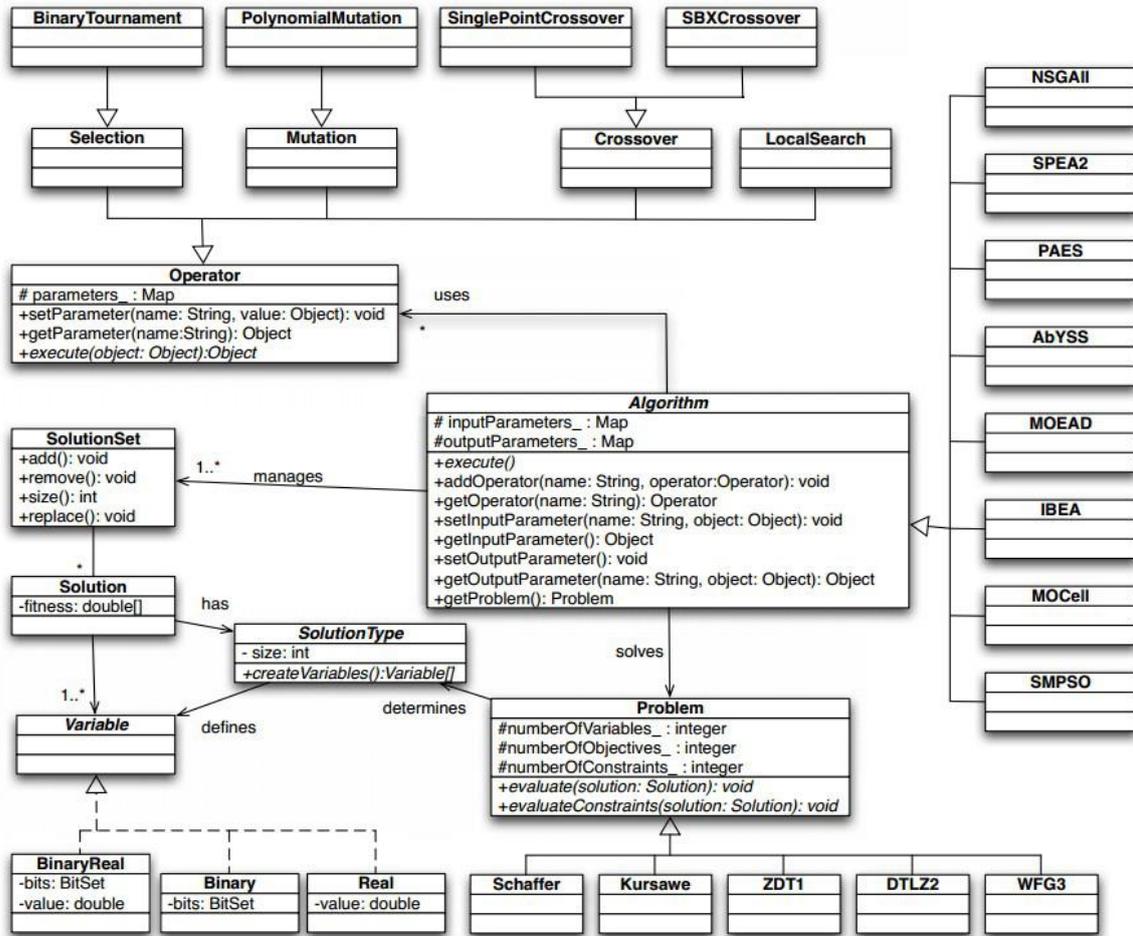


Fig.2.9. Diagrama de Bloques de Jmetal.

2.2. Marco referencial.

Para el desarrollo de este proyecto se tomaron como referencia los siguientes artículos de investigaciones:

OPTIMIZACIÓN MULTI OBJETIVO USANDO UN ALGORITMO GENÉTICO Y UN OPERADOR ELITISTA BASADO EN UN ORDENAMIENTO NO-DOMINADO (NSGA-II).

En este artículo se presentan conceptos básicos de la optimización multiobjetivo, los cuales están enfocados hacia la descripción del algoritmo NSGA-II, que es conocido en la literatura especializada por su buen desempeño y por los pocos cambios necesarios respecto al algoritmo mono-objetivo para ser implementado apropiadamente[1].

“OPTIMIZACIÓN MULTI OBJETIVO USANDO ALGORITMOS GENÉTICOS CULTURALES”

En esta tesis se diseñó e implementó un algoritmo cultural para resolver problemas de optimización multiobjetivo. El algoritmo propuesto en este trabajo fue la metaheurística NSGA-II. La propuesta consiste en incorporar un espacio de creencias a NSGA-II y utilizarlo como guía en la selección. Además, fue incorporado un procedimiento de búsqueda local basado en la información contenida en el espacio de creencias[10].

UNA VERSIÓN PARALELA DEL ALGORITMO EVOLUTIVO PARA OPTIMIZACIÓN MULTI OBJETIVO NSGA-II Y SU APLICACIÓN AL DISEÑO DE REDES DE COMUNICACIONES CONFIABLES

Aquí se presenta una versión Paralela del Algoritmo Evolutivo para Optimización Multiobjetivo NSGA-II, original de Deb et al. (2000). Además se agregan detalles de diseño e implementación de una versión paralela basada en subpoblaciones semi-independientes y se analiza la calidad de resultados y la eficiencia computacional, comparando con los resultados y tiempos de ejecución de la versión secuencial del algoritmo NSGA-II sobre un conjunto de problemas de prueba estándar[21].

“DESARROLLO DE UNA GUÍA METODOLÓGICA SOBRE COMPUTACIÓN EVOLUTIVA Y ALGORITMOS GENÉTICOS, PARA LA OPTIMIZACIÓN EVOLUTIVA MULTI OBJETIVO”

En este texto se buscó generar una guía que se enfocará en los principios básicos de la Computación Evolutiva y de la optimización multiobjetivo. Además se revisará la parte teórica así como los conceptos y técnicas desarrolladas para el diseño de algoritmos que permiten la solución de problemas multiobjetivo[11].

METODOLOGÍA PARA EL PLANEAMIENTO DE SISTEMAS DE DISTRIBUCIÓN DE ENERGÍA ELÉCTRICA CONSIDERANDO OPTIMIZACIÓN MULTI OBJETIVO

Este trabajo propone una metodología empleando como técnica de solución un algoritmo elitista de ordenamiento no dominado (NSGA-II), la cual considera la ubicación y capacidad de nuevos tramos de red, ubicación y capacidad de nuevas subestaciones, ampliación de subestaciones existentes y reconductorización de tramos de red existentes[7].

CAPÍTULO III

METODOLOGÍA DE LA INVESTIGACIÓN.

3.1. Localización.

Este texto se enmarca dentro del campo de la investigación y del desarrollo de nuevas propuestas, además cabe mencionar que el desarrollo de este documento se llevará a cabo como proyecto de investigación dentro de la Unidad de Titulación de la Universidad Técnica Estatal de Quevedo como requisito para la obtención del título de Ingeniería en Sistemas.

3.2 Tipo de investigación

3.2.1 Exploratoria

Este proyecto de investigación permitirá introducir una nueva propuesta dentro de los algoritmos de optimización multiobjetivo puesto que el algoritmo de mallas variables no había sido estudiado en este campo. Encontrando así mediante el estudio de otros algoritmos multiobjetivos los procedimientos adecuados para desarrollar una nueva propuesta basada en VMO.

3.3. Métodos de investigación.

3.3.1 Método analítico

El método analítico permitirá interpretar una situación de una manera analítica-crítica y así poder identificar las relaciones comunes y particulares que tiene el objeto de estudio. Para de esta manera poder comprender mejor el objeto de estudio y lograr alcanzar el fin propuesto en el inicio del proyecto de investigación.

3.4. Fuentes de recopilación de información.

Para la recopilación de información de este proyecto de investigación se utilizaron fuentes de información secundaria, puesto que dentro del campo de la optimización multiobjetivo existen varios artículos de revistas y libros de textos, que han basado sus investigaciones en algoritmos de optimización multiobjetivo ya existentes; así como también se han desarrollado nuevas propuestas.

3.5. Diseño de la investigación.

3.5.1 Diseño experimental

3.5.1.1 Diseño Cuasi-Experimental

En este proyecto de investigación, se empleará estatécnica de investigación la cual nos permite evaluar el impacto que tendrá el algoritmo de optimización multiobjetivo basado en mallas variables.

3.5.2 Pasos de un Cuasi Experimento

Este estudio será realizado mediante una medición del algoritmo desarrollado en comparación con otras metaheurísticas multiobjetivos ya existentes.

Esquema del diseño P1 X P2

Dónde:

X: Algoritmo de optimización multiobjetivo basado en mallas variables.

P1: Resultados obtenidos con el algoritmo desarrollado

P2: Resultado de las metaheurísticas ya existentes dentro del campo de estudio

3.6. Instrumentos de investigación.

Los instrumentos de investigación utilizados en el desarrollo de proyecto es el análisis de documentos, puesto que para obtener datos e información referente a la optimización se leyeron varios artículos que se encuentran dentro del campo de la meta heurística de optimización multiobjetivo.

3.7. Tratamiento de los datos.

En esta sección se utilizó una herramienta que pertenece a Microsoft office como los es Microsoft Excel. Mediante esta herramienta se obtuvo el promedio de los resultados obtenidos mediante la ejecución del algoritmo VMO- Multiobjetivo. Para luego realizar los gráficos correspondientes de cada una de las versiones del VMO-Multiobjetivo.

3.8. Recursos humanos y materiales.

Los materiales empleados durante este proyecto son:

3.8.1 Hardware

Tabla 4.

Materiales de Hardware

MATERIAL	DESCRIPCIÓN
COMPUTADOR	Utilizado en el desarrollo del proyecto de investigación. HP 15
IMPRESORA	HP Photosmart c4400

3.8.2 Software

Tabla 5.

Materiales de Software

MATERIAL	DESCRIPCIÓN
SISTEMA OPERATIVO	Microsoft Windows: Versión Windows 8
APLICACIONES	Microsoft Office 2010: Word PowerPoint Excel HerramientaJmetal

3.8.3 Suministros

Tabla 6.

Suministros

MATERIAL	DESCRIPCIÓN
INTERNET	Plan de Internet Banda Ancha de CNT
MATERIALES DE OFICINA	1 Caja de Lapiceros 1 Caja de Lápices 1 Resma de Hojas A4 3 CDSRW 2 DVD R 1 Borrador 2 Resaltadores 1 Regla 1 Grapadora 1 Caja de Grapas

3.8.4 Personal

Tabla 7.

Personal

PERSONAL	DESCRIPCIÓN
DESARROLLADORA:	Naira Haro Bustamante
DIRECTOR DE PROYECTO DE INVESTIGACIÓN:	Phd. Amilkar Yudier Puris Cáceres

CAPÍTULO IV

RESULTADOS Y DISCUSIÓN

4. Resultados y discusión

En este capítulo se presentarán los resultados generados por los indicadores de calidad SPREAD, HIPERVOLUMEN mediante la implementación de la nueva propuesta de VMO multiobjetivo en los problemas de optimización: ZDT1, ZDT2, ZDT3, ZDT4, ZDT6. Este conjunto de valores fueron obtenidos mediante la implementación del algoritmo VMO-Multiobjetivo en la Herramienta Jmetal. Posteriormente se realiza un estudio comparativo de los resultados obtenidos por la metaheurística VMO multiobjetivo con otro conjunto de soluciones de los algoritmos multiobjetivos más utilizados en el estado del arte.

4.1 Medidas de rendimiento

Cuando nos referimos a las medidas de rendimiento estamos hablando de los indicadores de calidad que tienen los algoritmos multiobjetivos. Para que el conjunto de soluciones encontrado sea considerado óptimo debe cumplir dos objetivos:

- Encontrarse lo más cerca posible del verdadero Frente de Pareto
- Tener una buena distribución sobre el Frente de Pareto

A continuación se revisará las métricas utilizadas dentro de la presente investigación:

4.1.1 Hipervolumen

El Hipervolumen es una métrica propuesta por Zitzler y Thiele[22] quienes lo definen como el tamaño del espacio cubierto o dominado. Este indicador es comúnmente utilizado para medir la calidad del conjunto de soluciones obtenidas por los algoritmos de optimización. Otra definición importante es de Bradstreet[23] quien manifiesta que esta métrica fue diseñada para medir los aspectos de convergencia y diversidad en un frente dado. Este indicador realiza el cálculo del volumen (en el espacio de objetivos) cubierto de los miembros de un conjunto dado Q , de las soluciones no dominadas para los problemas donde todos los objetivos han de ser minimizados.

Una de las características importantes de este indicador es que de las soluciones obtenidas de un problema en particular, aquella solución que maximice al hipervolumen está compuesta únicamente por los puntos de Pareto óptimos.

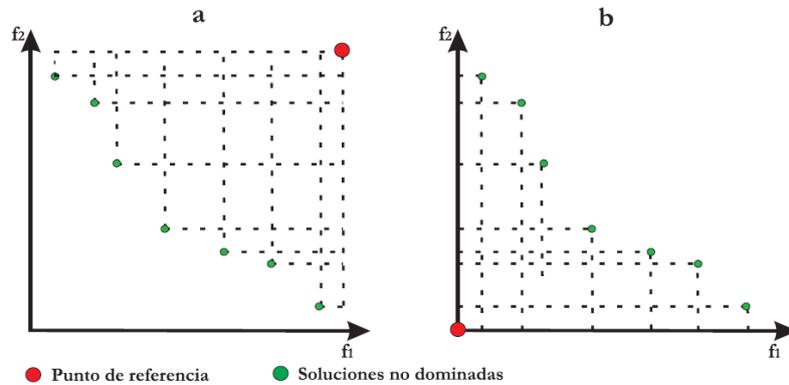


Fig.4.1. (a) El Hipervolumen centrado en un punto de referencia. (b) El Hipervolumen centrado en el punto de origen.

Matemáticamente, se describe de la siguiente forma:

$$I_h(\mathbf{P}) = \mathcal{S}(\mathbf{P}, \mathbf{y}_{ref}) = \Lambda(\cup_{\mathbf{y} \in \mathbf{P}} \{\mathbf{y}' | \mathbf{y} < \mathbf{y}' < \mathbf{y}_{ref}\}) \quad (5)$$

Este indicador tiene dos ventajas importantes, las cuales son:

1. Es sensible a cualquier tipo de mejora, es decir, cuando se calcula el hipervolumen para un conjunto A de soluciones, que domina a otro conjunto B de soluciones, entonces el aporte del hipervolumen será de mayor calidad para el primer conjunto que para el segundo.
2. Como resultado de la primera propiedad, el hipervolumen garantiza que, cualquier aproximación al conjunto A de soluciones que alcanza el valor máximo de calidad posible para un problema particular contiene todo el conjunto de óptimos de Pareto[11].

4.1.2 Spread

Esta métrica fue propuesta por Deb[4] también es conocida como dispersión y mide la diversidad a lo largo del Frente de Pareto (su valor ideal es cero). Esta es una medida de la distribución de los puntos contenidos en un frente, y se necesita conocer por lo menos los extremos del Frente de Pareto. Esta información adicional (los extremos del frente) permite tener una medida más precisa de la cobertura del frente de Pareto.

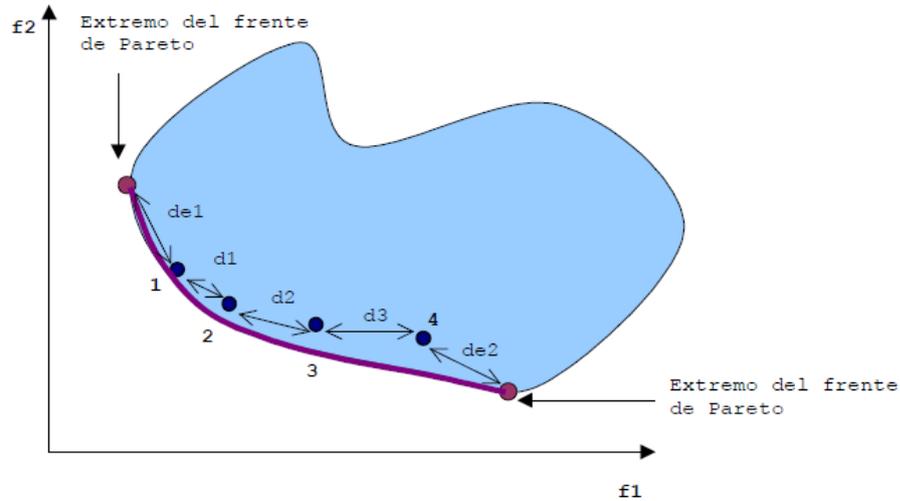


Fig. 4.2 Puntos involucrados en el cálculo de la métrica spread

Matemáticamente se lo escribe de la siguiente forma la Métrica Spread:

$$Spread = \frac{\sum_{k=1}^M d_k^e + \sum_{i=1}^q |\bar{d} - d_i|}{\sum_{k=1}^M d_k^e + q \cdot \bar{d}} \quad (6)$$

4.2 Modelamiento del Algoritmo VMO Multiobjetivo

En esta sección se detallaran las modificaciones realizadas a algunos elementos del algoritmo VMO para poder obtener la nueva propuesta multiobjetivo y luego aplicarla con los problemas de optimización multiobjetivo que se encuentran dentro de nuestra base de conocimiento.

4.2.1 Definición del nodo

Durante el proceso del desarrollo del algoritmo se utilizó la definición original de formar una malla de nodos que representan a las posibles soluciones. Aunque dentro del campo de la optimización multiobjetivo se definirá a un nodo como un array de números reales, y este representa a una posible solución dentro del conjunto de soluciones. Para este caso utilizamos la clase Solution que se encuentra implementada en la herramienta Jmetal.

4.2.2 Calculo de distancia entre nodos

Aquí nos referimos a la distancia que debe existir entre dos nodos a partir de sus estructuras. Cabe mencionar que para realizar este cálculo se implementó la función que se encuentra en el Jmetal. Y de forma general el proceso de calcular distancia, siempre será utilizado para obtener todas las distancias posibles de la malla.

4.2.3 Generación de nuevas redes

Este es un elemento principal de la fase de expansión de la malla puesto que aquí se define la manera en que se generan las nuevas soluciones a partir de las ya existentes. En el desarrollo de la nueva propuesta multiobjetivo VMO-MO se implementaron dos formas de generación de nodos:

- Generación de nodos hacia los mejores vecinos
- Generación de nodos hacia la mejor solución

A continuación se detallan estas dos formas:

4.2.3.1 Generación de nodos hacia los mejores vecinos

Este es el primer tipo de exploración que realiza el algoritmo VMO y se lleva a cabo en el vecindario de cada nodo (n_i). Para este proceso el vecindario de (n_i) está compuesto por los nodos k más cercanos (en términos de distancia) a la misma. Y luego se selecciona el mejor nodo (calidad) en el vecindario como el extremo local. Para calcular el factor de

cercanía y representar la relación entre el nodo actual y su extremo local utilizaremos la ecuación (7) tomando en cuenta que se necesita un intervalo de [0, 1].

$$Pr(n_i, n_i^*) = \frac{1}{1+|fitness(n_i)-fitness(n_i^*)|} \quad (7)$$

4.2.3.2 Generación de nodos hacia la mejor solución.

Esta parte es usada para acelerar la convergencia del algoritmo. Para realizar este proceso se explora en dirección del nodo que tiene mejor calidad dentro de la población actual, llamado el extremo de la malla global (n_g). X nodos nuevos se generan ($X=P-1$), uno para cada n_i hacia n_g . Utilizando la ecuación (8) para realizar los cálculos de este proceso.

$$n_x = G(n_i, n_g, Pr(n_i, n_g)) \quad (8)$$

4.2.4 Selección de las soluciones no dominadas

Este elemento consiste en seleccionar de un conjunto de soluciones las no dominadas. Para esto utilizamos tomamos en cuenta el cálculo del Frente de Pareto el cual se obtiene mediante los siguientes elementos:

- a) **Dominancia:** Consiste en que X_1 domina a X_2 (denotado por $X_1 < X_2$) si existe una función objetivo en la que X_1 sea mejor que X_2 sin ser peor en los otros objetivos. A continuación en la Fig. 4.3 lo antes mencionado de forma matemática.

$$X_1 < X_2 \Leftrightarrow \begin{cases} \forall i = 1, \dots, k \Rightarrow w_i f_i(X_1) \leq w_i f_i(X_2) \\ \exists j = 1, \dots, k: w_j f_j(X_1) < w_j f_j(X_2) \end{cases}$$

Fig. 4.3 Definición formal de Dominancia

- b) **Óptimo de Pareto:** Decimos que un punto x es Pareto óptimo siempre y cuando no exista ningún otro punto que pertenezca al conjunto de soluciones mejor en un objetivo sin ser peor en al menos otro.
- c) **Conjunto de Óptimos de Pareto:** Dentro de un problema multiobjetivo dado, se define al conjunto óptimo de Pareto como:

$$x^* \in X^* \Leftrightarrow \nexists x \in X: x < x^*$$

Fig. 4.4 Definición formal de conjunto óptimos de Pareto

- d) **Frente de Pareto:** Consiste en conjunto de puntos del espacio de búsqueda no dominados. El frente de Pareto se define como:

$$PF^* := \{ \vec{u} = \vec{f} = (f_1(x), \dots, f_k(x)) \mid x \in P^* \}$$

Fig. 4.5 Definición formal de Frente de Pareto

4.3 Integración del VMO en la Herramienta Jmetal

A continuación se detallara como el modelo VMO que es una metaheurística parecida a los algoritmos genéticos. Mediante la utilización de clases que se encontraban implementadas en el Jmetal y ciertas funciones propias del algoritmo VMO. Se logró integrar la aplicación VMO-MO a la herramienta de Jmetal.

Paso 1 [Generación malla inicial]: se genera una población inicial de M nodos (soluciones) generadas de manera aleatoria mediante la clase principal VMO_main.java esta clase tiene el método execute() el cual nos permite generar aleatoriamente el conjunto de soluciones iniciales, a cada una de las soluciones obtenidas se las evalúa utilizando el método evaluate() y evaluateConstraints() dependiendo del problema que se esté ejecutando. Este cálculo es almacenado en una variable clase que nos facilita en pasos más adelante saber si la solución domina o es dominada sin tener que volver a evaluarla.

Paso 2 [Generación por extremo local]: para este paso se calcula una matriz de distancias utilizando la clase Distance.java propia del jmetal, que obtiene la matriz de distancia entre los nodos de la malla, luego se selecciona entre las redes vecinas a una, la de mejor calidad y se obtiene una nueva red entre esta y la actual a través del método vmo_F_Funtion(valueX1, valueX2, distanceShare, Pr) que reciben como parámetro dos vectores (valueX1 y valueX2) que contiene los valores de la soluciones que se van a utilizar para la nueva generación, una distancia de búsqueda y la probabilidad.

Paso 3 [Generación por extremo global]: se selecciona aleatoriamente una solución entre todas las de la malla y se utiliza esta para obtener nuevas redes a través del método vmo_G_Funtion(valueX1, valueX2, Pr) que reciben como parámetro dos vectores (valueX1 y valueX2) que contiene los valores de la soluciones que se van a utilizar para la nueva generación y la probabilidad va representada de la solución nos da la ecuación (7).

Paso 4 [Seleccionar la malla inicial para la siguiente iteración]: para el proceso de contracción de la malla se utilizan el método ranking.getSubfront(i) de la clase principal Ranking.java que viene incluida en el software jmetal, este método nos permite obtener el frente de Pareto, es decir del conjunto seleccionado nos retorna el conjunto de soluciones dominante, las cuales serán utilizadas en la siguiente iteración.

Cabe destacar que después del procedimiento ejecutado en el apartado anterior, se le agrego un método el cual nos permite tener una memoria selectiva, es decir tener todas las soluciones dominantes resultantes de cada iteración del algoritmo para volver a calcular el frente de Pareto al final de las iteraciones.

Este proceso se repite hasta que se cumple con una cantidad definida de iteraciones.

4.3 Resultados de la implementación del algoritmo VMO Multiobjetivo

A continuación se detallan los resultados obtenidos mediante la implementación del algoritmo VMO Multiobjetivo en la Herramienta Jmetal. Tomándose en cuenta los siguientes aspectos:

- Se fijó un número promedio de 10 ejecuciones para cada problema.
- Se implementó una versión totalmente pura del algoritmo VMO, es decir no se utilizó ningún operador de la herramienta Jmetal.
- Se calcularon las medidas de desempeño como son: Hipervolumen y Spread y se seleccionaron los mejores resultados.

4.3.1 Gráficos de las soluciones obtenidos por el algoritmo VMO-Multiobjetivo

Problema	Indicadores	
	Hipervolumen	Spread
ZDT1	7,62E-01	5,41E-01

Fig. 4.6 Indicadores de desempeño del Problema ZDT1

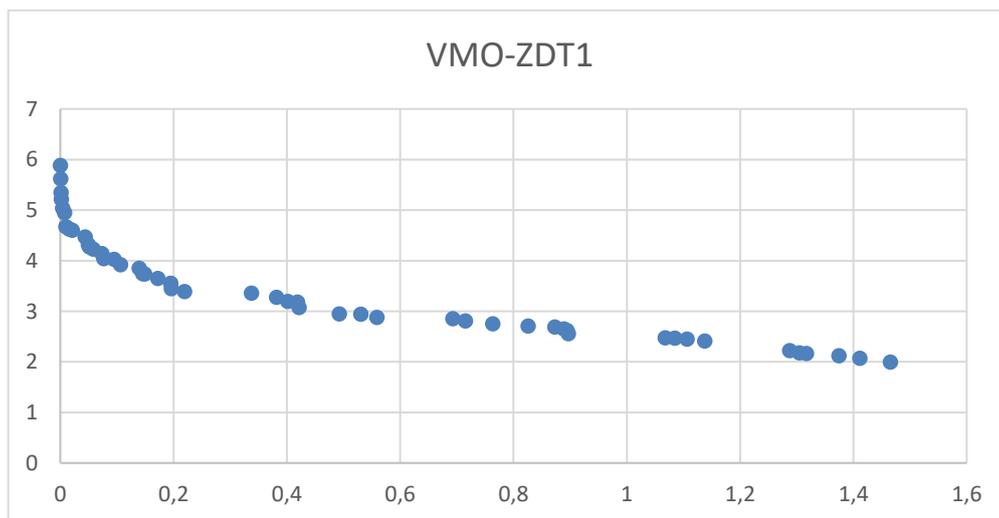


Gráfico 1. Soluciones generadas por el algoritmo VMO-MO para el problema ZDT1

Problema	Indicadores	
	Hipervolumen	Spread
ZDT2	5,06E-01	6,54E-01

Fig. 4.7. Indicadores de desempeño del Problema ZDT2

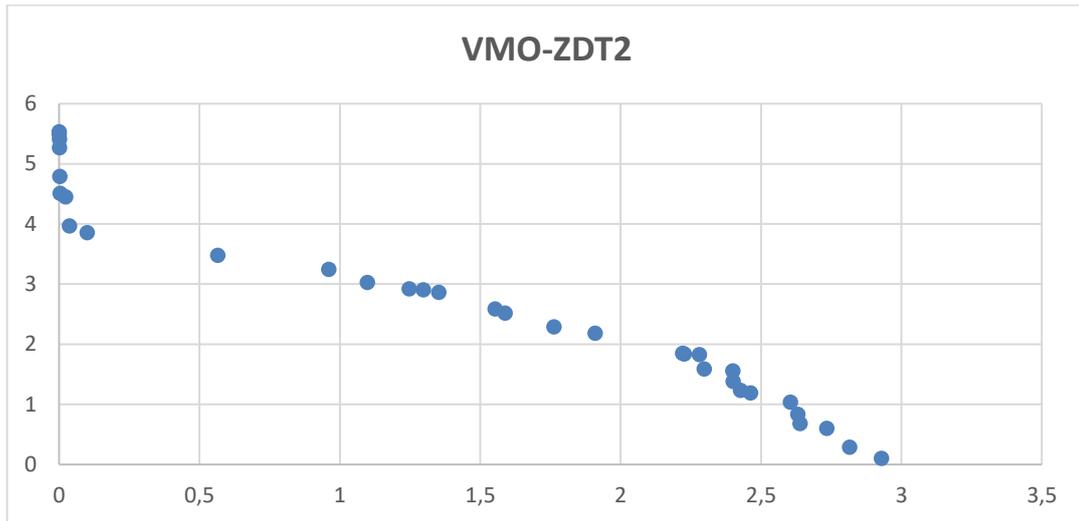


Gráfico 2. Soluciones generadas por el algoritmo VMO-MO para el problema ZDT2

Problema	Indicadores	
	Hipervolumen	Spread
ZDT3	6,79E-01	7,55E-01

Fig. 4.8. Indicadores de desempeño del Problema ZDT3

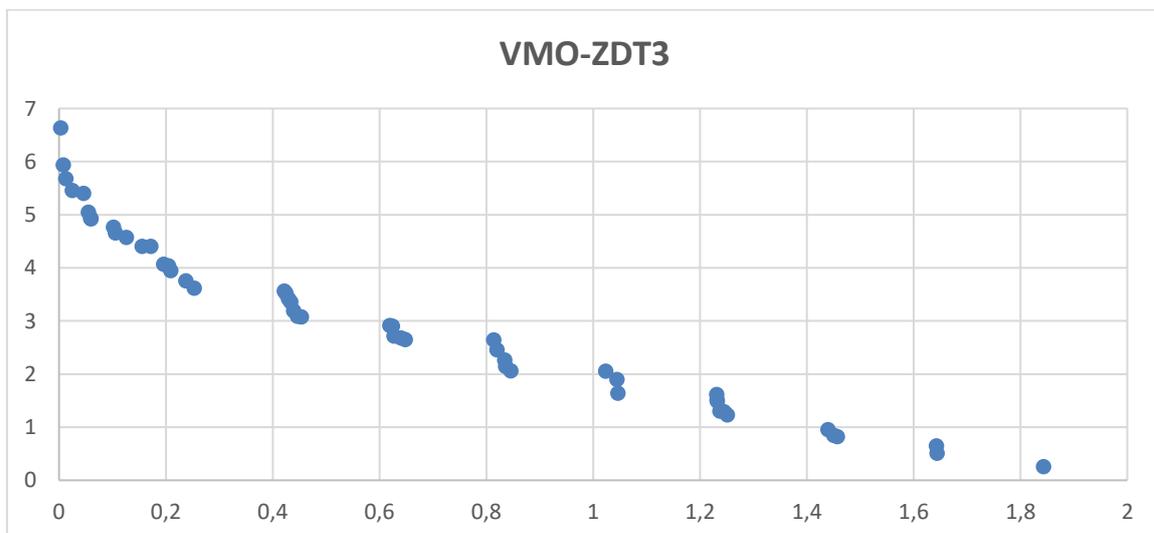


Gráfico 3. Soluciones generadas por el algoritmo VMO-MO para el problema ZDT3

Problema	Indicadores	
	Hipervolumen	Spread
ZDT4	6,89E-01	5,59E-01

Fig. 4.9. Indicadores de desempeño del Problema ZDT4

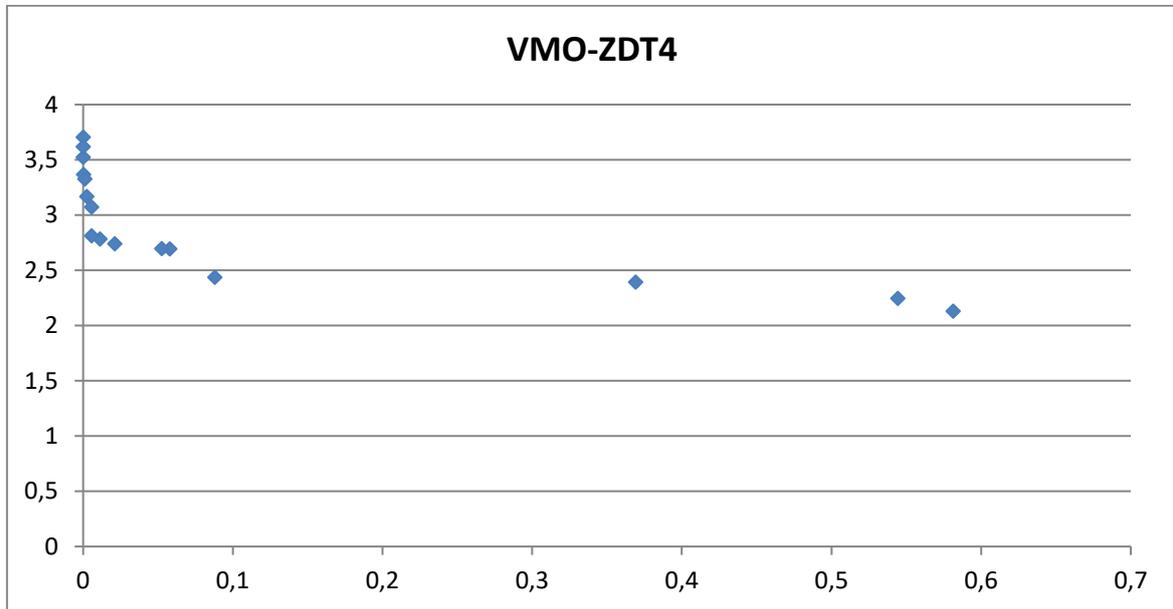


Gráfico 4. Soluciones generadas por el algoritmo VMO-MO para el problema ZDT4

Problema	Indicadores	
	Hipervolumen	Spread
ZDT6	4,92E-01	5,57E-01

Fig. 4.10. Indicadores de desempeño del Problema ZDT6

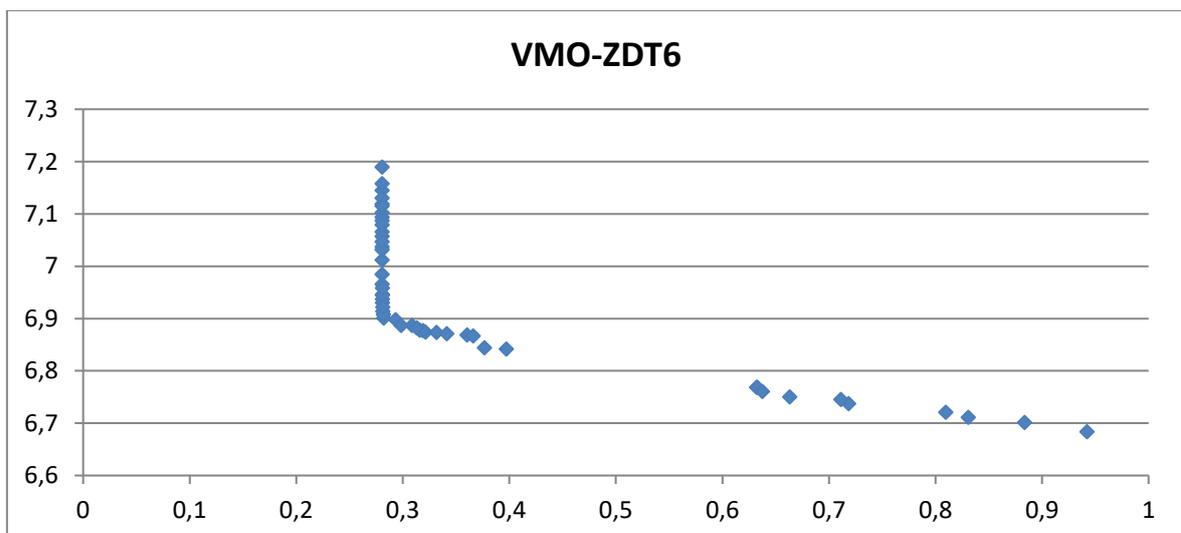


Gráfico 6. Soluciones generadas por el algoritmo VMO-MO para el problema ZDT6

4.3 Análisis de los resultados

El análisis de los resultados se basa en los indicadores de calidad Hipervolumen (convergencia y diversidad) y Spread (distribución), los cuales fueron obtenidos mediante la herramienta Jmetal. Realizaremos el análisis del algoritmo VMO-Multiobjetivo (VMO-MO) en comparación con otras metaheurísticas multiobjetivo importante como NSGA-II y PAES, seleccionando estas dos metaheurísticas por ser las más utilizadas en el estado del arte. Cabe recalcar que los valores que se muestran en la Fig. 4.11 y 4.12 sobre las metaheurísticas a comparar fueron obtenidas del artículo “*Análisis comparativo del Algoritmo MOLS con otras Metaheurísticas de Optimización Multiobjetivo: NSGAI y PAES*”.

En primer lugar analizaremos los resultados de convergencia y distribución obtenidos mediante el indicador Hipervolumen, estos se presentan en la Fig. 4.11, y como se puede observar el algoritmo VMO-MO tiene mejores resultados en todos los problemas ZDT, puesto que los valores generados son mayores a los otros dos algoritmos. Aunque podemos destacar que el algoritmo NSGA-II también genera buenos resultados pero solo en comparación con el algoritmo PAES.

Problemas	Algoritmos Multiobjetivos		
	VMO-MO	NSGA-II	PAES
	Hipervolume	Hipervolume	Hipervolume
ZDT1	7,62E-01	6,60E-01	6,58E-01
ZDT2	5,06E-01	3,27E-01	3,25E-01
ZDT3	6,79E-01	5,15E-01	5,11E-01
ZDT4	6,89E-01	6,59E-01	6,53E-01
ZDT6	4,92E-01	3,98E-01	3,97E-01

Fig. 4.11. Resultados del indicador Hipervolumen de los algoritmos VMO-MO, NSGA II Y PAES.

A continuación en el grafico 7 se muestra que el algoritmo propuesto VMO-MO se encuentra por encima en comparación a los algoritmos NSGA-II y PAES.

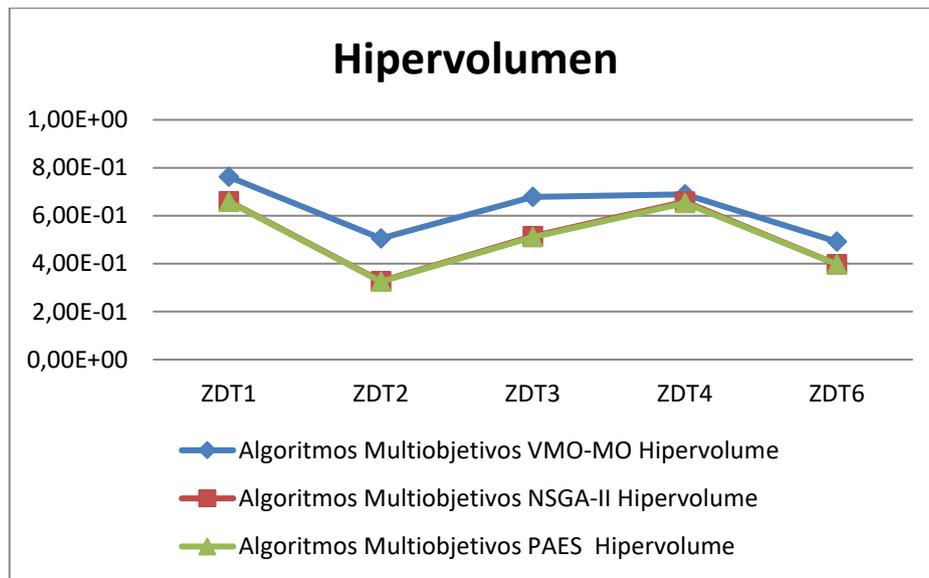


Grafico 7. Comparación de indicador Hipervolumen de los Algoritmos VMO-MO, NSGA-II y PAES

Ahora realizaremos el análisis de acuerdo a la diversidad de los frentes de Pareto obtenidos, mediante el indicador Spread, estos resultados se encuentran detallados en la Fig. 4.12. Aquí observamos que el algoritmo VMO obtiene mejores resultados solo en el problema ZDT6, puesto que tiene un valor menor que los otros dos algoritmos. Mientras que en este indicador en los problemas ZDT1 y ZDT2 obtiene los mejores resultados con el algoritmo NSGA-II, pero cabe recalcar que el algoritmo VMO-MO tiene buenos resultados en estos problemas en comparación con el algoritmo PAES. En cambio en los problemas ZDT3 Y ZDT4 los mejores resultados los genera el algoritmo PAES, y el VMO obtiene los peores resultados en comparación con el NSGA-II.

Problemas	Algoritmos Multiobjetivos		
	VMO-MO	NSGA-II	PAES
	Spread	Spread	Spread
ZDT1	5,41E-01	3,72E-01	7,08E-01
ZDT2	6,54E-01	3,80E-01	6,98E-01
ZDT3	7,55E-01	7,44E-01	1,03E-01
ZDT4	5,59E-01	4,04E-01	1,11E-01
ZDT6	5,57E-01	6,42E-01	7,69E-01

Fig. 4.12. Resultados del indicador Spread de los algoritmos VMO-MO, NSGA II Y PAES.

En el grafico 8 tenemos una visión más clara de cómo nuestra propuesta no obtiene los mejores resultados en comparación con los otros dos algoritmos, puesto que en el indicador de spread el valor cero es ideal en esta métrica. Y nuestro algoritmo obtiene el valor más bajo solo en uno de los 5 problemas estudiados.

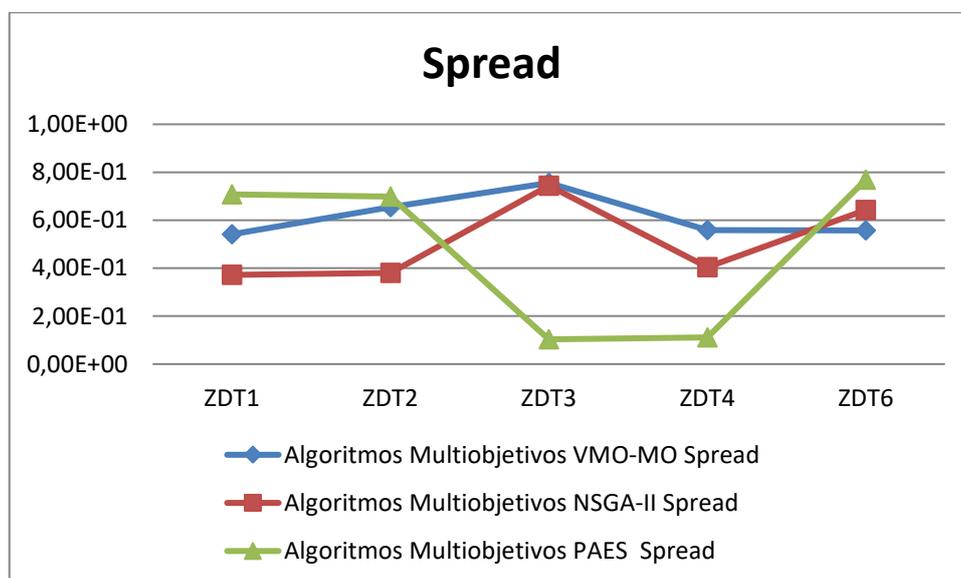


Grafico 8. Comparación de indicador Spread de los Algoritmos VMO-MO, NSGA-II y PAES

En la Fig. 4.13 se muestra una tabla general de los resultados obtenidos mediante la implementación del algoritmo de optimización multiobjetivo basado en mallas variables.

Problemas de optimización	Algoritmos multiobjetivo					
	VMO-MO		NSGA-II		PAES	
	Hipervolumen	Spread	Hipervolumen	Spread	Hipervolumen	Spread
ZDT1	7,62E-01	5,41E-01	6,60E-01	3,72E-01	6,58E-01	7,08E-01
ZDT2	5,06E-01	6,54E-01	3,27E-01	3,80E-01	3,25E-01	6,98E-01
ZDT3	6,79E-01	7,55E-01	5,15E-01	7,44E-01	5,11E-01	1,03E-01
ZDT4	6,89E-01	5,59E-01	6,59E-01	4,04E-01	6,53E-01	1,11E-01
ZDT6	4,92E-01	5,57E-01	3,98E-01	6,42E-01	3,97E-01	7,69E-01

Fig. 4.13Tabla general de los resultados obtenidos por el algoritmo VMO-MO

CAPITULO V

CONCLUSIONES Y RECOMENDACIONES

5.1. Conclusiones.

Las metaheurísticas, en particular los algoritmos multiobjetivos han sido tomados en cuenta para el desarrollo de varios proyectos investigativos, esto se debe al buen desempeño que tienen estas metaheurísticas multiobjetivo en una gran cantidad de problemas difíciles. En este trabajo se ha formulado una propuesta de optimización multiobjetivo basada en mallas variables la cual fue implementada dentro de la herramienta Jmetal.

En el desarrollo de esta propuesta se utilizó la versión pura del algoritmo de Mallas variables (VMO). Y también se redefinió en el proceso de expansión existente en este algoritmo, puesto que en este proceso se agregaron ciertas funciones matemáticas que fueron obtenidas de estudios realizados anteriormente sobre el algoritmo VMO. Además de estos cambios también fue utilizada la función Clearing la cual como se dijo en un capítulo anterior permite escoger los resultados más representativos que existen tanto en la malla actual como en los nodos recién generados y una memoria selectiva la cual nos permitió guardar las soluciones dominantes. Estas funciones fueron implementadas con el fin de lograr que el algoritmo VMO-Multiobjetivo (VMO-MO) muestre la mayor cantidad de soluciones posibles y así obtener resultados mejores en comparación con otros algoritmos clásicos dentro del campo de la optimización multiobjetivo.

Por último en base a los indicadores de calidad utilizados (Hipervolumen y Spread) se determina que la propuesta VMO-MO apunta a ser una metaheurísticas competitiva en el campo de la optimización multiobjetivo gracias a su buen rendimiento en comparación a algoritmos clásico como el NSGA-II y PAES. Puesto que se logra demostrar que el algoritmo propuesto en este trabajo genera soluciones con mayor convergencia y diversidad al Frente de Pareto de todos los problemas ZDT. En cuanto al indicador de distribución (spread) a pesar de obtener el mejor resultado solo en el problema ZDT6, el segundo mejor resultado en los problemas ZDT1 Y ZDT2 y perder en los problemas ZDT3 Y ZDT4. Esto puede ser a consecuencia de que el VMO-MO selecciona solo a las mejores soluciones y esto hace que el número máximo de funciones objetivos sea de 52 mientras que los otros dos algoritmos tiene 100 como número máximo de funciones objetivos.

5.2. Recomendaciones.

Luego de haber desarrollado el presente proyecto de investigación y de haber cumplido con las expectativas gracias al funcionamiento del algoritmo VMO-MO. Se recomienda que:

- Como trabajo de investigación futuro, se recomienda aplicar el algoritmo VMO-MO a la resolución de problemas reales; es decir, por ejemplo adaptar el algoritmo VMO-MO a problemas de optimización combinatoria o también implementarlo en la resolución de problemas en el campo de las telecomunicaciones.
- Explorar que otras funciones de los algoritmos multiobjetivos pueden ser implementadas dentro del algoritmo VMO-MO, para lograr mejorar los resultados obtenidos con la métrica Spread.
- Utilizar otros conjuntos de problemas para así tener una mejor visión de los casos para los cuales el algoritmo propuesto resulta ser el más adecuado.
- Identificar qué cambios podrían realizarse en los parámetros del algoritmo VMO-MO, para conseguir resultados mejores dentro del campo de la optimización multiobjetivo.

CAPITULO VI

BIBLIOGRAFÍA

Referencias Bibliográficas

- [1] CÉSAR AUGUSTO PEÑUELA, *OPTIMIZACIÓN MULTI OBJETIVO USANDO UN ALGORITMO GENÉTICO Y UN OPERADOR ELITISTA BASADO EN UN ORDENAMIENTO NO-DOMINADO (NSGA-II)*. Pererira : Scientia et Technica Año XIII, 2007.
- [2] y L. Thiele E. Zitzler, "Multiobjective Evolutionary Algorithms: A comparative Case Study and the Strength Pareto Approach," *Trans. Evolutionary Computation*, vol. 3, no. 4, pp. 257-271, 1999.
- [3] M. Laumanns, and L. Thiele E. Zitzler, *SPEA2: Improving the strength pareto evolutionary algorithm*. Zurich, Switzerland: Swiss Federal Institute of Technology (ETH), 2001.
- [4] K., Amrit, P., Agarwal, S., Meyarivan Deb, *A fast and elitist multiobjective genetic algorithm"*.: Technical report, Indian Institute of Technology, 2000.
- [5] Amilkar Puris, *Malla de Variables.*, 2012.
- [6] E. Uresti, *Optimizacion Multiobjetivo; una introduccion.*, 2014.
- [7] ANA MARÍA MEJÍA SOLANILLA, *METODOLOGÍA PARA EL PLANEAMIENTO DE SISTEMAS DE DISTRIBUCIÓN DE ENERGÍA ELÉCTRICA CONSIDERANDO OPTIMIZACIÓN MULTI OBJETIVO*. Pereira: UNIVERSIDAD TECNOLÓGICA DE PEREIRA, 2013.
- [8] Amilkar Puris, Rafael Bello, Daniel Molina , and Francisco Herrera, "Variable mesh optimization for continuous optimization problems," *Soft Computing*, pp. 511-525, 2012.
- [9] D.H. Wolpert and W.G. Macready, *No free lunch theorems for optimization*. *IEEE Transactions on Evolutionary Computation*.: IEEE Transactions on Evolutionary Computation, 1997.
- [10] Fernando Gutierrez Mendez, *Optimizacion Multiobjetivo Usando Algoritmos Geneticos Culturales*. Mexico: Centro de Investigacion y de Estudios Avanzados, 2011.
- [11] CLAUDIA MABEL SUBÍA PICO, *DESARROLLO DE UNA GUÍA METODOLÓGICA SOBRE COMPUTACIÓN EVOLUTIVA Y ALGORITMOS GENÉTICOS, PARA LA OPTIMIZACIÓN EVOLUTIVA MULTI OBJETIVO*. Quito: UNIVERSIDAD CATÓLICA DEL ECUADOR, 2014.
- [12] N. Srinivas y K. Deb, *"Multiobjective optimization using nondominated sorting in genetic algorithms*. India, 1993.
- [13] Amilkar Puris, *Estudio del desempeño de la optimización basada en mallas variables en problemas con óptimos en las fronteras del espacio búsqueda.*, 2012.
- [14] Ricardo Landa Becerra, "Algoritmos Culturales Aplicados a Optimización con Restricciones y Optimización Multiobjetivo," in *Algoritmos Culturales Aplicados a Optimización con*

Restricciones y Optimización Multiobjetivo. Mexico d.f, Mexico, 2002, ch. 1, p. 5.

- [15] Dr. Yván Jesús Túpac Valdivia, *Algoritmos Evolutivos en Optimización Multiobjetivos*.: Universidad Católica San Pablo, 2011.
- [16] Carlos A. Coello, *Introducción a la Optimización Evolutiva Multiobjetivo*. Málaga, 2002.
- [17] Ismael Valderrama Saltos Carlos Jordán Villamar, "Aplicación de SPEA2 al cálculo de esquemas de dosificación para el tratamiento quimioterapéutico del cáncer," in *Aplicación de SPEA2 al cálculo de esquemas de dosificación para el tratamiento quimioterapéutico del cáncer*. Guayaquil, Ecuador : Espol, 2014, p. 72.
- [18] Amilkar Puris, "Comparación del desempeño entre los algoritmos VMO, DE y ODE," 2012.
- [19] Luis Moreira Zamora, "Aprendizaje estructural de redes bayesianas utilizando la meta-heurística optimización basada en mallas variables ," in *Aprendizaje estructural de redes bayesianas utilizando la meta-heurística optimización basada en mallas variables*. Quevedo, Ecuador, 2014, pp. 22-25.
- [20] K. Deb, y L. Thiele E. Zitzler, "*Comparison of Multiobjective Evolutionary Algorithms on Test Functions of Different Difficulty*". Florida, 1999.
- [21] Sergio Nasmachnow, *Una Versión Paralela del Algoritmo Evolutivo para Optimización Multiobjetivo NSGA-II y su Aplicación al Diseño de Redes de Comunicaciones Confiables*. Uruguay: Instituto de Computación, 2004.
- [22] Álvaro Arranz Aranz, "Análisis práctico de algoritmos meta heurísticos multi objetivo en el modelado de parques eólicos.," in *Análisis práctico de algoritmos meta heurísticos multi objetivo en el modelado de parques eólicos*. Alcalá, España, 2014, pp. 18-21.
- [23] wordpress. (2008) Definicion.de. [Online]. [Definición de optimización - Qué es, Significado y Concepto http://definicion.de/optimizacion/#ixzz3kRi0ARFm](http://definicion.de/optimizacion/#ixzz3kRi0ARFm)
- [24] Christian von Lucken Augusto Hermosilla Benjamín Baran, *Algoritmos Evolutivos para Optimización Multiobjetivo: un Estudio Comparativo en un Ambiente Paralelo Asíncrono*. Paraguay: Universidad Nacional de Asunción, 2004.
- [25] Libardo López Ricardo Alberto Hincapié Ramón Alfonso Gallego, *PLANEAMIENTO MULTI OBJETIVO DE SISTEMAS DE DISTRIBUCIÓN USANDO UN ALGORITMO EVOLUTIVO NSGA-II*. Medellín: Escuela de Ingeniería de Antioquia, 2011.
- [26] Joel Prieto, Jorge Crichigno, Benjamín Barán Francisco Talavera, *Comparación de Algoritmos Evolutivos Multi-Objetivos en un ambiente Multicast*. Asunción: Centro Nacional de Computación, 2004.
- [27] K. Deb, y L. Thiele E. Zitzler, "*Comparison of Multiobjective Evolutionary Algorithms: Empirical*

Results,”: Evolutionary Computation, 2000.

- [28] D. E. Goldberg, *Genetic Algorithms in Search, Optimization and Machine Learning*. Massachusetts: Addison-Wesley Publishing Company, 1989.
- [29] D. A. van Veldhuizen, *Multiobjective Evolutionary Algorithms: Classifications, Analyses, and New Innovations*. Ohio, 1999.
- [30] C. A. Coello Coello, “A Short Tutorial on Evolutionary Multiobjective Optimization”, 2001.
- [31] K., Ooka, R Genku, “Application multi-objective genetic algorithm for optimal design method of distributed energy system”. Glasgow: Eleventh International IBPSA Conference, 2009.
- [32] R.A., Escobar, A.H., Toro, Gallego, “Técnicas metaheurísticas de optimización”. Pereira: Textos Universitarios, 2008.
- [33] Benjamn Baran, *Multi-Objective Optimization*. paraguay, 2006.
- [34] J. Horn y N. Nafpliotis, “Multiobjective Optimization using the Niche Pareto Genetic Algorithm. Illinois, 1993.
- [35] A. Beguelin, J. Dongarra, W. Jiang, R. Manchek, y V. Sunderam A. Geist, *PVM:Paralell Virtual machine - A user's guide and Tutorial for Networked parallel Computing*. Cambrigde, 1994.
- [36] y A. Bhaya E. Kaszkurewicz, “Parallel asynchronous team algorithms:Convergence and performance analysis., 1996.

CAPITULO VII

ANEXOS

5. Anexos

5.1 Tablas de funciones objetivos del Algoritmo VMO-MO

VMO-ZDT1	
0,051548646	4,275718281
0,015427016	4,626342115
0,095029231	4,029539197
0,004123827	5,038526367
0,021468949	4,601285087
0,073288938	4,141446066
0,145359638	3,746949988
0,076603301	4,043139392
0,196162862	3,447192885
0,049510315	4,319077355
0,194796433	3,556315248
0,043714475	4,468872557
0,057929015	4,232349535
0,172316122	3,65139474
0,149140435	3,738195361
0,009775422	4,674659288
0,106366151	3,924102807
0,337741022	3,359301817
0,71500219	2,810202658
1,137662917	2,412713323
0,826017885	2,709597587
0,219166363	3,389901972
0,692850677	2,853080044
0,492792961	2,945975863
0,000227708	5,886276894
0,13952317	3,849005673
0,401908899	3,192834954
0,007847253	4,949567462
0,001657742	5,212953706
0,381896399	3,277756496
0,888793952	2,652326837
0,418464037	3,180874053
1,106596303	2,450426809
0,559082988	2,879005425
0,000658042	5,620678077
0,421385323	3,07626627
1,084782679	2,470091859
0,872976651	2,692355663
0,530673325	2,944380605
0,763459054	2,752826531
1,465537479	1,993853874
1,411435874	2,070281476
1,287747681	2,223780616
1,304811747	2,181654054
1,06799593	2,478201454
0,001227743	5,345582766
1,316855967	2,166966736
1,37455084	2,123996574
0,896673863	2,555638325
0,894850205	2,617886945

Tabla 14. Valores de las funciones objetivos obtenidas mediante la ejecución del Algoritmo VMO-MO en el problema ZDT1

VMO-ZDT2	
0,00060771	5,53478377
0,00071347	5,49521105
0,00092664	5,41231
0,00100286	5,26888789
0,00246891	4,79230845
0,00377507	4,50808432
0,02283698	4,45310705
0,03639696	3,97107345
0,10023543	3,85824058
0,5653158	3,4806874
0,96030515	3,24874779
1,09760029	3,02962452
1,24687967	2,92472345
1,29709304	2,90838702
1,35256406	2,86409182
1,55303309	2,59056567
1,58838151	2,51905508
1,76276131	2,29126415
1,90939309	2,18288781
2,22048064	1,85073082
2,22679045	1,84127329
2,27972169	1,8308007
2,29784131	1,58735332
2,39938748	1,55590614
2,40106523	1,38158189
2,42686461	1,23494836
2,46196544	1,19035897
2,6047781	1,03831922
2,63092387	0,83798753
2,63859698	0,68315015
2,73454837	0,60352248
2,81531922	0,29038386
2,92884338	0,10197741

Tabla 15. Valores de las funciones objetivos obtenidas mediante la ejecución del Algoritmo VMO-MO en el problema ZDT2

VMO-ZDT3	
0,20521217	4,03715257
0,19624169	4,06821011
0,00335325	6,63234105
1,23734018	1,30370645
0,84585992	2,05505891
0,42215463	3,55855276
0,10230413	4,76050248
0,42953859	3,41383003
0,64072991	2,67480512
0,23752305	3,75281587
0,10589158	4,653732
0,20946798	3,947024
0,12613602	4,57268765
1,04487776	1,89554376
0,15598656	4,40150644
1,45751712	0,81832842
0,43912648	3,18658586
0,62724994	2,71434413
0,1720688	4,40059465
1,25119744	1,22918283
0,64828856	2,64695999
1,24555902	1,28588913
0,81387187	2,64077722
0,04615604	5,39783077
0,25342505	3,61645606
0,44672577	3,08421752
0,83649674	2,14127952
0,0129683	5,6760538
0,61945791	2,91053749
0,83472921	2,26146867
1,84336616	0,2557522
1,0233792	2,05479084
1,04643045	1,63685664
0,05913185	4,93740386
0,4535618	3,07233448
0,4337701	3,35378025
1,23141203	1,61069195
0,05526303	5,04621412
0,00816078	5,93596017
0,81995443	2,45530006
1,64423331	0,50940204
0,42466139	3,52146561
1,64264043	0,64295865
1,23246451	1,48896059
0,6242061	2,90010191
0,0596685	4,92108153
1,4508511	0,8437433
0,0249524	5,45593929
1,23216895	1,51292153
1,4399953	0,95488482

Tabla 16. Valores de las funciones objetivos obtenidas mediante la ejecución del Algoritmo VMO-MO en el problema ZDT3

VMO-ZDT4	
0,000200734	3,522116681
0,36923802	2,393029079
0,000258266	3,36830689
0,021174167	2,740203584
0,544374898	2,24620768
0,011265155	2,782961725
0,08794424	2,437248295
0,581456574	2,129135472
1,29776E-05	3,704322028
0,001210565	3,32648087
0,057847036	2,694126645
0,005741158	2,813266188
0,002477743	3,168888206
0,052491517	2,697216223
0,000180869	3,619686209
0,00559739	3,073225107

Tabla 17. Valores de las funciones objetivos obtenidas mediante la ejecución del Algoritmo VMO-MO en el problema ZDT4

VMO-ZDT6	
0,28207691	6,90110446
0,28077537	6,98447433
0,28110499	6,9300456
0,28096102	6,93687076
0,2809362	6,94501023
0,32145172	6,87390437
0,28133469	6,91393824
0,28077532	7,14477398
0,29329133	6,89803274
0,28184348	6,90779099
0,28077643	6,96584654
0,28091125	6,94602539
0,28123694	6,92138974
0,28084142	6,9584808
0,28077532	7,10204112
0,28077534	7,03691464
0,28077532	7,11872705
0,28077535	7,01169685
0,28077532	7,09404407
0,28077534	7,03129012
0,28077532	7,13043576
0,28077532	7,11539238
0,28077532	7,0659681
0,28077532	7,15792472
0,28077532	7,08714905
0,28077532	7,05685776
0,28077533	7,04693536
0,30875477	6,88688051
0,28077532	7,07933827
0,29857629	6,88690004
0,36033212	6,868808
0,31596638	6,87798076
0,33169107	6,87319786
0,31913243	6,87726725
0,34123667	6,87132714
0,31326966	6,88185833
0,28077532	7,18976118
0,88380994	6,70117411
0,63214702	6,76859907
0,66311843	6,75038539
0,37662341	6,84440682
0,36624692	6,86672992
0,39720404	6,84179624
0,63284353	6,76825576
0,94227552	6,68349817
0,71855287	6,73762494
0,83086829	6,71088091
0,71127037	6,74546531
0,80963891	6,72089717
0,63764054	6,76058125

Tabla 18. Valores de las funciones objetivos obtenidas mediante la ejecución del Algoritmo VMO-MO en el problema ZDT6